

SwissText 2026

**Proceedings of the 11th Edition of the
Swiss Text Analytics Conference**

June 10, 2026
University of Zurich
Zurich, Switzerland

The SwissText organizers gratefully acknowledge the support from the following sponsors.

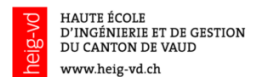
Sponsors



swissdox.ch



Partners





©2026 Association for Computational Linguistics

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

Introduction

This year, the 11th SwissText 2026 was hosted at the University of Zurich (UZH) on 10th June in Zurich, Oerlikon, and co-organized by SwissNLP, with over 100 participants attending keynotes, talks, and the poster session.

Our call for papers asked for contributions in four major tracks: Scientific Track (archival), Corpus Track (archival), Applied Track (non-archival), and Demonstration Track (non-archival). We received a total of 42 submissions, and selected 12 for oral presentations, and 25 as posters.

The SwissText 2026 proceedings contain:

- 20 papers for the **Scientific Track**, with technical research papers from the international scientific community, including corpus- and benchmark-related research papers with a focus on Swiss languages from the scientific community and industry,
- 2 papers for the **Corpus Track**, with Swiss-related NLP datasets.

We would like to thank our keynote speakers Alexandra Birch and Valentina Pyatkin. Their perspectives and contributions were much appreciated.

We are grateful to our sponsors and partners, who supported us. In addition, we would like to express our sincere gratitude to everyone who has contributed to this conference. Thanks to conference fellow members of the organizing committee, and the program committee for their excellent work.

Yours sincerely,

Rico Sennrich, Gerold Schneider, Yingqiang Gao, Tilia Ellendorff
and Jannis Vamvas, University of Zurich

Mark Cieliebak, Zurich University of Applied Sciences

General Chair:

Rico Sennrich, University of Zurich

Program Chairs:

Gerold Schneider, University of Zurich

Yingqiang Gao, University of Zurich

Tilia Ellendorff, University of Zurich

Jannis Vamvas, University of Zurich

Local Organizers:

Jannis Vamvas, University of Zurich

Yingqiang Gao, University of Zurich

Michelle Wastl, University of Zurich

Gerold Schneider, University of Zurich

Tilia Ellendorff, University of Zurich

Maria Christina Panagiotopoulou, University of Zurich

Program Committee:

Elliott Ash, Fernando Benites, Miriam Butt, Mark Cieliebak, Simon Clematide, Yingqiang Gao, Jonathan Gerber, Brigitte Krenn, Wolfgang Menzel, Margot Mieskes, Andrei Popescu-Belis, Annette Rios, Josef Ruppenhofer, Tanja Samardzic, Tatjana Scheffler, Helmut Schmid, Gerold Schneider, Manfred Stede, Don Tuggener, Ahmet Yavuz Uluslu, Junling Wang, Egon Werlen, Torsten Zesch, Heike Zinsmeister, Vilém Zouhar

Invited Speakers:

Alexandra Birch, School of Informatics, University of Edinburgh / Aveni.ai

Valentina Pyatkin, Allen Institute for AI / ETH AI Center

Volunteers:

Zachary Hopton

Annalena Alber

Sophia Conrad

Kirill Semenov

Jana Massoud

Keynote Talk
**From Metrics to Models: Reproducible Research in
Multilingual NLP**

Alexandra Birch

School of Informatics, University of Edinburgh / Aveni.ai

Abstract: *Reproducibility in multilingual NLP depends not only on open models, but also on evaluation methods that produce reliable scientific conclusions. In this talk, I connect insights from Extrinsic Evaluation of MT Metrics, which reframed metric assessment around downstream utility and robustness, with EuroLLM, an effort to make large-scale multilingual model development more transparent and accessible. Together, these works highlight two complementary pillars of reproducible research: building systems that others can replicate and measuring them in ways that others can trust and verify. I will discuss how this perspective has shaped the field's move toward open multilingual models, stronger evaluation practices, and more reproducible progress in AI.*

Keynote Talk
**Lessons from Building, Training and Evaluating Open
Language Models**

Valentina Pyatkin
Allen Institute for AI / ETH AI Center

Abstract: *This talk will outline the current landscape of fully open, open-weight, and closed language models, examining what distinguishes fully open models like Apertus from open-weights releases, and discussing the research and engineering considerations that shape open LLM development. Using OLMo and Apertus as case studies, I will walk through the full stack of decisions involved in training modern LLMs: from team dynamics and data curation to compute requirements and post-training methodology. The talk concludes with a discussion of the current capabilities and limitations of open models, with a particular focus on instruction-following capabilities as measured by IFBench.*

Table of Contents

1 Scientific Track	1
<i>Reinforcement Learning for Latent-Space Thinking in LLMs</i> Enes Özeren and Matthias Assenmacher	2
<i>Enhancing Retrieval via Cognitively Motivated Document Expansion</i> Giacomo Loss, Andreas Stephan and Matthias Assenmacher	18
<i>An Efficient Approach for Answering Not Readily Attainable Questions for RAG-based Applications</i> Zhengdao Chen, Christian Heumann and Matthias Assenmacher	30
<i>Concept Extraction and Webb’s Depth of Knowledge: Comparing LLM Question Generation Pipelines for Educational Assessment</i> Dmitriy An, Andrew Paice, Petra Müller-Csernetzky and Aliaksei Andrushevich	53
<i>Optimizing Large Language Models for Robust Domain-Specific Text-to-SQL: From Prompting to Preference Alignment</i> Noah Hampp, Katya Mirylenka and Michael Glass	64
<i>Call Support Copilot: A Reproducible Multimodal System for Speech Emotion Recognition, Intent Understanding, and Agent Assistance</i> Khoshimov Rakhmatillokhon, Dmitry Rudshin and Yanyang Luo	76
<i>Text vs. Phoneme Intermediates for Low-Resource Swiss German</i> Reza Kakooee, Vincenzo Timmel, Daniel Olivier Perruchoud, Michael Graber and Manfred Vogel	83
<i>Extending the Contact Hypothesis: Cross-Linguistic Evaluation of Religion and Nationality Bias When Prompting LLMs in German and Icelandic</i> Ikae Catherine, Guðbjörg Linda Rafnsdóttir, Ragna Kemp Haraldsdóttir and Mascha Kurpicz-Briki	91
<i>Robust Language Identification for Romansh Varieties</i> Charlotte Model, Sina Ahmadi and Jannis Vamvas	102
<i>Skill Extraction from Resumes and Job Offers across Six Languages</i> Laura Vásquez-Rodríguez, Bertrand Audrin, Samuel Michel, Samuele Galli, Julneth Rogenhofer, Jacopo Negro Cusa and Lonneke van der Plas	112
<i>RUMLEM: A Dictionary-Based Lemmatizer for Romansh</i> Dominic P. Fischer, Zachary Hopton and Jannis Vamvas	126
<i>Graph-Augmented LLMs for Swiss MP Ideology Prediction</i> Yifei Yuan, Luis Salamanca, Sophia Schlosser and Laurence Brandenberger	134
<i>Extracting Article-Level Legal Dependencies from Swiss Federal Law using LLMs</i> Steven Cho, Anna Kiepora, Jessica Lam and Richard Hahnloser	147
<i>Data Augmentation for Historical NER: A Systematic Comparison of Lexical and LLM-based Approaches</i> Léa Blinière, Maud Ehrmann, Emanuela Boros, Simon Clematide and Frederic Kaplan	155
<i>Code-Switching Detection in Multilingual Child Speech with SwissBERT</i> Agnese D’Angelo, Sina Ahmadi, Moritz M. Daum and Stephanie Wermelinger	172

<i>Which Skills Debate Reaches the Public? Comparing Scientific Literature and Media Coverage of AI and LLM Skill Impacts (2022–2025)</i>	
Maud Reveilhac, Gerold Schneider, Bohdan Trembovelskyi, Nisha Yadav, Vlada Druta, Aurelia Tamò-Larrieux, Simon Mayer, Clement Guitton and Joshua C. Yang	180
<i>The Same Email, Signed Differently: Testing Negotiation Bias and Recommendation Stability in LLMs</i>	
Jasmin Heierli and Alexandre de Spindler	192
<i>A Bounded Coordination-Support Capability for Multi-Party Settings: Task-State Monitoring in Firefighter Incident Command</i>	
David Grünert, Barbara Morgenstern, Florian Peczinka, Dario Holenstein, Roland Brunner and Alexandre de Spindler	197
<i>Can Large Language Models Replace Statistical Software?</i>	
Yves Staudt	206
<i>Automated German Alt Text Generation for News Charts</i>	
Alessia Vannini, Julia Locher, Marianne Santaholma, Claudia Amsler, Jonas Oesch and Arzu Coltekin	215
2 Corpus Track	225
<i>A Dataset of Latin Etymologies Extracted from Wiktionary</i>	
Javier de Torres Méndez, Marco Carlo Passarotti, Giovanni Moretti, Francesco Mambrini and Matteo Pellegrini	226
<i>How Good is AI on Swiss Voting Booklets? A Multilingual OCR and Alignment Benchmark</i>	
Elina Stüssi and Jannis Vamvas	234

Chapter 1
Scientific Track

Reinforcement Learning for Latent-Space Thinking in LLMs

Enes Özeren¹, Matthias Assenmacher^{1,2}

¹LMU Munich, Department of Statistics, Germany

²Munich Center for Machine Learning (MCML), Germany

Correspondence: enozeren@gmail.com matthias@stat.uni-muenchen.de

Abstract

Chain-of-Thought (CoT) reasoning typically utilizes the discrete language space for thinking, which is inherently inefficient, as many generated tokens only enforce linguistic rules that are not required for reasoning. To bypass this, *latent-space thinking* allows models to think using the continuous embedding space. While existing methods for training those models show domain-specific gains, they fail to maintain performance in complex tasks, such as mathematical reasoning. We experimentally demonstrate that the Coconut approach, a form of supervised fine-tuning for latent-space thinking, is highly sensitive to design choices and exhibits several inherent limitations. To address these issues, we investigate reinforcement learning (RL) techniques — an under-explored direction in latent-space thinking — including GRPO and design a novel *Latent RL* method for directly optimizing the latent thinking steps. Our experimental results reveal that these RL-trained models still lag behind traditional language-space CoT models in the mathematical reasoning domain. We make our codebase publicly available. ¹

1 Introduction

Language model (LM) training is typically divided into two phases: pre-training, where models acquire general language understanding and world knowledge, and post-training, which adapts them for downstream use cases such as dialogue, code completion, or translation (Grattafiori et al., 2024). Among the most widely used post-training methods is reinforcement learning (RL), which aligns LM behavior with human preferences (Ouyang et al., 2022; Touvron et al., 2023b). RL has also been applied to strengthen reasoning abilities in domains such as mathematics and programming (Shao et al., 2024). While effective, such methods incur substantial computational costs, as models trained in

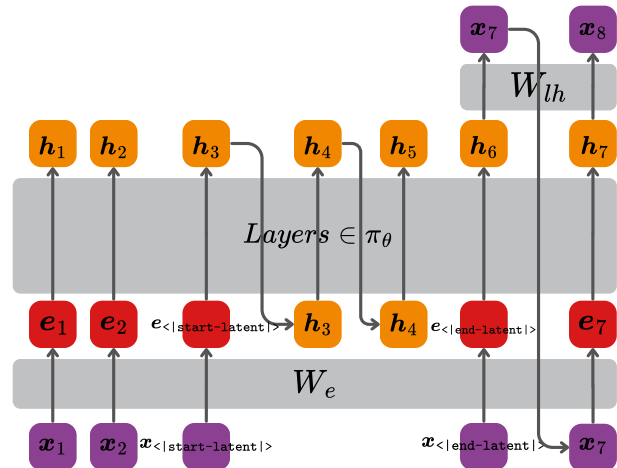


Figure 1: Illustration of latent-space thinking. Starting from the input (x_1, x_2), the $\langle \text{start-latent} \rangle$ is inserted, then the model switches to the latent mode and performs two latent steps ($C = 2$). Once the $\langle \text{end-latent} \rangle$ token is reached, the model switches back to language mode to answer in language-space (x_7, x_8).

this way tend to produce longer responses during inference (Guo et al., 2025; Zheng et al., 2025).

An alternative direction improves efficiency by shifting reasoning from the language space (discrete tokens) to the latent space (i.e., the continuous hidden states, see Fig. 1, Hao et al., 2025; Zhu et al., 2025b), inspired by neuroscience findings that language primarily serves communication rather than thought (Fedorenko et al., 2024). A notable example is the *Coconut* (Chain of Continuous Thought) method (Hao et al., 2025), which directly reuses final-layer embeddings as subsequent input, rather than decoding them into tokens. This allows the model to operate in its continuous latent space, avoiding the bottleneck imposed by discrete language tokens. However, training such models is non-trivial, as no ground-truth latent-space thinking² data exists. To address this, Coconut

¹github.com/enesozeren/latent-space-thinking-model

²With "Latent-Space Thinking" we refer to the same con-

follows the strategy of [Deng et al. \(2024\)](#), employing a supervised fine-tuning (SFT) approach where language-space reasoning steps are gradually replaced by latent-space thinking steps (referred to as *Coconut SFT* procedure). A crucial aspect of this procedure is that latent thinking steps do not receive direct supervision signals; instead, they are guided only through indirect signals during training. As a result, even though [Hao et al. \(2025\)](#) reported improvements in the logical reasoning domain, the performance drops in the mathematics domain when compared with the language-space thinking models.

In this work, we explore RL-based methods as a way to improve latent-space thinking models. Initial experiments applying group relative policy optimization (GRPO, [Shao et al., 2024](#)) show no measurable performance gains. As an alternative, we explore a novel RL method tailored to latent-space thinking to try to overcome the indirect learning signal limitation of the Coconut SFT method. However, this approach results in unstable training and fails to deliver performance improvements. Finally, we conduct an in-depth analysis of the Coconut SFT procedure, including ablation studies, which demonstrate that the procedure is sensitive to several factors, with performance being highly fragile. This analysis also evaluates several modifications, with the results indicating some promising adaptations as well as methodological weaknesses to be improved in future work.

The main contributions of this work are summarized as follows:

1. An empirical study of training latent-space thinking models with GRPO.
2. The introduction of a novel RL method tailored to latent-space thinking models.
3. An empirical in-depth analysis of the Coconut SFT procedure.

2 Related Work

Latent-Space Thinking In their recent survey paper, [Zhu et al. \(2025b\)](#) highlight the growing body of work on latent-space thinking. One of the earliest studies by [Goyal et al. \(2023\)](#) demonstrated that inserting dummy <pause> tokens before generating a response improves LM performance on

cept coined as "Latent Reasoning" by [Hao et al., 2025](#)). In our opinion, this is more adequate, as only a few latent steps are probably not yet real reasoning.

some tasks. This finding suggests that using more compute, even without producing language tokens, is beneficial for enabling the model to create richer (internal) representations before generating the answer. Similarly, [Pfau et al. \(2024\)](#) showed that even adding meaningless tokens like ‘.’ (dot) can achieve performance comparable to Chain-of-Thought (CoT) reasoning in certain problems.

Instead of focusing on latent thinking, [Deng et al. \(2024\)](#) aimed to internalize the CoT process within LMs. They proposed an SFT procedure that gradually removes explicit CoT steps from the training data, allowing the model to internalize the CoT process. Their results showed that this approach outperforms models trained without CoT, but still falls short of the performance achieved by models trained with CoT steps. Building on this idea, [Hao et al. \(2025\)](#) introduced the Coconut method, which incorporates latent thinking steps while also gradually removing the CoT steps during SFT. The goal is to enable the model to reason in an unrestricted latent space, rather than in the discrete language space. They define *latent reasoning* as feeding the last hidden layer’s last embedding back to the model as input, rather than decoding it into a token and feeding the corresponding token embedding in the next autoregressive iteration, as typically done in standard text generation. Their experiments show that this approach provides slight improvements on certain logical reasoning benchmarks but performs worse than explicit CoT in language space on the mathematical benchmark GSM8K ([Cobbe et al., 2021](#)). [Zhu et al. \(2025a\)](#) build on Coconut by developing a theoretical framework to explain why latent-space thinking can enhance LM performance on certain tasks, with a particular focus on the graph reachability problem.

Reinforcement Learning One of the first applications of RL for LM post-training was proximal policy optimization (PPO, [Schulman et al., 2017](#)), which laid the groundwork for reinforcement learning from human feedback (RLHF, [Ziegler et al., 2019](#); [Ouyang et al., 2022](#)) for LMs. It aligns models with human preferences by increasing the likelihood of responses favored by annotators. Because collecting human feedback for every output is costly, a reward model is trained on limited annotations to assign a scalar reward to each complete response ([Ouyang et al., 2022](#)). However, this single score gives sparse feedback for long outputs. To refine this, a value model ([Mnih et al., 2016](#))

predicts token-level contributions to the reward, offering more fine-grained feedback and improving training efficiency.

Recently, exploration of RL methods to enhance the reasoning capabilities of LMs has gained momentum (Shao et al., 2024; Lambert et al., 2024; Guo et al., 2025; Zheng et al., 2025). One of the first effective open-source approaches, Shao et al. (2024), introduced the GRPO method, which uses rule-based rewards tailored to mathematics and coding tasks—based on answer correctness or unit-test success. They reported consistent improvements during training and found that GRPO encourages progressively longer reasoning chains. Despite extensive exploration of SFT strategies in latent-space thinking models, the application of RL in this context remains largely unexplored.

3 Methodology

3.1 Latent Space Thinking

A LM π_θ with parameters θ processes text by tokenizing it into t tokens, each represented as a one-hot vector over the vocabulary V , forming $X^t \in \mathbb{R}^{t \times |V|}$. Tokens are embedded via the embedding matrix $W_e \in \mathbb{R}^{|V| \times D}$:

$$E^t = X^t W_e \quad (1)$$

The embeddings are contextualized by the model layers:

$$H^t = \text{Layers}(E^t), \quad H^t \in \mathbb{R}^{t \times D} \quad (2)$$

To predict the next token, the final hidden state h_t is projected into the vocabulary space using the language head $W_{lh} \in \mathbb{R}^{|V| \times D}$:

$$y_t = W_{lh} h_t, \quad p_t = \text{softmax}(y_t) \quad (3)$$

A decoding strategy selects the next token, and the process repeats auto-regressively until the end-of-sequence token is generated.

Latent-space thinking, as proposed by Hao et al. (2025), modifies the first and final step of the autoregressive generation process as illustrated in Figure 1. As a starting point, three new tokens, $\langle \text{start-latent} \rangle$, $\langle \text{latent} \rangle$, $\langle \text{end-latent} \rangle$, are added to the vocabulary and their embeddings are initialized randomly in the embedding matrix W_e . The $\langle \text{latent} \rangle$ acts purely as a placeholder for latent steps, and its embedding is never used by the model. When an input text X^t is provided, the $\langle \text{start-latent} \rangle$

token is directly appended as the $(t + 1)$ -th token, forming X^{t+1} .

Instead of proceeding with token generation using the language head matrix W_{lh} and a decoding strategy, the latent-space thinking model performs a latent step by taking the last contextualized token embedding h_{t+1} and appending it directly to the embedded sequence E^{t+1} . This is done via row-wise concatenation operation, denoted as concat :

$$E^{t+2} = \text{concat}(E^{t+1}, h_{t+1}) \quad (4)$$

This process, referred to as latent-space thinking, is repeated until a predefined number C of latent steps have been performed. After that, the $\langle \text{end-latent} \rangle$ token is appended. Following this, the latent-space thinking model reverts to the standard token generation in language space. To implement this mechanism, we used the Huggingface transformers library (Wolf et al., 2019) with targeted modifications to the generate and forward methods.

3.2 Existing Training Methods

Supervised Fine-Tuning For the language-space thinking model, we apply standard SFT without any modifications. Given a question-response pair (X^{q_i}, X^{r_i}) , we do one forward pass and calculate the cross-entropy loss on each response token:

$$\mathcal{L}_{\text{SFT}} = - \sum_{i=1}^N \log(\pi_\theta(X^{r_i} | X^{q_i})) \quad (5)$$

For the latent-space thinking model, we follow the Coconut SFT procedure (Hao et al., 2025). Given (X^{q_i}, X^{r_i}) , the model first processes all tokens in the question X^{q_i} , then generates latent steps auto-regressively as described in Section 3.1. After the latent steps are generated, the embeddings of the question tokens E^{q_i} , the latent steps E^{l_i} , and the response tokens E^{r_i} are concatenated and passed through the model in a single forward pass. During training, a cross-entropy loss is computed over the response tokens only; question tokens and latent steps are masked and excluded from the loss. Thus, the objective minimized during the Coconut SFT procedure is:

$$\mathcal{L}_{\text{SFT}} = - \sum_{i=1}^N \log(\pi_\theta(X^{r_i} | X^{q_i}, E^{l_i})) \quad (6)$$

This method optimizes the latent steps indirectly through the language space tokens in the response part.

(Modified) GRPO Applying RL to latent-space thinking models represents an unexplored research direction. We therefore experiment with the GRPO algorithm to investigate its impact. The language-space thinking model is trained with the GRPO implementation from the Hugging Face TRL package (von Werra et al., 2020). In contrast, training the latent-space thinking model with GRPO requires a modified version of this module. We modify the TRL package to exclude latent-step embeddings from the GRPO loss calculations since the GRPO loss requires token probabilities.

Similar to the Coconut SFT procedure, this modified GRPO setup optimizes the latent steps indirectly through the probabilities of tokens in the language space, and thus only through the final answer portion. In GRPO training, two types of rewards are used: an accuracy and a format reward (see Appendix A).

3.3 Proposing Latent RL

Neither Coconut SFT nor GRPO provides explicit supervision for training latent steps, as it is not straightforward to determine what constitutes a “good” latent step, since these steps are represented by continuous vectors and no ground-truth latent vectors are available.

To address this challenge, we propose an alternative RL approach, referred to as *Latent RL*. In the Latent RL training setup, a value model $\rho_{\theta+\phi}$ is employed to directly optimize the latent steps. The value model $\rho_{\theta+\phi}$ shares some of its parameters θ with the underlying policy model π_{θ} and is extended with additional value head parameters ϕ on top.

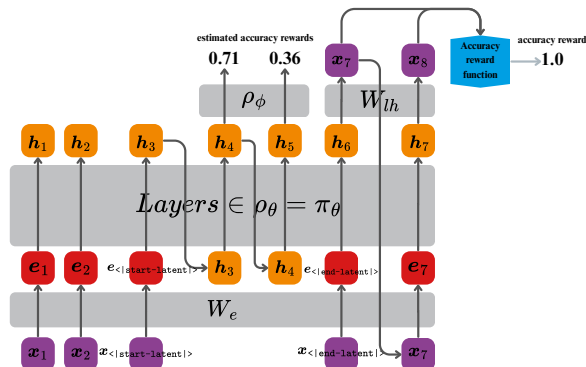


Figure 2: Illustration of Latent RL training. The value model head estimates the accuracy reward for each latent step.

The Latent RL training procedure is illustrated in Figure 2. In the first phase, we train $\rho_{\theta+\phi}$ by

freezing parameters θ . In the second phase, we train the policy model by utilizing the value model. To achieve this, we generate responses with latent steps for given questions and then maximize the value model predictions:

$$\arg \min_{\theta} L_{LAT}, \quad L_{LAT} = -\rho_{\theta+\phi}(\mathbf{h}) \quad (7)$$

Further details of the Latent RL procedure can be found in Appendix B.

3.4 Performance Measures

We primarily evaluate LMs for their mathematical capabilities and efficiency. Hence, the following two evaluation metrics are employed.

pass@k This metric estimates the expected probability that a model generates a correct answer to a question within its first k attempts (Chen et al., 2021; Guo et al., 2025; Dang and Ngo, 2025). It is defined as

$$\text{pass}@k = \mathbb{E}_{\text{problems}} \left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}} \right], \quad (8)$$

where n denotes the number of generated answers for a given question and c denotes the number of correct answers among them. We set $n = 4$ and report both pass@1 and pass@4.

Tokens The average number of tokens generated by the LM to produce an answer is a measure of output length and efficiency.

4 Experimental Setup

4.1 Models

The base LM used in this study is Qwen2.5 1.5B Base (Yang et al., 2024). This choice strikes a balance between computational feasibility and model capability in this study. For brevity, the model is referred to simply as Qwen Base. All methods and implementations proposed here are model-agnostic and can be applied to other decoder-only transformer LMs.

To analyze the effects of different training methodologies, we train and compare the models as outlined in Table 1.

4.2 Datasets

Since we employ two SFT and RL training methodologies, each with distinct requirements for the training data, we utilize different datasets for each of them (see Table 2).

Model Name	Training
Qwen Base	Base model
Qwen SFT	Qwen Base + SFT
Qwen SFT & GRPO	Qwen SFT + GRPO
LaST SFT	Qwen Base + Coconut SFT
LaST SFT & GRPO	LaST SFT + GRPO
LaST SFT & Latent RL	LaST SFT + Latent RL

Table 1: Models compared in this study. LaST is short for *Latent-Space Thinking Model*.

Dataset	Purpose	Usage Vol.
GSM8K-Aug-NL	Training (SFT)	300K
OpenR1-Math-220k	Training (RL)	10K
GSM8K	Testing	1.3K
MATH500	Testing	500

Table 2: The volumes of the datasets used in this study. K denotes thousands. For the training part, full datasets are not utilized for efficiency purposes. The Usage Vol. column reports the amount we used in our experiments.

For SFT, the primary requirement is that the dataset must contain question-response pairs. The Coconut SFT procedure, however, has the additional requirement that the responses X^{r_i} must be structured as a list of sequential steps. This means that, e.g., the response should contain a sequence of sentences or mathematical formulations, each corresponding to an individual, *separate* reasoning step. To satisfy these requirements, the GSM8K-Aug-NL dataset (Deng et al., 2023) is selected for SFT.

For pure RL training methods, a dataset containing question-answer pairs (X^{q_i}, X^{a_i}) is sufficient; the full response³ is not necessary. We use OpenR1-Math-220k (Hugging Face, 2025).

4.3 Training Details

The models in Table 1 are trained using the datasets in Table 2. The details of the training are given in this section. All the training hyperparameters are provided in Appendix C.

Qwen SFT. This model is obtained by SFT training of the Qwen Base model on 300K question-response pairs from GSM8K-Aug-NL.

Qwen SFT & GRPO. This model is trained with the GRPO algorithm using rule-based rewards, starting from the Qwen SFT model. The training

³We strictly differentiate *answers* (just the final result) from *responses* (including the whole reasoning trace) to emphasize the different requirements.

is performed on 10K samples from OpenR1-Math-220k.

LaST SFT. This model is trained using the Coconut SFT procedure proposed by Hao et al. (2025), starting from the Qwen SFT model and using the same 300K question-response pairs from GSM8K-Aug-NL as in the Qwen SFT training process. This procedure follows a multi-stage approach: in each stage, one language step (typically one sentence) from the ground-truth response is removed and replaced with two latent steps. In the final stage, all language steps are removed, and the model relies solely on latent steps to predict the final answer. Consistent with Hao et al. (2025), a three-stage approach was adopted, allowing the model to use up to six latent steps.

A key difference to Hao et al. (2025) lies in the choice of training data: instead of the GSM8K-Aug dataset, which contains only mathematical calculation steps, the GSM8K-Aug-NL dataset was used, where the steps consist of both language tokens and mathematical calculations. This choice was made to align with the main motivation for introducing latent steps - namely, to enable the model to internalize not only mathematical calculations but also language.

LaST SFT & GRPO. This model is trained on the same 10K question-answer pairs from the OpenR1-Math-220k dataset as used for the Qwen SFT & GRPO model, but starting from the LaST SFT model.

LaST SFT & Latent RL. The Latent RL method consists of two stages. First, the value model is trained to estimate accuracy rewards for latent steps. For this purpose, 20K question-answer pairs from the OpenR1-Math-220k are selected, and responses are generated with the LaST SFT model. Accuracy rewards for the generated responses are computed using the reward function, and the value model is trained to predict these rewards from the latent steps. The value head itself consists of a linear layer that maps each latent step embedding to a scalar score representing the estimated reward. In the second stage, training is performed using 10K question-answer pairs from the OpenR1-Math-220k dataset, distinct from those used for value model training. These pairs are the same pairs as used in the LaST SFT & GRPO model training to be consistent.

Model	Shot	GSM8K			MATH500		
		pass@1	pass@4	# Tokens	pass@1	pass@4	# Tokens
Qwen Base	Zero	24.8	49.7	156	12.1	27.4	310
Qwen Base	One	57.0	73.4	575	25.4	40.2	589
Qwen SFT	Zero	62.6	76.0	105	22.0	32.0	290
Qwen SFT & GRPO	Zero	72.6	82.5	216	38.7	50.0	366
LaST SFT	Zero	22.6	29.9	19	8.5	12.4	19
LaST SFT & GRPO	Zero	21.8	24.1	19	9.2	14.0	18
LaST SFT & Latent RL	Zero	22.7	30.1	19	8.4	13.2	19

Table 3: Evaluation results on GSM8K and MATH500 benchmarks.

5 Results

This chapter summarizes the main conclusions drawn from the experiments (cf. Table 3) and an ablation study for Coconut SFT.

5.1 Main Results

Evaluating Qwen Base is challenging. In the zero-shot setting, the Qwen Base model often fails to follow the instructed answer format, making it difficult to evaluate performance based on boxed answers in the answer block (see Appendix D for more details). To address this, the model was also tested in a one-shot setting with a single example showing the expected format. However, prior work shows that few-shot prompting, including one-shot, can boost performance beyond format compliance (Brown et al., 2020; Wei et al., 2022), so the model’s true zero-shot ability likely lies between its zero- and one-shot results presented here. Notably, in one-shot evaluation, the model sometimes hallucinated additional questions and answers, resulting in notably longer responses.

SFT on GSM8K-Aug-NL improves the performance on GSM8K. The Qwen SFT model outperforms the Qwen Base model under both zero-shot and one-shot prompting on the GSM8K benchmark. This improvement can be attributed to the use of the GSM8K-Aug-NL dataset during SFT, which contains questions similar to those in GSM8K and thereby enhances domain-specific performance. On the MATH500 benchmark, however, the Qwen SFT model surpasses the Qwen Base model under zero-shot prompting but performs worse than under one-shot prompting. This indicates that the fine-tuning provides less benefit for MATH500, suggesting more limited transfer

compared to the gains observed on GSM8K.

GRPO on OpenR1-Math-220k improves the performance and increases the response length. The Qwen SFT & GRPO model, obtained through training with the GRPO algorithm with rule-based rewards, achieves higher performance on both benchmarks compared to the Qwen SFT model. In addition, the average response length increases after GRPO training, consistent with previous studies (Guo et al., 2025; Dang and Ngo, 2025).

Coconut SFT harms model performance. Similar to the findings in Hao et al. (2025), the LaST SFT model trained with the Coconut SFT procedure performs worse than the Qwen SFT model on mathematical benchmarks. And since the model only produced six latent steps prior to the answer block, the number of tokens generated is significantly lower than other language-space thinking models.

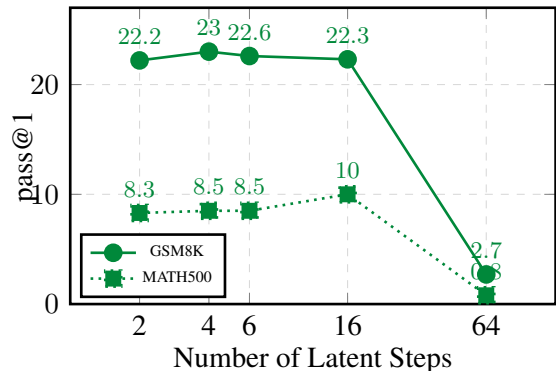


Figure 3: Effect of varying the number of latent steps on the pass@1 score with the LaST SFT model. The x-axis has the log scale.

To examine the influence of the number of latent steps in the LaST SFT model, evaluations were con-

ducted using configurations of 2, 4, 6, 16, and 64 latent steps, despite the model having been trained with a maximum of 6 latent steps (Figure 3). The results show that increasing the number of latent steps does not improve performance. Although the model stays within distribution up to 16 latent steps, its performance declines sharply at 64 latent steps.

GRPO training does not improve the performance of LaST SFT. Using the GRPO algorithm, which ignores the latent steps, does not yield performance gains for the LaST model, in contrast to results for the Qwen SFT & RL model. Two main factors may explain this outcome:

First, the GRPO algorithm is designed for language-space reasoning models, where it increases the probability of generating intermediate language tokens that eventually lead to correct answers. In the LaST SFT model, however, no such intermediate language tokens are produced before the final answer, which prevents the learning signal from being effective. To address this limitation, we introduced Latent RL in Section 3.3. Second, GRPO typically enhances performance by encouraging models to generate longer responses before the final answer, thereby increasing inference-time computation. In the LaST SFT model, the number of latent steps is fixed. However, as shown in Figure 3, simply increasing the number of latent steps does not enhance performance. This indicates that the effect of this factor is limited.

Latent RL training is unstable. Even though we address the limitations of other training techniques for latent-space thinking model training, we observe that Latent RL training shows unstable optimization dynamics (Appendix E). This is also reflected in the benchmark evaluations, where performance does not increase despite the Latent RL training. These observations suggest that the unstable Latent RL struggles to improve latent-space thinking. Exemplary model outputs for a question are presented in Appendix F.

5.2 Coconut SFT Ablations

We conduct ablation studies for the Coconut SFT procedure, as the baseline latent-space thinking model is trained with it.

Necessity of the Coconut SFT Procedure. When latent-space thinking is introduced to the Qwen Base model without any additional training, the model produces out-of-distribution and nonsen-

sical outputs (Figure 4). This occurs because the model is fed its own latent steps as input—data it has never encountered during pre-training. Therefore, an SFT procedure is essential before applying any RL training, as RL relies on having at least some correct responses. To this end, we employ the Coconut SFT method (Hao et al., 2025).

```

Qwen Base with Latent-Steps
<|start-latent|><|latent|><|end-latent|>
[\\(\\)\$);,;)\{degrees\\} ( (); shi zi equ zi
. hua\\}<|end-latent|><|end-latent|>
.all\\_()\{\\}

```

Figure 4: Qwen Base response to the question in Figure 10 after introducing latent-steps.

Language Removal Approach. Instead of a step-wise removal of pre-defined reasoning steps (as in Coconut SFT), we investigate a token-wise removal approach, where a fixed number of tokens is removed. This relaxes dataset constraints, enabling the use of any question-response dataset for latent-space thinking model training beyond GSM8K-Aug-NL. For this investigation, all reasoning steps in GSM8K-Aug-NL answers are concatenated. During LaST SFT training, we remove 12 tokens per latent step—half the average step length—to match the step-wise setup.

Removal Technique	GSM8K pass@1	MATH500 pass@1
Step-wise	17.9	11.3
Token-wise	7.5	3.4

Table 4: Investigation of different language token removal techniques in Coconut SFT.

Step-wise removal performs better (Table 4), presumably because token-wise removal can truncate reasoning steps mid-thought, making it more difficult for the model to internalize the reasoning process.

Number of Epochs per Stage. The Coconut SFT method uses 3 epochs per stage, with each stage involving one language step removal and two latent space insertions. We explore different numbers of epochs per stage:

Training the LaST model with one epoch per stage performs better than with two epochs (Table 5). In comparison, the Coconut method (Hao et al., 2025) reported using 3 epochs per stage. Fur-

Epochs/Stage	GSM8K pass@1	MATH500 pass@1
1	17.9	11.3
2	17.0	10.4

Table 5: Ablation study on num. of epochs/stage.

ther experiments were omitted due to the slow training of Coconut SFT.

Final Stage Training Epochs Effect. We also explore the impact of each epoch on the final stage of Coconut SFT procedure, showing that this last stage enhances the performance of the LaST SFT model on the GSM8K benchmark (Figure 5). However, the observed performance drop on the MATH500 benchmark indicates that extended training on the GSM8K-Aug-NL dataset reduces the generalization.

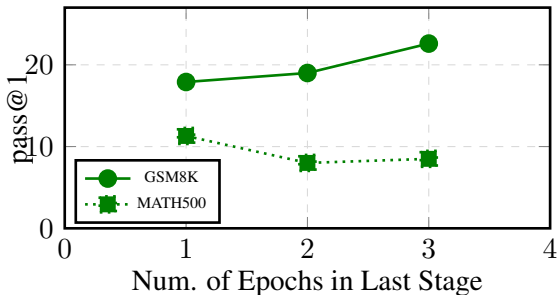


Figure 5: The effect of number of epochs in the last stage of the Coconut SFT procedure.

6 Conclusion

In this work, we explored different training strategies for latent-space thinking models, starting from the Coconut SFT procedure (Hao et al., 2025) and extending it to RL-based methods. While Coconut SFT provides a structured way to train latent-space thinking models, it consistently underperforms compared to language-space models on mathematical benchmarks. We showed, that its success depends heavily on design choices, such as the number of epochs in the final stage and token removal strategies. The method’s main limitation lies in the absence of direct learning signals to latent steps and its reliance on auto-regressive processing, which limits scalability.

RL approaches were then evaluated as potential improvements. Extending GRPO to Coconut-trained latent-space thinking models brought no gains, since GRPO also lacks direct latent-step su-

pervision. To overcome this, we introduced the Latent RL method, which provides explicit learning signals to latent steps through a value model. However, training was unstable and did not improve benchmark performance.

Future work should aim to enhance SFT training efficiency, stabilize Latent RL, and develop mechanisms for adaptive control over latent steps. Overall, the findings suggest that current latent-space thinking training methods still lag behind language-space models, primarily due to their instability or the absence of supervision for latent-space thinking.

Limitations

This study examines training techniques for latent-space thinking models in the mathematical domain. While this focus allows for precise evaluation, it limits generalizability to other tasks such as general knowledge, logical reasoning, or programming. Prior work (Hao et al., 2025; Zhu et al., 2025a) suggests that latent-space thinking may perform differently in those domains, highlighting a key direction for future research.

Another limitation is the use of a single base model, which ensures consistency but narrows the scope of conclusions. Models with different sizes, architectures, or pretraining strategies may respond differently to these training methods, and exploring such variations could offer deeper insight.

A further challenge is efficiency: unlike standard LMs, latent-space thinking models require auto-regressive generation during both training and inference, increasing computational cost and limiting scalability. This makes large-scale training impractical with current setups.

Finally, latent-space thinking reduces model interpretability, making it harder to analyze internal processes or ensure transparency in deployment. Future work should therefore focus on improving interpretability—such as by studying attention or representation changes across latent steps—to better understand and control latent-space thinking behavior.

Acknowledgments

We gratefully acknowledge the Leibniz Supercomputing Centre (LRZ) of the Bavarian Academy of Sciences and Humanities and the Munich Center for Machine Learning (MCML) for generously providing computational resources. Matthias

Aßenmacher received funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the National Research Data Infrastructure – NFDI 27/1 - 460037581 - BERD@NFDI.

References

- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Quy-Anh Dang and Chris Ngo. 2025. Reinforcement learning for reasoning in small llms: What works and what doesn't. *arXiv preprint arXiv:2503.16219*.
- Yuntian Deng, Yejin Choi, and Stuart Shieber. 2024. From explicit cot to implicit cot: Learning to internalize cot step by step. *arXiv preprint arXiv:2405.14838*.
- Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. 2023. Implicit chain of thought reasoning via knowledge distillation. *arXiv preprint arXiv:2311.01460*.
- Evelina Fedorenko, Steven T Piantadosi, and Edward AF Gibson. 2024. Language is primarily a tool for communication rather than thought. *Nature*, 630(8017):575–586.
- Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. 2023. Think before you speak: Training language models with pause tokens. *arXiv preprint arXiv:2310.02226*.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shitong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E Weston, and Yuandong Tian. 2025. Training large language models to reason in a continuous latent space. In *Second Conference on Language Modeling*.
- Hugging Face. 2025. Open r1: A fully open reproduction of deepseek-r1.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PmlR.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Jacob Pfau, William Merrill, and Samuel R Bowman. 2024. Let's think dot by dot: Hidden computation in transformer language models. *arXiv preprint arXiv:2404.15758*.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseek-math: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shrutu Bhosale, and 1 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Galouédec. 2020. Trl: Transformer reinforcement learning. <https://github.com/huggingface/trl>.

- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, and 1 others. 2025. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*.
- Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. 2025a. Reasoning by superposition: A theoretical perspective on chain of continuous thought. *arXiv preprint arXiv:2505.12514*.
- Rui-Jie Zhu, Tianhao Peng, Tianhao Cheng, Xingwei Qu, Jinfa Huang, Dawei Zhu, Hao Wang, Kaiwen Xue, Xuanliang Zhang, Yong Shan, and 1 others. 2025b. A survey on latent reasoning. *arXiv preprint arXiv:2507.06203*.
- Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Reward Functions

For the GRPO method, accuracy and format rewards are used; while the accuracy reward is the same, the format reward differs as indicated below. The Latent RL method uses the same reward functions as GRPO for the latent-space thinking model.

Accuracy Reward

Reward	Condition
$r = 1$	The correct answer appears in <code>\boxed{...}</code> inside <code><answer>...</answer></code> tags.
$r = 0$	Otherwise.

The correctness is verified by exact string match or mathematical equivalence similar to [Hugging Face \(2025\)](#).

Format Reward — Language-Space Thinking Models

Reward	Format
$r = 1$	<code><think>...</think></code> <code><answer> \boxed{...} </answer></code>
$r = 0.5$	Same format but the <code>\boxed{...}</code> is missing.
$r = 0$	Otherwise.

Format Reward — Latent-Space Thinking Models

Reward	Format
$r = 1$	<code>< start-latent >...< end-latent ></code> <code><answer> \boxed{...} </answer></code>
$r = 0.5$	Same format but the <code>\boxed{...}</code> part is missing.
$r = -1$	Any of <code>< start-latent ></code> , <code>< end-latent ></code> , <code><think></code> appear after the first latent block.
$r = 0$	Otherwise.

B Latent RL Methodology

Latent RL method has two stages:

1. Training a value model $\rho_{\theta+\phi}$ by freezing parameters θ .
2. Training the policy model π_{θ} by leveraging the predictions of the value model for latent steps.

In the first stage, data is collected to train $\rho_{\theta+\phi}$. A policy model with latent space thinking π_{ϕ} is used to generate responses to questions. For each response, an accuracy reward is computed using

the reward function described in Section A. From these responses, the final-layer embeddings \mathbf{h} are extracted and paired with the corresponding accuracy reward r , forming training pairs (\mathbf{h}_i, r_i) .

The value model $\rho_{\theta+\phi}$ maps the embedding \mathbf{h} to an estimated reward $\rho_{\theta+\phi}(\mathbf{h})$. To train it, only the value head parameters ϕ are optimized by minimizing the binary cross-entropy (BCE) loss L_{BCE} over embedding–reward pairs (\mathbf{h}_i, r_i) . Crucially, the loss is computed exclusively on embeddings corresponding to latent steps, while embeddings from other parts of the sequence—such as the question tokens or the final language answer—are ignored. This ensures that the value model learns to evaluate the quality of latent-space thinking steps.

$$\arg \min_{\phi} L_{BCE}(\rho_{\theta+\phi}) \quad (9)$$

$$L_{BCE}(\rho_{\theta+\phi}) = r_i \log(\rho_{\theta+\phi}(\mathbf{h}_i)) + (1 - r_i) \log(1 - \rho_{\theta+\phi}(\mathbf{h}_i)) \quad (10)$$

In the second stage, the policy model π_{θ} is provided with a question and generates a sequence consisting of latent steps followed by a final language answer. This complete sequence—comprising the given question, the latent steps, and the generated answer—is then passed to the value model $\rho_{\theta+\phi}$, ensuring that it receives the same input as the policy model. Although the value model outputs estimated rewards for all positions in the sequence, the training objective is applied only to the latent step positions. The goal is to maximize the estimated reward of the latent steps, which is equivalent to minimizing L_{LAT} in Equation (7).

When optimizing Equation (7), the value head parameters ϕ are frozen. This is necessary because if ϕ were also updated, the value head could trivially learn to always predict high rewards, thereby collapsing the training signal. Instead, only the shared parameters θ are updated, which means that optimizing the value model $\rho_{\theta+\phi}$ also updates the policy model π_{θ} , since the two models share θ .

After this update step, the value head parameters ϕ are refined separately using the BCE objective in Equation (9), with the true accuracy reward from that step as the target. This ensures that the value head remains aligned with the updated policy and continues to provide reliable reward estimates.

C Training Hyperparameters

The training set ups are presented in Table 6 and 7. All training runs use AdamW optimizer (Loshchilov and Hutter, 2017).

D Instruction Following

The ratio of generated responses that are fitting to the instructed format is given in Figure 6.

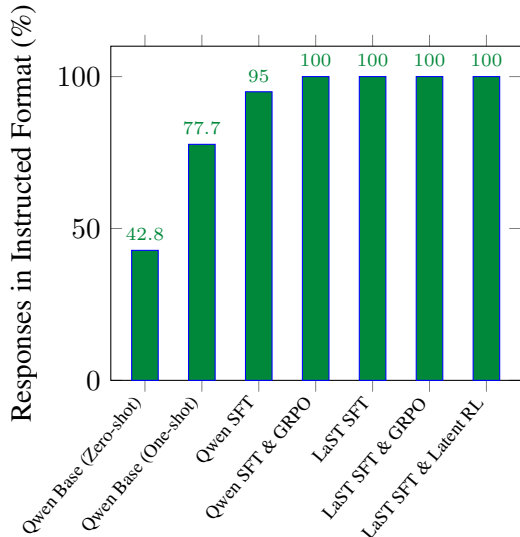


Figure 6: Ratio of answers following the instructed format during evaluation of models in GSM8K and MATH500 benchmarks. All models evaluated in zero-shot setting unless otherwise indicated.

E Latent RL Experiments

The training curves in Latent RL training procedure for the LaST SFT & Latent RL model in in Table 3 are presented in this section.

As described in Section 3.3, in the first phase we train a value model (Figure 7). Although the model achieves a high ROC-AUC score, its F1 score remains comparatively lower. This discrepancy arises because ROC-AUC measures the overall ranking ability of the model across thresholds, whereas the F1 score reflects the balance of precision and recall at a fixed threshold. Since the second phase of Latent RL training relies on the predicted probabilities rather than thresholded labels, ROC-AUC is a more informative indicator of value head performance in this setting.

After training the value model, in the second stage both the policy model and the value models are trained sequentially. However, we see that the

both policy loss and value head loss are fluctuating, indicating an unstable training (Figure 8).

To stabilize training, we freeze the value head during the second phase of Latent RL. This prevents it from becoming a moving target and leads to smoother optimization for the policy model, as shown in Figure 9. Although the policy loss decreases steadily, overall accuracy does not improve.

This stagnation arises from two causes: misalignment between the frozen value head and the evolving policy, and reward hacking. As the policy changes, the static value head becomes less accurate, while the policy exploits its fixed reward signal instead of improving reasoning or task accuracy. Thus, smoother training does not yield better benchmark performance.

F Example Responses from MATH500 Benchmark

Example answers from the models reported in Table 3 are shown in Figure 11 to Figure 16 for a representative question from the MATH500 benchmark, illustrating their behavior.

For this MATH500 example, the language-space models show variation in their responses, whereas all latent-space thinking models produce identical final answer. Since the latent steps are continuous, changes in their behavior cannot easily be inferred without additional interpretability studies.

Hyperparameter	Qwen SFT	LaST SFT
Dataset	GSM8K-Aug-NL	GSM8K-Aug-NL
Num Examples	300K	300K
Base Model	Qwen Base	Qwen Base
Hardware	4 × Nvidia A100-80GB	8 × Nvidia A100-40GB
Training Time	~ 1 hour	~ 8 hours
Num Epochs	1	5
Per Device Batch Size	4	1
Gradient Accum. Steps	32	128
Effective Batch Size	512	512
Learning Rate	5e-5	5e-5
LR Scheduler	StepLR	StepLR
LR Scheduler Args	$\gamma = 0.8$, step size=50	$\gamma = 0.8$, step size=50
Warmup Steps	–	–
Weight Decay	0.01	0.01

Table 6: Training hyperparameters for Qwen SFT and LaST SFT models. Note that the optimizer and scheduler reset after each stage in LaST SFT training.

Hyperparameter	Qwen SFT & GRPO	LaST SFT & GRPO	LaST SFT & Latent RL
Dataset	OpenR1-Math-220k	OpenR1-Math-220k	OpenR1-Math-220k
Num Examples	10K	10K	10K
Base Model	Qwen SFT	LaST SFT	LaST SFT
Hardware	4 × Nvidia A100-80GB	4 × Nvidia A100-80GB	1 × Nvidia A100-40GB
Training Time	~ 4 hours	~ 1 hour	~ 1.5 hours
Num Epochs	1	1	1
Batch Size / Device	2	4	2
Grad. Accum. Steps	32	16	128
Effective Batch Size	256	256	256
Learning Rate	5e-6	5e-6	5e-6
LR Scheduler	Cosine With Min LR	Cosine With Min LR	Cosine With Min LR
LR Scheduler Args	min lr rate = 0.1	min lr rate = 0.1	min lr rate = 0.1
Warmup Steps	10	10	10
Weight Decay	0.01	0.01	0.01
Max Completion Len.	1024	1024	1024
Temperature	0.7	0.7	0
Num Generations	8	8	1
KL Div. Coeff. β	0.0	0.64	–

Table 7: Training hyperparameters for Qwen SFT & GRPO and LaST SFT & GRPO models. Note that the effective batch size contains 256 responses to $256/8 = 32$ different questions and 8 responses for each questions in GRPO training.



Figure 7: Training curves for the value model training in the first stage of Latent RL. Note that only the linear value head is trained. (a) BCE loss. (b) F1 and Roc Auc scores.

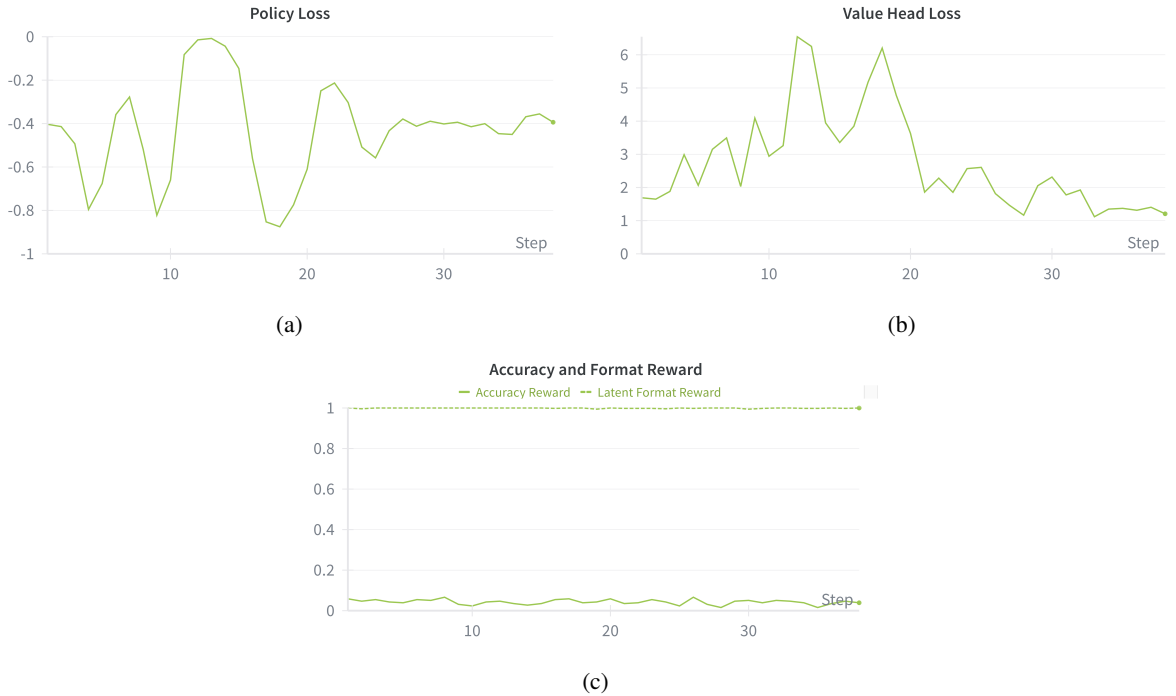
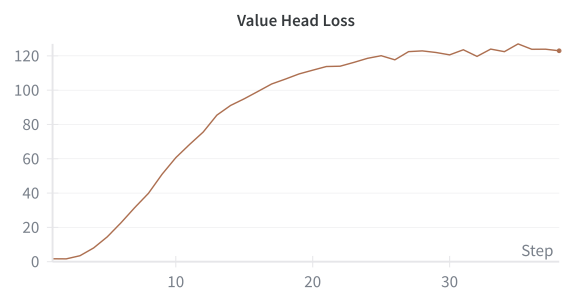


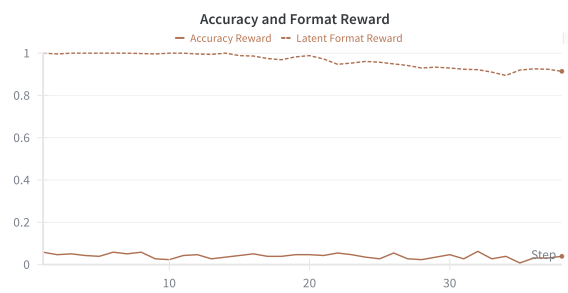
Figure 8: Training details of Latent RL: (a) policy loss, (b) value head BCE loss, (c) accuracy and latent format rewards.



(a)



(b)



(c)

Figure 9: Training dynamics of Latent RL with a frozen value model. (a) Policy loss shows smooth convergence. (b) Value model loss, monitored but not optimized. (c) Accuracy and latent format rewards.

Question

How many positive whole-number divisors does 196 have?

Figure 10: Example question from MATH500 benchmark. The question, inserted into the system prompt which is omitted here for brevity. The correct answer is 9.

Qwen Base Response

```
<think>196 is a perfect square, so it has an even number of divisors. The prime factorization of 196 is  $2^2 * 7^2$ . The number of divisors is  $(2+1)(2+1) = 9$ . So, 196 has 9 positive whole-number divisors.</think><answer>\boxed{9}</answer>
```

Figure 11: Example Qwen Base (zero-shot) response to the question in Figure 10.

Qwen SFT Response

```
<think>First prime factorize  $196=2^2*7^2$ . The prime factorization of any divisor of 196 can only include the primes 2 and 7. We are free to choose either 0, 1, 2, or 3 as the exponent of 2 (4 options). For the exponent of 7, we can choose 0 or 1 (2 options). In total, there are  $4 \times 2=8$  possibilities.</think><answer>\boxed{8}</answer>
```

Figure 12: Example Qwen SFT response to the question in Figure 10.

Qwen SFT & GRPO Response

```
<think>The problem requires us to find the number of positive whole-number divisors of 196. First, we need to determine the prime factorization of 196. We start by dividing 196 by the smallest prime number, which is 2:
```

```
\[196 \div 2 = 98\]
```

Next, we divide 98 by 2:

```
\[98 \div 2 = 49\]
```

Now, 49 is not divisible by 2, so we check the next smallest prime number, which is 7:

```
\[49 \div 7 = 7\]
```

And again, 7 is a prime number. Therefore, the prime factorization of 196 is:

```
\[196 = 2^2 \times 7^2\]
```

To find the number of positive whole-number divisors, we use the formula that if a number has a prime factorization of the form $(p_1^{e_1} \times p_2^{e_2} \times \dots \times p_k^{e_k})$, then the number of divisors is given by $((e_1 + 1)(e_2 + 1) \dots (e_k + 1))$. For the factorization $(2^2 \times 7^2)$, the exponents are 2 and 2. Adding 1 to each of these exponents gives 3 and 3. Therefore, the number of divisors is:

```
\[(2+1)(2+1) = 3 \times 3 = 9\]
```

```
Thus, the number of positive whole-number divisors of 196 is \(\boxed{9}\).</think><answer>\boxed{9}</answer>
```

Figure 13: Example Qwen SFT & GRPO response to the question in Figure 10.

LaST SFT Response

```
<|start-latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|end-latent|><answer>\boxed{14}</answer>
```

Figure 14: Example LaST SFT response to the question in Figure 10.

LaST SFT & GRPO Response

```
<|start-latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|end-latent|><answer>\boxed{14}</answer>
```

Figure 15: Example LaST SFT & GRPO response to the question in Figure 10.

LaST SFT & Latent RL Response

```
<|start-latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|latent|><|end-latent|><answer>\boxed{14}</answer>
```

Figure 16: Example LaST SFT & Latent RL response to the question in Figure 10.

Enhancing Retrieval via Cognitively Motivated Document Expansion

Giacomo Loss¹ and Andreas Stephan² and Matthias Aßenmacher^{1,3}

¹Department of Statistics, LMU Munich,

²LDA - Legal Data Analytics GmbH, Munich, Germany,

³Munich Center for Machine Learning (MCML), LMU Munich

Correspondence: matthias@stat.uni-muenchen.de

Abstract

This study examines the potential of leveraging large language model (LLM) capabilities to enhance performance in document retrieval tasks. Using human-written prompts based on the 5E Instructional Model from educational psychology, we generate alternative versions of documents in a given corpus using an LLM, tapping into its vast knowledge base. These generated texts can then be used in retrieval tasks, complementing or replacing the original corpus before applying fusion algorithms to combine the results. While the generated texts individually do not outperform the original corpus, fusing retrieval results from multiple generated corpora with those of the original corpus often leads to performance improvements. This suggests that LLM-generated documents, while not a substitute for the original, can complement it to enhance retrieval performance.

1 Introduction

Information is abundant and distributed across various heterogeneous sources, including web pages, digital libraries, and forums. Yet, even when an answer exists, it is often difficult to locate the most relevant information. This is the core problem of Information Retrieval (IR): given a user query and a large document collection, retrieve the most relevant documents. Early IR methods relied on exact term matching, which suffers from the *Vocabulary Mismatch Problem* (Furnas et al., 1987), as they ignore semantic similarity between queries and documents that use different wording.

Recent advances in deep learning have transformed IR and NLP. Word embeddings (Mikolov et al., 2013) and Transformer-based contextual representations (Vaswani et al., 2017) help mitigate vocabulary mismatch by capturing semantic and contextual information. Building on these advances, large language models (LLMs) trained on massive, heterogeneous corpora have proven effective in

many tasks, including generation, translation, and dialogue. In this work, we investigate how the knowledge retained by modern LLMs can be leveraged for IR. Specifically, we study their use for *document expansion*—a technique that, alongside the widely adopted *query expansion*, enriches the term space to better match user information needs and improve retrieval effectiveness.

Contributions. Our study has two main goals. First, inspired by the 5E Instructional Framework, we design a set of targeted, cognitively motivated prompts and utilize them with an LLM to generate alternative texts for each document in a corpus, which can also be used for retrieval (cf. Fig. 1). Here, we focus on expanding document corpora in a query-independent manner, leveraging the extensive retained knowledge and generalisation capabilities of LLMs. We thereby contribute to broadening the body of research, as this line of research has received comparatively less attention than, for example, query expansion. Second, we investigate whether combining retrieval results from different generated corpora can enhance retrieval performance. We find that, although individual expansions rarely outperform the original corpus alone, combining retrieval outcomes from multiple generated corpora with those from the original corpus consistently improves effectiveness. This suggests that LLMs can generate complementary views of a corpus, thereby enhancing retrieval performance. Our code is publicly available on GitHub.¹

2 Background & Related Work

We integrate concepts from educational science (the 5E Instructional Model) with IR techniques (document expansion), leveraging LLMs' broad

¹<https://github.com/GiacoL/Enhancing-Retrieval-via-Cognitively-Motivated-Document-Expansion>

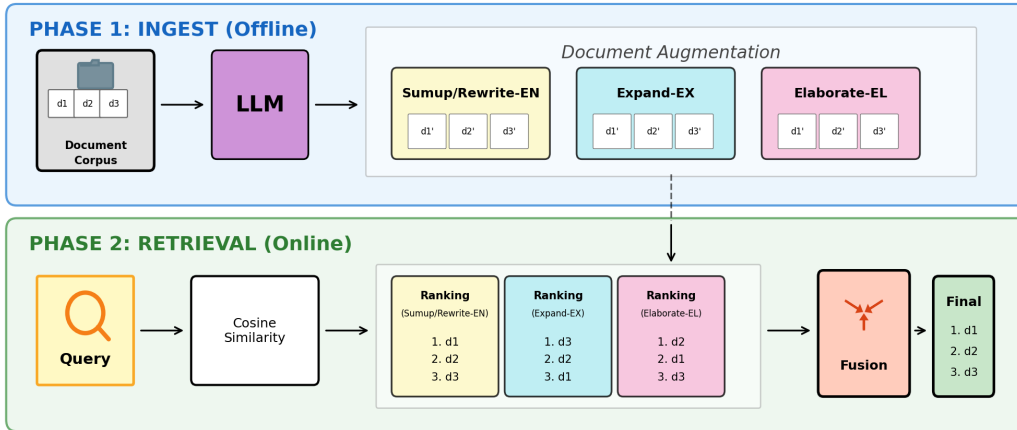


Figure 1: During the **ingestion** phase, each document in the corpus is passed to an LLM using prompts designed to summarize, expand, or elaborate the text. Each prompt yields an alternative version of the corpus, resulting in multiple generated corpora. In the **retrieval** phase, a set of queries is run against each generated corpus, producing a ranking for every (query, corpus variant) pair. Finally, these rankings are fused to obtain a single final document ranking for each query.

knowledge to connect these perspectives. This synthesis provides a coherent structure for organizing and positioning the related work.

2.1 5E Instructional Model

The 5E instructional model (Bybee, 1990), developed by the Biological Sciences Curriculum Study (BSCS), is a pedagogical framework designed to support effective instruction in science education (Duran and Duran, 2004; Joswick and Hulings, 2024). It belongs to the family of constructivist approaches, which posit that learning integrates prior knowledge with hands-on experience. The 5E model comprises five phases: (i) *Engage*: students are presented with new evidence from which a problem emerges and are encouraged to reconcile it with *prior* knowledge; (ii) *Explore*: students *actively* use retained knowledge to make observations and propose solutions to the problems posed by the new evidence; (iii) *Explain*: the teacher formalizes the new concepts in a way that can be assimilated; (iv) *Elaborate*: students apply the new ideas in different contexts and scenarios; (v) *Evaluate*: the focus is on how well students *transfer* concepts to new contexts. As noted by Ruiz-Martín and Bybee (2022), the 5E model rests on recognizable cognitive principles such as the activation of prior knowledge in the *Engage* phase and its use in the following phase, the *Explore* phase, to find solutions to the problems posed by the new evidence; this aligns with our goal of expanding or

shortening corpus documents with an LLM, using a cognitively motivated prompt scheme based on the 5E model, with the LLM acting as the *student*.

2.2 Query & Document Expansion Strategies

Given a large text corpus and a set of queries, the goal of IR is to retrieve the most relevant documents for each query. The process is typically decomposed into two stages. First, in the *retrieval* stage, the relevance of each document to the query is estimated to select the most promising candidates. Second, in the *reranking* stage, these candidates are reordered to refine their relevance-based ranking. This study focuses on the *retrieval* stage: given a set of queries and a text corpus, we aim to retrieve, for each query, the relevant documents.

Early retrieval algorithms were sparse, term-based methods such as *BM25* (Robertson et al., 1995) suffered from the *Vocabulary Mismatch Problem*. Especially after the advent of Transformers (Vaswani et al., 2017), dense retrievers entered the IR landscape (Xiong et al., 2021; Karpukhin et al., 2020). Dense representations of words and sentences enabled a shift from a purely term-based to a more semantic-based setting. However, these dense retrieval systems are often supervised and thus require large amounts of training data, which can be costly to obtain in many scenarios.

A common challenge in IR is that queries are typically short, whereas documents are relatively lengthy and may be written in ways that deviate

substantially from the user query. To mitigate this problem, various *expansion* techniques have been proposed. On the one hand, *query expansion* methods augment queries with external resources or synonymous terms, as in Voorhees (1994). On the other hand, document expansion has also been explored. Tao et al. (2006) or Efron et al. (2012) expand short documents by submitting them as pseudo-queries to a search engine and appending retrieved content; Nogueira et al. (2019b) expand instead documents with queries generated from the documents themselves. Another line of research has focused not on expanding the documents themselves, but on enriching their embedding representations. For example, Zhang et al. (2022) generate multi-view embeddings that can align with different queries, while ColBERT (Khattab and Zaharia, 2020) adopts an interaction-based paradigm to model fine-grained relationships between query and document terms.

2.3 LLMs in IR

A recent line of work has employed LLMs for IR tasks. As documented in the extended survey by Zhu et al. (2023), there are two major approaches to using LLMs for retrieval. On the one hand, LLMs and their extensive retained knowledge can be exploited to reformulate documents or queries to better align with the user’s intent. On the other hand, they can be used for data augmentation, for example, by acting as a substitute for human annotators to provide pseudo-relevance feedback. According to Zhu et al. (2023), these models can thus enhance existing retrieval methods and enable new developments via *in-context learning*, which improves retrieval performance by leveraging only a few examples.

So far, most attention has been devoted to query expansion or rewriting with LLMs: Wang et al. (2023) generate pseudo-documents with an LLM and concatenate them to the original query, while Jagerman et al. (2023) expand queries by exploring different zero-shot, few-shot, and chain-of-thought prompting strategies. More recently, Xia et al. (2025) combine LLMs with knowledge graphs to jointly model semantic and structural relations between documents and queries.

By contrast, *document expansion* has received comparatively less attention, likely due to the substantial computational cost of expanding large document collections. Nogueira et al. (2019b) and Nogueira et al. (2019a) use LLMs to predict queries

relevant to a document, which are then appended to the document to create an expanded corpus. Gao et al. (2023) generate hypothetical documents with an LLM given a query and use their vector encodings to retrieve relevant documents from the corpus. Yu et al. (2022) generate contextual documents for a given query, whereas Bonifacio et al. (2022) generate queries from documents to construct query–document training pairs for dense retrievers. Jeong et al. (2021) expand the text corpus with additional sentences generated by a pretrained LLM, in a manner similar to our approach.

Although expansion appears beneficial for retrieval, Weller et al. (2024) show in a comparative analysis of different LLM-based expansion techniques that such expansion tends to yield gains for weaker (retrieval) models, while it can degrade performance for stronger models.

2.4 Fusion Methods

We employ multiple prompt-generated variants of the same document collection for IR and fuse the resulting rankings into a single summarized result. This fusion task falls under *internal metasearch* (also known as *data fusion*), which combines outputs from multiple retrieval systems that share the same document set. As summarized by Montague and Aslam (2002), different retrieval systems perform differently on the same task, and fusing their outputs yields more consistent results.

There exists a multitude of fusion algorithms, both supervised and unsupervised, which differ along several dimensions—most importantly, whether they rely on similarity scores or ranks to form the fused list. Fox and Shaw (1994) propose several score-aggregation methods; Zhang et al. (2002) introduce a method based on reciprocal ranks; and Mourão et al. (2014) extended this idea by considering inverse squared ranks. By contrast, other authors adopt probabilistic approaches that estimate the likelihood of a system returning a relevant document for a given query. In *ProbFuse* (Lillis et al., 2006), estimated probabilities depend on the documents retrieved during training and their positions in the ranking. Other probabilistic methods include *Bayes-Fuse* (Aslam and Montague, 2001), which estimates document relevance by summing, across systems, the log-ratio of the probability that the document at a given rank is relevant versus non-relevant, and *SegFuse* (Shokouhi, 2007), which—similar to *ProbFuse*—partitions rank lists into segments that grow larger down the ranking.

Key	Prompt
EN.v1	Can you summarize the text keeping the most relevant information and just give me the output without your comments:
EN.v2	Can you rewrite the text with other more conspicuous vocabulary and just give me the output without your comments:
EX.v1	Can you expand the text with your knowledge keeping, expanding, and explaining the most relevant information and just give me the output without your comments:
EX.v2	Can you reduce and summarize the text integrating some of your knowledge, keeping, expanding, and explaining the most relevant information and just give me the output without your comments:
EL.v1	Can you rewrite the text with a more negative stance and just give me the output without your comments:
EL.v2	Can you expand the text, making comments that should help a person to better understand the content of the text:

Table 1: Prompts used to generate texts with LLMs. The *Key* indicates the macrocategory and prompt identifier.

Lillis et al. (2010) investigate modeling relevance probabilities without large training sets by using a system’s *Mean Average Precision* to relate relevance probability to rank. Additional algorithms arise from *Social Choice Theory*, where retrieval systems act as voters. Boehmer et al. (2023) explain how to use positional scoring rules to transform voters’ rankings into scores, while Montague and Aslam (2001) and Montague and Aslam (2002) describe methods based on the *Borda Count Principle* and the *Condorcet* method.

3 Methodology

Our method is based on multiple text expansions via LLMs. In the first step, an LLM is fed each of the n_D documents in the corpus D ; for each document, a set of variants is generated using a predefined list of n_P prompts based on the 5E Instructional Model. This yields $n_P + 1$ versions per document (including the original), i.e., $n_D(n_P + 1)$ documents in total. We then compute embeddings for documents (original and generated) and queries to measure similarity using *cosine similarity*, a widely used metric. For each query q and each set of generated documents, we build a retrieved set of relevant documents. These retrieved sets are then fused using one or more fusion methods, and the final results are evaluated using $nDCG@10$.

3.1 Document Expansion via the 5E Model

Although LLMs are often perceived as human-like, they lack genuine human capabilities. Therefore, to employ the 5E model in this study, certain adaptations were necessary. The five phases of the original scheme were used as guidelines for designing prompts to generate alternative text corpora. We

make two key modifications: (i) Excluding the *Evaluation* phase, as our goal is not to teach a model, although one could imagine future work where an LLM evaluates and refines its own outputs; and (ii) Merging the *Engage* and *Explore* phases. This leaves three prompting dimensions:

Engage and Explore (EN) The LLM reformulates a given text by altering vocabulary or summarizing it, aiming to mitigate the *Vocabulary Mismatch Problem* while preserving the content.

Explain (EX) The LLM explains the text using its retained knowledge, enriching the content and aligning it more closely with human understanding.

Elaborate (EL) The LLM applies the concepts of a given text to other contexts (e.g., analogies, examples, or tone shifts). Such reformulations may improve alignment between texts and queries.

The exact prompt design (cf. Table 1) inevitably involves some arbitrariness, which we recognize as a key limitation of our work. In preliminary experiments (not reported here), we pre-tested a larger set of prompts based on the 5E Model on a subset of the BEIR dataset. We selected the best performing ones for the main analysis across all NanoBEIR datasets in order to keep computational costs manageable.

4 Experiments

We investigate whether the LLMs’ capabilities can be leveraged to summarize or expand a given text, and whether such generated texts can serve as effective substitutes for or complements to the original text in IR tasks. In particular, we aim to combine rankings from multiple generated corpora and

assess whether their fusion improves retrieval performance. We further explore whether adopting the E5 framework to guide text generation yields additional improvements over the original texts and which strategy is most effective. We examine the following hypotheses:

H1 Generated texts will improve performance in IR tasks compared to original texts, and performance will further increase as the number of generated texts used in fusion grows.

H2 Larger LLMs will generate texts that are better suited for IR tasks than smaller models.

H3 Employing a cognitive framework to guide text generation may provide one way to enhance IR performance compared to using the original texts alone.

4.1 Data

Given the computational challenges associated with generating texts, we opted for the NanoBEIR dataset (Thakur et al., 2022), a compact version of BEIR (Thakur et al., 2021), a widely used benchmark in IR research. The NanoBEIR variant contains 50 queries and up to 10k documents per individual dataset, in contrast to the millions of documents included in the standard BEIR datasets. Both comprise generalist and domain-specific datasets, covering a broad range of topics.

4.2 LLMs & Embedding Models

LLMs For text generation, we employ models from the Qwen family (Yang et al., 2024; Qwen Team, 2024) to analyze the impact of model size on retrieval performance. Specifically, we use the medium-sized, instruction-tuned Qwen2.5 models with 3B, 7B, and 14B parameters, which enables a systematic comparison across different scales.

Embeddings To ensure comparability with state-of-the-art approaches, model selection was guided by the MTEB Leaderboard for retrieval tasks (Muennighoff et al., 2023). We employ the INF-Retriever-v1-1.5B (Infly, Yang et al., 2025) embeddings for the main analyses (§5), gte-Qwen2-1.5B-instruct (Li et al., 2023) for the ablation (§6). While the former is of particular interest due to its results on NanoBEIR, making it especially relevant for evaluating retrieval under resource-constrained conditions, the latter has demonstrated competitive performance across general retrieval benchmarks.

4.3 Fusion Algorithms

In preliminary analyses, we evaluated a range of fusion algorithms commonly discussed in the literature. While retrieval effectiveness was comparable across the best-performing methods², execution times varied substantially, particularly for large document corpora. Since design choices in this study are guided by both retrieval effectiveness and computational efficiency, we opted to employ a single method for all analyses: the *logn inverse squared rank* (Mourão et al., 2014). This algorithm provides a favorable trade-off, achieving strong retrieval performance while maintaining manageable computational costs. For the implementation, we relied on the Ranx library (Bassani and Romelli, 2022). For score normalization, the widely used *min-max normalization* was applied prior to fusion.

5 Main Results

Table 2 shows average nDCG@10 differences (and standard deviations) between the retrieval results based on only the original corpus and three clusters of strategies: *Single prompts* refers to retrieval using only a single corpus of generated texts, while *Prompt Combinations* and *Prompt Combinations + Original* correspond to fused retrieval results based on multiple generated corpora, with or without the original corpus. Using multiple generated corpora yields average improvements over using only the original corpus in five out of thirteen datasets, while using single generated corpora alone outperforms fusion in only one case (SciFact). When results from generated corpora are fused with retrieval results from the original corpus, gains appear most consistently. Overall, generated texts rarely serve as a full replacement for the original corpus, but when combined with it, they can provide clear improvements—particularly for generalist datasets such as ArguAna, open-domain general QA datasets (NQ and HotpotQA), or scientific/medical datasets (SciFact and NFCorpus).

Table 3 reports how performance differences relative to using only the original corpus vary with the *number* of fused retrieval results based on generated corpora. Although no strictly monotonic trend emerges, the largest average gains—both with and without including the original corpus—tend to

²We tested CombMIN, CombMAX, CombMED, CombSUM, CombANZ, CombMNZ (Fox and Shaw, 1994), ISR, log-ISR, logn-ISR (Mourão et al., 2014), RRF (Cormack et al., 2009), MAPFuse (Lillis et al., 2010).

Dataset	arguana	climatefever	dbpedia	fever	fiqa	hopotqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touch2020
Original (Absolute Values)	63.85	42.64	65.95	93.63	66.82	83.41	64.31	38.70	67.91	95.96	47.67	81.72	52.39
Single Prompts	-0.74 (± 2.58)	-6.28 (± 1.73)	-2.07 (± 1.00)	-1.98 (± 2.26)	-5.45 (± 2.66)	-0.70 (± 1.56)	-7.43 (± 3.11)	-0.08 (± 0.75)	-1.27 (± 3.05)	-9.30 (± 2.98)	-2.90 (± 1.29)	0.83 (± 1.61)	-3.78 (± 1.36)
Prompt Combinations	1.38 (± 1.53)	-5.95 (± 1.28)	-1.36 (± 1.05)	-1.60 (± 1.23)	-3.18 (± 1.23)	0.34 (± 0.73)	-5.92 (± 1.83)	0.24 (± 0.58)	0.05 (± 2.32)	-5.62 (± 2.42)	-2.35 (± 1.19)	1.12 (± 1.34)	-2.58 (± 0.67)
Prompt Combinations with Original	1.82 (± 0.95)	-4.31 (± 1.06)	-1.05 (± 0.75)	-0.42 (± 0.47)	-1.35 (± 0.67)	0.81 (± 0.54)	-3.72 (± 1.39)	0.39 (± 0.48)	0.74 (± 1.41)	-2.12 (± 1.44)	-1.49 (± 1.01)	1.09 (± 0.92)	-1.49 (± 0.41)
Mean difference to Original	0.82 (± 1.12)	-5.51 (± 0.86)	-1.49 (± 0.43)	-1.33 (± 0.66)	-3.33 (± 1.68)	0.15 (± 0.63)	-5.69 (± 1.52)	0.19 (± 0.20)	-0.16 (± 0.83)	-5.68 (± 2.93)	-2.25 (± 0.58)	1.02 (± 0.13)	-2.62 (± 0.94)

Table 2: Averaged nDCG@10 differences (\pm std. dev.) across all models and prompts for the different datasets and prompt combinations. **Green** indicates improvements, **red** a decrease, relative to the Original baseline.

nr_fused_texts	arguana	climatefever	dbpedia	fever	fiqa	hopotqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touch2020	Mean
0 (only Original)	63.85	42.64	65.95	93.63	66.82	83.41	64.31	38.70	67.91	95.96	47.67	81.72	52.39	66.54
1	-0.74 (± 2.58)	-6.28 (± 1.73)	-2.07 (± 1.00)	-1.98 (± 2.26)	-5.45 (± 2.66)	-0.70 (± 1.56)	-7.43 (± 3.11)	-0.08 (± 0.75)	-1.27 (± 3.05)	-9.30 (± 2.98)	-2.90 (± 1.29)	0.83 (± 1.61)	-3.78 (± 1.36)	-3.17
2	0.40 (± 1.54)	-5.94 (± 1.23)	-1.48 (± 1.24)	-2.19 (± 1.33)	-4.25 (± 1.10)	-0.03 (± 0.82)	-6.41 (± 2.10)	0.05 (± 0.72)	-0.47 (± 2.68)	-7.55 (± 2.20)	-2.65 (± 1.29)	0.88 (± 1.44)	-2.84 (± 0.68)	-2.50
2+Original	1.43 (± 0.88)	-3.54 (± 0.71)	-0.93 (± 0.89)	-0.37 (± 0.33)	-1.33 (± 0.91)	0.68 (± 0.68)	-3.15 (± 1.39)	0.15 (± 0.54)	0.49 (± 1.30)	-2.43 (± 1.50)	-1.16 (± 0.91)	0.95 (± 1.00)	-1.38 (± 0.54)	-0.81
4	2.11 (± 1.11)	-5.97 (± 1.38)	-1.24 (± 1.08)	-1.26 (± 0.94)	-2.45 (± 0.49)	0.59 (± 0.60)	-5.54 (± 1.52)	0.40 (± 0.41)	0.55 (± 2.13)	-4.37 (± 1.46)	-2.04 (± 1.12)	1.14 (± 1.28)	-2.38 (± 0.72)	-1.57
4+Original	2.07 (± 0.81)	-4.74 (± 0.85)	-1.14 (± 0.75)	-0.43 (± 0.54)	-1.39 (± 0.47)	0.86 (± 0.41)	-3.99 (± 1.06)	0.56 (± 0.40)	1.09 (± 1.42)	-1.75 (± 1.55)	-1.75 (± 1.04)	1.13 (± 0.86)	-1.51 (± 0.29)	-0.85
6	2.14 (± 1.16)	-5.90 (± 1.61)	-1.34 (± 0.32)	-0.82 (± 1.23)	-2.18 (± 0.55)	0.70 (± 0.40)	-5.60 (± 2.17)	0.32 (± 0.60)	0.11 (± 2.12)	-3.59 (± 0.61)	-2.36 (± 1.29)	1.79 (± 1.47)	-2.43 (± 0.30)	-1.47
6+Original	2.25 (± 1.43)	-5.32 (± 1.15)	-1.15 (± 0.40)	-0.54 (± 0.71)	-1.25 (± 0.53)	1.01 (± 0.52)	-4.64 (± 2.02)	0.61 (± 0.29)	0.43 (± 2.01)	-2.34 (± 1.00)	-1.70 (± 1.29)	1.44 (± 1.10)	-1.75 (± 0.24)	-1.00
Mean difference to Original	1.38 (± 1.06)	-5.38 (± 0.89)	-1.34 (± 0.34)	-1.08 (± 0.69)	-2.62 (± 1.51)	0.44 (± 0.56)	-5.25 (± 1.35)	0.29 (± 0.24)	0.13 (± 0.72)	-4.47 (± 2.67)	-2.08 (± 0.56)	1.17 (± 0.32)	-2.30 (± 0.78)	-1.62

Table 3: Averaged nDCG@10 differences (\pm std. dev.) across all models and prompts for the different datasets and numbers of combined prompts, with Infly embeddings. **Green** indicates improvements, **red** a decrease, relative to the Original baseline.

occur when more texts are fused, with the greatest improvements appearing when retrieval results from the original corpus are fused together with those from generated texts. This suggests that increasing the number of fused texts can help the system converge on a shared notion of which documents are relevant, illustrating the *Chorus Effect* in data fusion (Vogt and Cottrell, 1999). Figure 2 (Appendix A) provides a graphical representation of Table 3, showing the average differences and the corresponding standard deviations. For most datasets, the substantial variability in performance suggests that improvements are driven not only by the act of fusing retrieval results from different generated texts, but also by which specific texts are combined. These results provide partial support for

H1, as using generated texts alone generally does not outperform the original corpus. However, when used in conjunction with it, they often enhance retrieval performance.

Table 4 reports the average performance across fusion strategies relative to the original corpus and provides evidence to investigate H2. The results indicate that larger LLMs do not yield consistently greater improvements. As we focus on using LLMs for expansion rather than on retrieval models themselves, this observation is broadly consistent with Weller et al. (2024), who report that expansion can harm the performance of stronger retrieval models. However, gains are observed in datasets previously found to benefit most from fusion strategies—namely, open-domain question answering

LLM	arguana	climatefever	dbpedia	fever	fiqa	hotpotqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touche2020
Original	63.85	42.64	65.95	93.63	66.82	83.41	64.31	38.70	67.91	95.96	47.67	81.72	52.39
Qwen2.5-3B-Instruct	2.21 (± 1.29)	-4.88 (± 1.13)	-1.07 (± 0.95)	-1.63 (± 0.78)	-2.95 (± 2.07)	0.18 (± 0.72)	-5.84 (± 2.36)	0.40 (± 0.55)	-0.14 (± 1.21)	-5.61 (± 3.42)	-0.97 (± 0.91)	0.03 (± 1.06)	-2.05 (± 1.03)
Qwen2.5-7B-Instruct	0.53 (± 2.19)	-6.64 (± 1.61)	-2.06 (± 0.71)	-0.96 (± 2.28)	-3.50 (± 2.54)	0.08 (± 1.60)	-4.56 (± 2.65)	0.23 (± 0.78)	-1.18 (± 2.91)	-4.81 (± 4.11)	-2.71 (± 1.01)	1.23 (± 1.14)	-2.73 (± 1.17)
Qwen2.5-14B-Instruct	-0.06 (± 1.93)	-4.91 (± 1.39)	-1.26 (± 1.10)	-1.31 (± 1.31)	-3.21 (± 2.47)	0.32 (± 1.08)	-6.41 (± 2.62)	-0.03 (± 0.46)	1.01 (± 2.39)	-6.08 (± 3.62)	-2.96 (± 0.88)	1.82 (± 0.99)	-2.90 (± 1.47)
Mean difference to Original	0.89 (± 0.96)	-5.48 (± 0.82)	-1.46 (± 0.43)	-1.30 (± 0.27)	-3.22 (± 0.22)	0.19 (± 0.10)	-5.60 (± 0.77)	0.20 (± 0.18)	-0.10 (± 0.89)	-5.50 (± 0.52)	-2.21 (± 0.89)	1.03 (± 0.74)	-2.56 (± 0.37)

Table 4: Averaged nDCG@10 differences (\pm std. dev.) across all different (numbers of) prompts for the different datasets and for different LLMs from the Qwen2.5 family and Infly embeddings. **Green** indicates improvements, **red** a decrease, relative to the Original baseline.

Strategy	arguana	climatefever	dbpedia	fever	fiqa	hotpotqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touche2020	Sum
Original	0	3	1	2	3	0	3	0	0	2	2	0	3	19
Original+Combination	2	0	1	1	0	1	0	1	1	1	1	0	0	9
EN+Combination+Original	2	0	0	1	0	0	0	1	1	1	0	0	0	6
EX+Combination+Original	1	0	1	1	0	0	0	0	1	0	1	0	0	5
EL+Combination+Original	1	0	0	0	0	1	0	1	0	1	0	0	0	4
EX+Combination	0	0	0	0	0	1	0	1	2	0	0	0	0	4
EN+Combination	1	0	1	0	0	2	0	0	0	0	0	0	0	4
EL+Combination	1	0	1	0	0	0	0	0	2	0	0	0	0	4
EL	0	0	0	0	0	0	0	0	0	0	0	2	0	2
EN	0	0	0	0	0	0	0	0	0	0	0	1	0	1
EX	0	0	0	0	0	0	0	1	0	0	0	0	0	1

Table 5: Frequencies of best-performing combinations across datasets and LLMs for different prompt strategies and their combinations. Infly embeddings are used. Each cell indicates the number of times a given strategy achieved the best performance, with the final column showing totals across datasets. When a column sum exceeds the number of LLMs (3), it indicates that multiple prompt groups were part of the combination yielding the best performance.

(NQ and HotpotQA). This outcome partially supports H2, suggesting that scaling model size can improve performance in open-domain tasks; however, it does not guarantee consistent benefits across all tasks.

Table 5 examines the effect of different prompt types, derived from the 5E cognitive model, on retrieval performance. The table shows how often a prompt (combination) achieved the highest absolute retrieval performance across all LLMs considered. The original corpus outperformed in most cases; nevertheless, combining it with generated texts led to non-negligible improvements in many instances. When comparing the three prompting categories, EN and EX prompts consistently performed better, in combination with or without the original corpus, than EL prompts. EN prompts focus on summarizing the original text, while EX prompts aim to expand it with knowledge retained

by the LLM, and EL prompts seek to re-elaborate the original text (e.g., reformulating it with a more negative stance). These results suggest that summarizing or expanding texts with LLMs can enhance retrieval performance, particularly when fused with the original corpus, whereas altering the text (EL) may not always yield comparable improvements. These findings suggest that employing a structured framework—such as the cognitively motivated one used in this work—can help more effectively leverage LLMs’ retained knowledge, yielding expansions that translate into improvements in retrieval performance. Thus, these observations support H3.

6 Ablations

Given the multitude of factors influencing retrieval performance, we conduct ablations regarding the two most important ones: the LLM for generation and the embedding model for retrieval.

Dataset	arguana	climatefever	dbpedia	fever	fiqa	hopotsqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touch2020
Original	63.85	42.64	65.95	93.63	66.82	83.41	64.31	38.70	67.91	95.96	47.67	81.72	52.39
Single Prompts (7B)	-1.61 (± 2.95)	-7.45 (± 1.56)	-2.70 (± 0.46)	-2.12 (± 3.77)	-6.33 (± 1.94)	-0.93 (± 2.57)	-6.74 (± 3.35)	-0.34 (± 0.96)	-2.87 (± 4.15)	-8.67 (± 4.18)	-3.47 (± 1.05)	1.12 (± 1.34)	-4.06 (± 0.80)
Single Prompts (14B)	-1.76 (± 2.41)	-5.68 (± 2.09)	-1.72 (± 1.28)	-2.06 (± 1.63)	-5.64 (± 3.04)	-0.78 (± 0.92)	-8.40 (± 3.21)	-0.18 (± 0.46)	-0.37 (± 2.78)	-9.90 (± 2.17)	-3.62 (± 0.54)	1.45 (± 1.45)	-4.22 (± 1.87)
Prompt Combinations (7B)	1.11 (± 0.87)	-7.44 (± 0.90)	-2.18 (± 0.52)	-0.91 (± 1.51)	-3.40 (± 1.66)	0.32 (± 0.87)	-4.52 (± 1.88)	0.26 (± 0.64)	-1.07 (± 2.58)	-5.07 (± 2.67)	-2.85 (± 0.69)	1.23 (± 1.24)	-2.74 (± 0.67)
Prompt Combinations (14B)	0.22 (± 1.32)	-5.15 (± 0.54)	-1.09 (± 1.18)	-1.64 (± 1.15)	-2.92 (± 1.18)	0.52 (± 0.84)	-6.67 (± 1.38)	0.03 (± 0.54)	1.41 (± 2.55)	-6.37 (± 2.46)	-3.26 (± 0.65)	2.29 (± 0.71)	-3.01 (± 0.47)
Prompt Combinations with Original (7B)	1.80 (± 0.62)	-5.14 (± 1.18)	-1.41 (± 0.48)	-0.01 (± 0.13)	-1.17 (± 0.46)	0.70 (± 0.54)	-2.73 (± 0.88)	0.69 (± 0.44)	0.17 (± 0.86)	-1.24 (± 1.39)	-1.93 (± 0.74)	1.32 (± 1.03)	-1.59 (± 0.28)
Prompt Combinations with Original (14B)	1.13 (± 0.72)	-3.99 (± 0.72)	-1.05 (± 0.89)	-0.34 (± 0.27)	-1.42 (± 0.70)	1.05 (± 0.61)	-4.44 (± 1.66)	0.05 (± 0.42)	1.79 (± 1.54)	-2.53 (± 1.44)	-2.08 (± 0.59)	1.66 (± 0.64)	-1.65 (± 0.30)
Mean difference to Original	0.15 (± 1.51)	-5.81 (± 1.38)	-1.69 (± 0.65)	-1.18 (± 0.90)	-3.48 (± 2.13)	0.15 (± 0.81)	-5.58 (± 2.05)	0.09 (± 0.36)	-0.16 (± 1.71)	-5.63 (± 3.38)	-2.87 (± 0.72)	1.51 (± 0.42)	-2.88 (± 1.13)

Table 6: Averaged nDCG@10 differences (\pm std. dev.) across all prompts for the different datasets and prompt combinations for the two largest of the three LLMs, Qwen2.5-7B-Instruct and Qwen2.5-14B-Instruct, with Infly embeddings. Green indicates improvements, red a decrease, relative to the Original baseline.

Dataset	arguana	climatefever	dbpedia	fever	fiqa	hopotsqa	msmarco	nfcampus	nq	quoraretrieval	scidocs	scifact	touch2020
Original (Absolute Values)	28.96	25.65	39.67	82.91	15.24	70.40	21.99	15.52	16.40	93.47	26.79	57.04	37.95
Single Prompts	-8.14 (± 5.54)	-5.89 (± 2.17)	0.98 (± 2.91)	-10.74 (± 7.03)	16.74 (± 3.54)	-6.91 (± 5.08)	4.05 (± 3.65)	4.99 (± 2.17)	7.80 (± 6.97)	-32.48 (± 9.22)	-8.39 (± 4.77)	4.37 (± 5.01)	-4.54 (± 3.40)
Prompt Combinations	-5.45 (± 3.31)	-4.57 (± 1.63)	4.23 (± 1.54)	-6.12 (± 3.40)	21.22 (± 2.65)	-2.94 (± 2.49)	7.75 (± 2.52)	6.84 (± 1.54)	12.18 (± 6.44)	-20.81 (± 7.56)	-5.83 (± 3.03)	7.37 (± 3.54)	-1.22 (± 1.97)
Prompt Combinations with Original	-1.70 (± 3.18)	-3.38 (± 1.04)	3.95 (± 1.38)	-2.48 (± 1.85)	19.41 (± 2.94)	-0.33 (± 0.92)	6.89 (± 1.96)	5.34 (± 1.43)	9.49 (± 6.15)	-7.90 (± 3.35)	-1.53 (± 2.24)	6.98 (± 3.21)	1.01 (± 1.69)
Mean difference to Original	-5.10 (± 2.64)	-4.61 (± 1.03)	3.05 (± 1.47)	-6.45 (± 3.38)	19.13 (± 1.84)	-3.40 (± 2.70)	6.23 (± 1.58)	5.72 (± 0.80)	9.82 (± 1.80)	-20.39 (± 10.04)	-5.25 (± 2.83)	6.24 (± 1.33)	-1.58 (± 2.28)

Table 7: Averaged nDCG@10 differences (\pm std. dev.) across all prompts for the different datasets and prompt combinations, with semantic similarity computed using gte-Qwen2-1.5B-instruct embeddings. Green indicates improvements, red a decrease, relative to the Original baseline.

LLMs The following tables reproduce Table 2 but only considering generation with the medium-sized LLM and the larger LLM (Qwen2.5-7B-Instruct and Qwen2.5-14B-Instruct, Table 6). As tables show, there is no *consistent* improvement over all datasets when using only a larger LLM. When using combinations of prompts, both models surpass the other on 50% of the datasets; when including the original as well, the *smaller* of the two models performs favorably in 7 out of 13 cases.

Embedding Model Table 7 reproduces table 2 but with gte-Qwen2-1.5B-instruct embeddings. With this embedding model, the baseline nDCG@10 is lower than with the Infly embeddings, and expansion appears to particularly benefit

weaker embedding models. In the main results and in preliminary (unreported) experiments with E5 (Wang et al., 2022), BGE (Xiao et al., 2023), and UAE (Li and Li, 2023) embedding models, fusion that included the original corpus consistently performed best, as one would expect. Interestingly, with this embedding model, fusion over generated corpora without the original corpus slightly outperforms fusion that includes it.

7 Conclusion and Future Work

The goal of this study was to investigate whether LLMs, guided by a set of cognitively motivated prompts, can be used to generate alternative versions of a given text corpus that enhance performance in retrieval tasks. Our results show that while using *single* generated texts underperforms

compared to using the original corpus, retrieval results obtained using combinations of the generated alternative corpora can improve performance. The effect is even more apparent when they are used in combination with the original corpus. Moreover, using prompts that expand texts by leveraging the retained knowledge of LLMs to complement and summarise the corpus (EN and EX prompts) yields greater improvements in retrieval performance than more exotic reformulations of the text (EL prompt), highlighting that LLMs can act as powerful allies in retrieval tasks.

However, this study and its findings raise several questions that should be addressed in future work. First, our hypotheses should be validated on additional datasets, ideally larger than those considered here. Second, alternative prompting schemes could be explored: rather than applying a single set of prompts to a heterogeneous group of datasets, prompting strategies might be tailored to the specific domain of each dataset. Third, although using the 5E Model to guide prompt design yielded relevant results, it represents only one possible approach to developing an evidence-based prompting strategy, rather than the only one. Future work could therefore investigate alternative prompting frameworks. Finally, a natural next step would be to investigate whether these cognitively motivated expansions can also improve the performance of dense retrievers.

Limitations

This study has several limitations. First, we rely on LLMs to generate texts from human-crafted prompts. Although the 5E framework guided prompt design, prompt writing remains inherently subjective, and the results and conclusions reported here may not hold under alternative prompting strategies. Moreover, the same fixed set of prompts is applied to text corpora that differ considerably from one another. Second, limited computational resources prevented us from systematically evaluating larger LLMs or using larger datasets. Our approach is feasible with smaller document corpora but pose significant computational challenges for larger text collections.³ Third, our analysis is ex-

³As an example, in preliminary analyses, using an NVIDIA Tesla V100 GPU with 16GB of memory, for the (full) FiQa dataset, which includes more than 57K documents, over 600 queries, and six prompts for text generation, the total runtime was 44.35 hours. The majority of this time, 30.93 hours (70%), was dedicated to text generation using the six prompts.

ploratory in nature. For example, when comparing performance differences between generated texts and the original corpus, we did not conduct statistical significance tests to assess the robustness of these differences. Finally, when reproducing retrieval results from the MTEB leaderboard (Muennighoff et al., 2023)—without using any LLMs—the results reported in this study do not exactly match the leaderboard scores, although the deviations remain within an acceptable range. Several factors may have contributed to these discrepancies, including variations in the instructions used to supplement the input during embedding computation.

Acknowledgments

Matthias Aßenmacher received funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the National Research Data Infrastructure – NFDI 27/1 - 460037581 - BERD@NFDI. We acknowledge EuroHPC JU for awarding the project ID EHPC-DEV-2024D05-033 access to the Meluxina Computing Services.

References

- Javed A Aslam and Mark Montague. 2001. Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284.
- Elias Bassani and Luca Romelli. 2022. [ranx.fuse: A python library for metasearch](#). In *CIKM*, pages 4808–4812. ACM.
- Niclas Boehmer, Robert Bredereck, and Dominik Peters. 2023. Rank aggregation using scoring rules. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 5515–5523.
- Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. Inpars: Unsupervised dataset generation for information retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2387–2392.
- Rodger W Bybee. 1990. Science for life & living: An elementary school science program from biological sciences curriculum study. *The American Biology Teacher*, 52(2):92–98.
- Computing the embeddings for documents and queries took 1.39 hours (3%) while calculating similarity scores required 11.35 hours (26%). Finally, the fusion and evaluation process took 0.68 hours (2%).

- Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759.
- Lena Ballone Duran and Emilio Duran. 2004. The 5e instructional model: A learning cycle approach for inquiry-based science teaching. *Science Education Review*, 3(2):49–58.
- Miles Efron, Peter Organisciak, and Katrina Fenlon. 2012. Improving retrieval of short texts through document expansion. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 911–920.
- Edward Fox and Joseph Shaw. 1994. Combination of multiple searches. *NIST special publication SP*, pages 243–243.
- George W. Furnas, Thomas K. Landauer, Louis M. Gomez, and Susan T. Dumais. 1987. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971.
- Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777.
- Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query expansion by prompting large language models. *arXiv preprint arXiv:2305.03653*.
- Soyeong Jeong, Jinheon Baek, ChaeHun Park, and Jong C Park. 2021. Unsupervised document expansion for information retrieval with stochastic text generation. *arXiv preprint arXiv:2105.00666*.
- Candace Joswick and Melissa Hulings. 2024. A systematic review of bscs 5e instructional model evidence. *International Journal of Science and Mathematics Education*, 22(1):167–188.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.
- Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.
- Xianming Li and Jing Li. 2023. Angle-optimized text embeddings. *arXiv preprint arXiv:2309.12871*.
- Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.
- David Lillis, Fergus Toolan, Rem Collier, and John Dunnion. 2006. Probfuse: a probabilistic approach to data fusion. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 139–146.
- David Lillis, Lusheng Zhang, Fergus Toolan, Rem W Collier, David Leonard, and John Dunnion. 2010. Estimating probabilities for effective data fusion. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 347–354.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Mark Montague and Javed A Aslam. 2001. Relevance score normalization for metasearch. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 427–433.
- Mark Montague and Javed A Aslam. 2002. Condorcet fusion for improved retrieval. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 538–548.
- André Mourão, Flávio Martins, and Joao Magalhaes. 2014. Inverse square rank fusion for multimodal search. In *2014 12th international workshop on content-based multimedia indexing (CBMI)*, pages 1–6. IEEE.
- Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. **MTEB: Massive text embedding benchmark**. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.
- Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019a. From doc2query to docttttquery. *Online preprint*, 6(2).
- Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.
- Qwen Team. 2024. **Qwen2.5: A party of foundation models**.
- Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.
- Héctor Ruiz-Martín and Rodger W Bybee. 2022. The cognitive principles of learning underlying the 5e model of instruction. *International Journal of STEM Education*, 9(1):21.

- Milad Shokouhi. 2007. Segmentation of search engine results for effective data-fusion. In *European Conference on Information Retrieval*, pages 185–197. Springer.
- Tao Tao, Xuanhui Wang, Qiaozhu Mei, and ChengXiang Zhai. 2006. Language model information retrieval with document expansion. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 407–414.
- Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. **BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models**. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.
- Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2022. Nanobeir: A lightweight subset of the beir benchmark. <https://huggingface.co/datasets/BeIR/nano-beir>. Subset of the BEIR benchmark for efficient evaluation.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Christopher C Vogt and Garrison W Cottrell. 1999. Fusion via a linear combination of scores. *Information retrieval*, 1(3):151–173.
- Ellen M Voorhees. 1994. Query expansion using lexical-semantic relations. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 61–69. Springer.
- Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.
- Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. *arXiv preprint arXiv:2303.07678*.
- Orion Weller, Kyle Lo, David Wadden, Dawn Lawrie, Benjamin Van Durme, Arman Cohan, and Luca Soldaini. 2024. When do generative query and document expansions fail? a comprehensive study across methods, retrievers, and datasets. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1987–2003.
- Yu Xia, Junda Wu, Sungchul Kim, Tong Yu, Ryan A Rossi, Haoliang Wang, and Julian McAuley. 2025. Knowledge-aware query expansion with large language models for textual and relational retrieval. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4275–4286.
- Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. **C-pack: Packaged resources to advance general chinese embedding**. *Preprint, arXiv:2309.07597*.
- Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. **Approximate nearest neighbor negative contrastive learning for dense text retrieval**. In *International Conference on Learning Representations*.
- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.
- Junhan Yang, Jiahe Wan, Yichen Yao, Wei Chu, Yinghui Xu, and Yuan Qi. 2025. **inf-retriever-v1 (revision 5f469d7)**.
- Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2022. Generate rather than retrieve: Large language models are strong context generators. *arXiv preprint arXiv:2209.10063*.
- Min Zhang, Ruihua Song, Chuan Lin, Shaoping Ma, Zhe Jiang, Yijiang Jin, Yiqun Liu, and Le Zhao. 2002. Thu trec 2002: Novelty track experiments. In *TREC*.
- Shunyu Zhang, Yaobo Liang, Ming Gong, Daxin Jiang, and Nan Duan. 2022. Multi-view document representation learning for open-domain dense retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5990–6000.
- Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *ACM Transactions on Information Systems*.

A Performance for different number of prompts

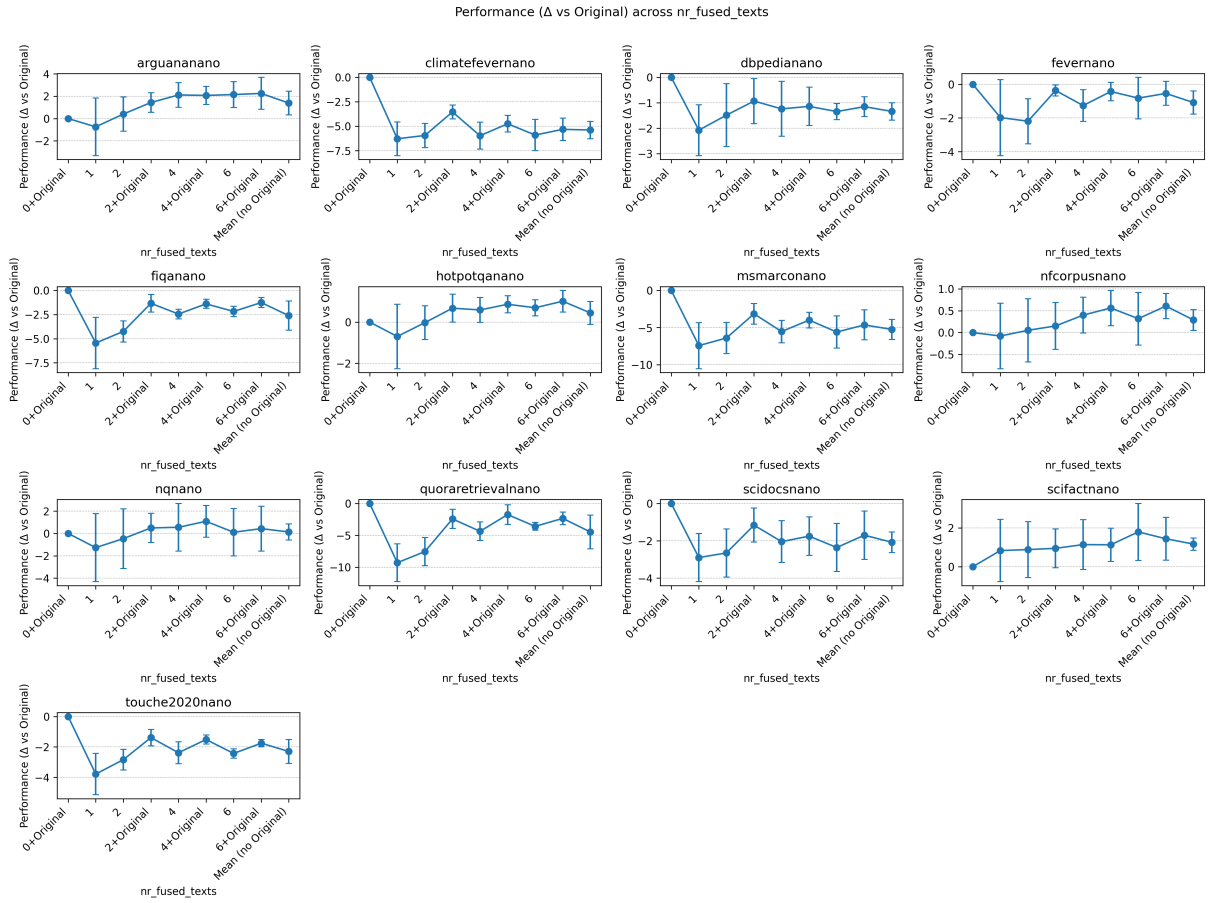


Figure 2: Averaged nDCG@10 differences (\pm std. dev.) across all models and prompts for the different datasets and prompt combinations, with Infly embeddings.

An Efficient Approach for Answering Not Readily Attainable Questions for RAG-based Applications

Zhengdao Chen¹ and Christian Heumann¹ and Matthias Aßenmacher^{1,2}

¹Department of Statistics, LMU Munich,

²Munich Center for Machine Learning (MCML), LMU Munich

Correspondence: matthias@stat.uni-muenchen.de

Abstract

Retrieval-augmented generation (RAG) is an established method for addressing challenges in applying large language models (LLMs), such as ensuring timeliness, incorporating domain-specific expertise, and minimizing hallucinations. However, the effective application of data-augmented LLMs remains challenging due to, e.g., reliance on retriever performance, token-limit restrictions for the input, or the inherent difficulty of global questions directed at large text corpora. Despite various efforts to address these challenges, there are still instances where finding correct answers to certain questions remains elusive. Moreover, as more modules are added to the RAG pipeline, its complexity and latency increase, so that the achieved performance improvements may become less practically significant. Based on these observations, we propose an efficient approach to addressing the issue of not readily attainable questions in a pragmatic way: by collecting questions with incorrectly generated answers, preparing the correct answers offline, and prepending a module for semantic search among the prepared question-answer pairs to the RAG system. If we consider a traditional RAG system an open-book exam, this QA search module can be likened to an open-question exam, similar to a driver’s license test.

1 Introduction

Retrieval-augmented generation (RAG) is widely utilized for Large language models (LLMs) to address associated challenges, such as timeliness and lack of awareness of domain-specific knowledge, particularly when it comes to private or on-premise information, as this can impact their reliability, leading to issues with faithfulness and the generation of hallucinations. The main idea of RAG is to condition a pre-trained LLM on relevant information retrieved from external data sources based on a user’s input during the generation process (Lewis et al., 2021; Ram et al., 2023).

However, RAG systems encounter their own set of challenges. Firstly, they inherit certain limitations from LLMs, such as constraints on input tokens and difficulties in effectively addressing broad questions within extensive text corpora. Secondly, a RAG system is inherently complex, comprising numerous components that require significant time and expertise to configure and integrate seamlessly. Thirdly, the retrieval process can introduce latency, impacting response times. Additionally, training both the retriever and generation models demands meticulous tuning and substantial computational resources. Furthermore, variations in the underlying data can complicate integration and scalability efforts. Approaches addressing these challenges include optimizing the storage and semantic representation of external data, aligning queries and documents to accurately retrieve the relevant external information (Wang et al., 2023; Ma et al., 2023a; Zheng et al., 2024), improving imperfect retrieval performance by analyzing, identifying and removing irrelevant information (Wang et al., 2025), generating query samples to train retrievers for specific domains (Dai et al., 2022), fine-tuning retriever (Shi et al., 2023), compressing (Yang et al., 2023; Xu et al., 2023; Ma et al., 2023b), re-ranking retrieved information before passing it to LLMs (Zhuang et al., 2023), and fine-tuning LLMs for generating results (Cheng et al., 2023; Kang et al., 2023; Li et al., 2023b). While each of these techniques, whether employed individually or in combination, has the potential to enhance performance, certain questions persist for which generating an accurate response remains challenging.

We propose the integration of an additional question-answer pairs (QA-pairs)-based search module, designed to be incorporated as a preparatory component preceding the RAG system (cf. Figure 1), in which a pre-prepared list of existing QA-pairs is directly utilized as context by LLMs to

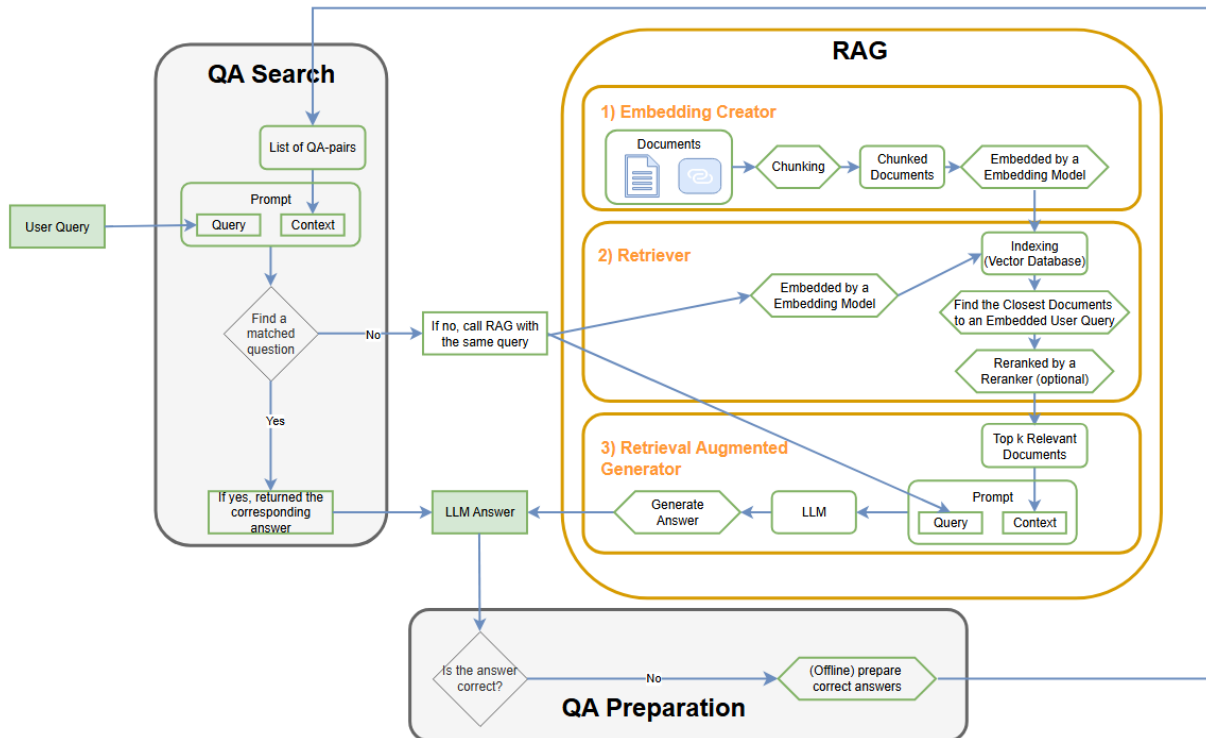


Figure 1: RAG with Integration of a QA Search Module

address user queries. If a question with the same semantic meaning is identified within the list, the associated answer is returned directly. Conversely, if no matching question is found, the user query is forwarded to the standard RAG process. The list of QA-pairs can be created either a priori or continuously during the process. As illustrated in Figure 1, questions for which the RAG system fails to generate correct answers can be collected. Subsequently, the expected answers to these questions can be prepared offline and appended to the list within the QA search module.

Contributions. Our proposed QA search module offers three key contributions:

1. It significantly enhances the overall accuracy of the RAG system. Through experiments conducted with various LLMs and QA datasets, we demonstrate the effectiveness of semantic search. For questions that are incorrectly answered by RAG, we curate correct responses and compile a QA-pairs catalog, enabling precise and efficient semantic retrieval.
2. The QA search module reduces average processing time and resource consumption, as evidenced by our experimental results, which highlight its efficiency.
3. All these benefits are achieved through a

lightweight extension of the RAG system, ensuring that the module remains easy to update and scale as needed.

2 Background and Related Work

To tackle challenges such as timeliness and the lack of awareness regarding private, domain-specific knowledge in pre-trained LLMs, various techniques have been developed. Among these, RAG is gaining significant attention and widespread application. While the implementation of RAG can vary in practice, the fundamental concept remains consistent: for a given input request, relevant documents are retrieved from external knowledge sources, incorporated with the input to assist LLMs during the generation process.

REALM (Guu et al., 2020) employs a masked language model (MLM) and fine-tunes it for open domain question-answering (Open-QA). Similarly focused on Open-QA, RAG (Lewis et al., 2021) trains an autoregressive language model (ALM), while Atlas (Izacard et al., 2022) utilizes an encoder-decoder LLM. In-Context RALM (Ram et al., 2023) and Replug (Shi et al., 2023) simplify the process further by treating the LLM as a black box, eliminating the need for additional training. In these approaches, retrieved information is prepended as context to the query before

being passed to the off-the-shelf LLMs for generation. kNN-LM (Khandelwal et al., 2020) offers an even more practical approach by treating a question as a prefix to its answer. It transforms the task of finding the correct answer into locating the most similar expressions to the input question within an embedding space. This space is created by a LLM on a knowledge dataset. The k most similar results are then interpolated with the original model outputs to construct the final answer. Moreover, both Zero-Shot-Learners (Wei et al., 2022) and Zero-Shot-Reasoners (Kojima et al., 2023) demonstrate exceptional language understanding capabilities inherent in LLMs, particularly in the zero-shot implementation of the so-called Chain of Thought (CoT, Wei et al., 2023).

3 Approach

To answer a question x using the classic RAG paradigm, two steps are involved, retrieval and generating. First, given a question x , the most relevant documents z are retrieved from a knowledge corpus \mathcal{Z} . This step can be represented by a probability distribution $p(z | x), z \in \mathcal{Z}$. Next, both the input question x and the retrieved documents z are provided to a LLM to generate the answer y . This generation step can be modeled as $p(y | z, x)$. By considering the retrieved documents z as a latent variable and marginalizing over all documents in \mathcal{Z} , the entire process can be formalized as:

$$p(y | x) = \sum_{z \in \mathcal{Z}} p(y | z, x) p(z | x) \quad (1)$$

Our new approach takes the simplification one step further by providing an existing list of QA-pairs as context, instead of retrieving most similar external documents to each given question and providing these as context to the question to a LLM to generate the answer. By doing so, we leverage the model to identify the most closely matched questions from the list and return the corresponding answer to new input questions. This approach transforms a QA task into a task of finding semantically similar expressions. It's common for the input question to differ in wording from those in the existing QA-list. However, LLMs excel at understanding and interpreting the semantic meaning of text, making them well-suited for this task.

We begin with an existing list of QA-pairs $\mathcal{Z} = \bigcup_{i=1}^n \{(x_i, y_i)\}$, which is assumed to encompass the

most frequently asked questions. Instead of employing a retrieval step, we provide this entire set as context to LLMs. This implies that $p(z | x) = 1$, resulting in

$$p(y | x) = \sum_{z \in \mathcal{Z}} p(y | z, x) \quad (2)$$

The initial list does not need to be exhaustive. If we encounter a new question x' , that is not already included in the set of questions ($x' \notin \bigcup_{(x_i, y_i) \in \mathcal{Z}} \{x_i\}$), we can prepare the corresponding answer y' and update the the QA-list to $\mathcal{Z}' = \mathcal{Z} \cup \{(x', y')\}$.

When the QA-list is too extensive to fit entirely as context into LLMs due to input token limit constraints, we can assign a reference number to each QA-pair. This forms an expanded set $\mathcal{Z}_{exp} = \bigcup_{i=1}^n \{(x_i, y_i, n_i)\}$. Instead of passing the list of QA-pairs to the LLMs, we provide a list of question-number pairs (QN-pairs) $\mathcal{Z}_{red} = \bigcup_{i=1}^n \{(x_i, n_i)\}$, which reduces the context size. The LLMs then return the corresponding reference number if the input question matches one in the list; otherwise, they return a special number, such as "-1". Using the returned reference number, we can identify the actual answer by performing a lookup in the expanded QA-list \mathcal{Z}_{exp} : $y_i = f_{lkp}(\mathcal{Z}_{exp}, n_i)$.

If the list of the QN-pairs still exceeds the input token limit even after replacing actual answers with reference numbers, we can divide the list into small blocks, each within the token limit. We can then process these blocks sequentially using LLMs. The loop can be terminated early if a valid reference number is returned.

The remaining question is how effectively LLMs can interpret various expressions of an input question x , which may differ in length and form. Specifically, we aim to assess their ability to identify the expression with the same meaning in set \mathcal{Z}_{exp} and return its corresponding reference number. To achieve this, we designed the question-question similarity check in a lightweight manner by implementing a direct search within a prompt, rather than relying on the cosine similarity of embedded questions. This approach leverages the strength of LLMs in capturing semantic information from diverse expressions effectively.

4 Feasibility Assessment Experiments

We assessed the feasibility of our approach by designing experiments across three dimensions: (i) Open-QA datasets, (ii) LLMs for rephrasing original questions and (iii) LLMs for answering original and rephrased questions.

4.1 Experimental Setup

Datasets We selected four publicly available QA datasets for our study. Two of these are in English: OpenBookQA (Mihaylov et al., 2018), CMU-Wiki-QA (Smith et al., 2008), one in Chinese: CMMLU (Li et al., 2023a) and one in German: LHM-Dienstleistungen-QA (Schröder et al., 2022).

LLMs The key success factor of our approach is the ability of LLMs to effectively interpret various expressions of a given question and identify it within a context. To evaluate this we selected several LLMs to rephrase the original questions and to answer both the original and rephrased questions.

These models are GPT3.5, GPT4o (OpenAI, 2023) and seven open-source models: Llama 3.1:8B, Llama 3.3:70B (META, 2024), Qwen2:7B, Qwen2.5:7B (Group, 2024), Mixtral8:7B, Mixtral8:22B (Jiang et al., 2024), Gemma2:9B, Gemma2:27B (DeepMind, 2024).

4.2 Implementation Details

We divided the entire implementation process into five distinct steps.

Preparing Datasets The four QA datasets were sourced from [Hugging Face](#). We selected a random sample of 150 QA-pairs from each dataset and assigned a reference number to each question.

Rephrasing Questions We utilized nine selected LLMs to generate six different variants of each original question across all datasets. These variants maintain the same meaning as the original question. Specifically, three variants were generated without any length restrictions, while the other three were limited to a maximum of 30 characters. The shorter variants were designed to mimic situations where users prefer to keep questions concise. For each original question, we obtained a total of 55 different expressions, including the original question itself: 1 (original question) + [3 (rephrased variants) + 3 (rephrased short variants)] * 9 (models for rephrasing) = 55. More details see Table 2. The prompt template utilized is available in Figure 6.

Algorithm 1 Answering Questions by QA Search

```
1:  $N_{qn\_blocks} \leftarrow \text{ceil}(n_{total\_qn}/\text{block\_size})$ 
2: for each question  $q$  do
3:   for  $i$  from 0 to  $N_{qn\_blocks}$  do
4:      $C_i \leftarrow \text{create\_context}(qn\_block_i)$ 
5:      $a\_n \leftarrow \text{call\_qa\_search}(q, C_i)$ 
6:     if  $a\_n > -1$  then
7:        $\text{answer} \leftarrow f_{l_{kp}}(L_{QAN}, a\_n)$ 
8:       return  $\text{answer}$ 
9:     end if
10:  end for
11: end for
```

Answering Questions We used ten LLMs to answer all question variants, by utilizing the list of original QN-pairs as context. As Algorithm 1 shows, if a match is not found in the list, the corresponding reference number is returned, otherwise "-1" is returned. The prompt template utilized is available in Figure 7.

Both Mixtral8 models returned numbers with wrapping text, such as "The reference number of its corresponding answer is: 1037", instead of "1037". An additional script was created to remove the extraneous text.

Using all 150 original QN-pairs as context worked well with both GPT models. However, for the open-source models, after several rounds of experimentation, we decided to split the full list into five blocks with 30 questions each. We then looped over these blocks until a reference number is returned. Due to text wrapping issues, we needed to collect results from all five blocks without breaking the loop early when using both Mixtral8 models.

Evaluating Answers We compared the reference numbers returned by the LLMs with the ground truth. If they match, we assigned a score of 1; otherwise, we assigned a score of 0. We then aggregated these scores to calculate the accuracy rate of the answers across all question variants for each model.

Robustness Testing with Different List and Block Sizes To demonstrate the robustness of performance as well as resource and time consumption, we conducted additional experiments with varying total numbers of QA-pairs (300, 600, 900, 1200, 1500) and different numbers of QA-pairs (30, 50, 100, 150, 300) used as query context.

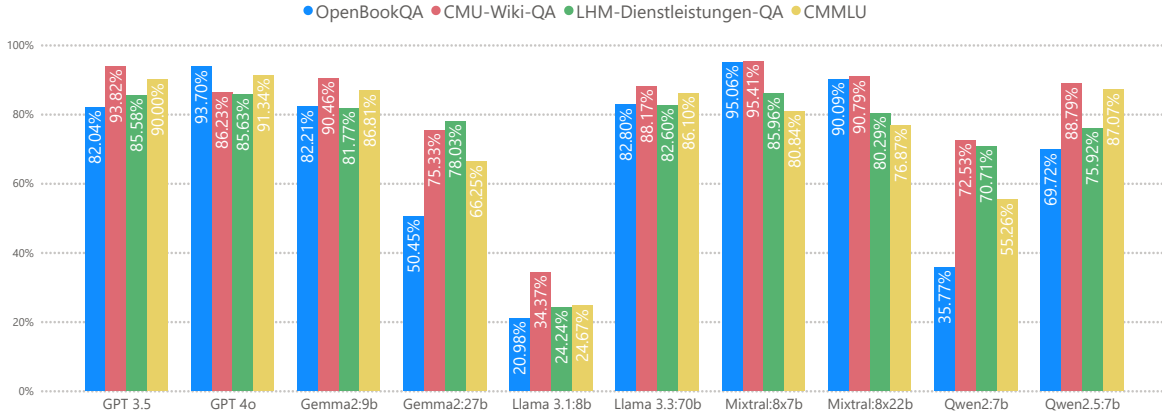


Figure 2: Average Percentage of Correctly Answered Questions for Four QA Datasets by Each Model

4.3 Preliminary Experimental Results

Comprehensive results for the feasibility experiment for the four QA datasets are available in Tables 3-6. Additionally, we calculated the average answering accuracy of each applied model across all four dataset. As illustrated in Figure 2, all models demonstrate consistent performance across the four datasets, which include different languages: English, Chinese, German. This consistency holds true regardless of whether the models achieve high or low accuracy. The GPT models, in particular, exhibit best overall performance. This is not only due to their average accuracy rates exceeding 80%, but also because they do not require the question list to be divided into blocks for iterative processing.

Among the open-source models, Mixtral8:7B, Gemma2:9B and Llama 3.3:70B achieve an accuracy rate greater than 80% across all datasets. Mixtral8:22B follows closely, with only one dataset falling below 80%, while Qwen2.5:7B has two datasets under this threshold. In contrast, the average accuracy rates for Gemma2:27B and Qwen2:7B fall below 80%, ranging between 36% and 78%. Surprisingly, Llama 3.1:8B performs the worst, with all accuracy rates below 35%.

By expanding the total number of QA-pairs in the catalog from 300 to 1500, while maintaining a fixed context size of 150 QAs, and using GPT4o as the answering LLM, we observed, as expected, no significant increase in CPU, GPU, or memory usage. However, this expansion resulted in longer average response time, increasing from 1.39 seconds per query to 6.15 seconds per query, due to increased number of max. needed requests. As illustrated in Figure 3, the mean accuracy experi-

enced a slight decline as the catalog size increased, dropping from 93.65% to 89.35%. This reduction in performance can be attributed to the higher likelihood of encountering questions with similar but not identical meanings as the catalog grows, which increases the difficulty of semantic search for the LLM.

If, instead, we fixed the total number of QA-pairs in the question catalog but expanded the context size from 30 to 300, we observed, as shown in Figure 4, again, no significant increase in CPU, GPU, or memory usage. And as expected, the average response time decreased from 4.36 seconds to 1.28 seconds per query. The mean accuracy improved significantly, increasing from 88.13% to 95.48%. This improvement can be attributed to the fact that, as long as the expanded context size remains within the token limit of the LLM, having more questions with similar but not identical meanings in a single context allows the LLM to better identify and select the one with the closest semantic match.

The detailed metrics for both Figure 3 and 4 are provided in Table 7 and 8. The four QA datasets, all codes and generated outputs for this experiment are available on GitHub¹.

5 Comparative Experiments between RAG and RAG with a QA Search Module

After successfully demonstrating the feasibility of a QA search, we conducted another group of experiments to showcase the enhanced effectiveness

¹<https://github.com/chenzhengdao/pattern-of-driving-license-written-exam>, <https://github.com/chenzhengdao/pattern-of-driving-license-written-exam-compare-with-rag>

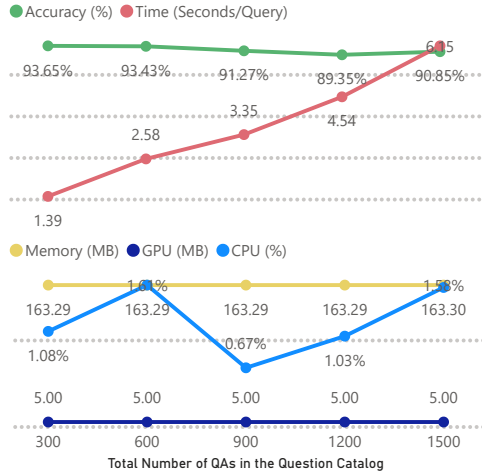


Figure 3: Accuracy, Resource and Time Consumption Across Varying Total Numbers of Questions in the Question Catalog, with 150 Questions as QA Search Context and GPT4o as the Answering LLM

and efficiency of a RAG system by integrating a QA search module. To achieve this, we setup experiments for the following four approaches and compared their results: i) RAG: baseline RAG, ii) QA search: as evaluated in the previous section, iii) RAG + QA search: applying QA search to questions incorrectly answered by RAG iv) QA search + RAG: applying RAG to questions incorrectly answered by QA search. The last approach is exactly what Figure 1 illustrates.

5.1 Experimental Setup

RAG As illustrated in the top-right section of Figure 1, the RAG system used in the following experiments consists of three main modules, inspired by (Lewis et al., 2021) and (Gao et al., 2024): (i) an embedding creator, (ii) a retriever, and (iii) a retrieval-augmented generator. Initially, documents are segmented into smaller chunks, which are then transformed into embeddings using an embedding model. These embeddings are stored in a vector database as a knowledge base through an indexing algorithm, making them ready for efficient retrieval. When a user submits a query to the RAG system, the query is embedded using the same embedding model and sent to the vector database to identify the top k most relevant documents. In some cases, a reranker, typically another LLM, may be employed to reorder the retrieved documents for improved relevance. Finally, a prompt is constructed by combining the user query with the top k documents as

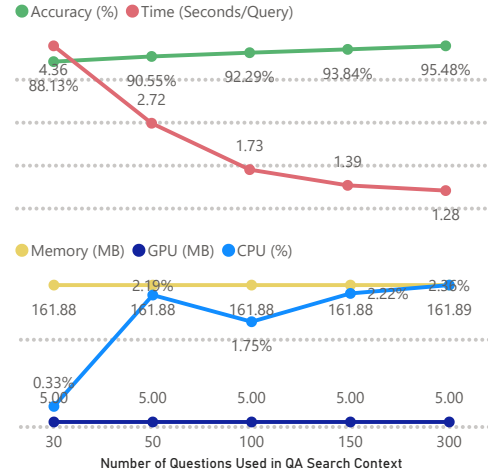


Figure 4: Accuracy, Resource and Time Consumption Across Varying Number of Questions Used as QA Search Context, with 300 as Total Number of Questions in Question Catalog and GPT4o as the Answering LLM

context, which is then sent to an LLM to generate the final response to the user’s query.

Datasets The four QA datasets used for the feasibility assessment in last section are unsuitable for this experiment, as they consist solely of QA-pairs and do not include the corresponding text blocks from which the answers can be generated. We chose two other publicly available document sets: (i) **it-tech-doc**, an IT-technical document set and (ii) **financial-news**, a financial news document set.

Building Synthetic QA-pairs We divided each document from both document sets into chunks, with a size of 2000 tokens and an overlap of 200 tokens. For each document chunk, we utilized two LLMs (GPT3.5 and GPT4o) to generate synthetic QA-pairs. We then evaluated the quality of the generated QA-pairs, assigning a score between 1 (poor) and 5 (excellent) based on three criteria: (i) **Groundedness**: can the question be answered from the given context? (ii) **Relevance**: is the question relevant for a given topic? (iii) **Stand-alone**: is the question understandable free of any context? We retained QA-pairs with best scores across all three criteria and selected 150 samples from each QA set for further processing. The prompt templates utilized here are available in Figure 8-11.

Building Test Cases for Evaluating RAG Evaluating a RAG system can be complex due to various factors that influence its overall performance. But

it is not the primary focus of this experiment to try out different combinations of influence factors. For this work we simply used two chunk sizes (200 and 400) to create chunking, applied one embedding model (GTE-small, a variant from the General Text Embedding (GTE) series developed by Alibaba Group (Li et al., 2023c)) to embed and retrieve embedded chunks. We then applied two approaches to the retrieved chunks: one without a reranker and one with a reranker (colbertv2.0, a model developed at Stanford University (Santhanam et al., 2022)). Furthermore, we used two LLMs (the two top performers from the feasibility experiment in Section 4: GPT4o and Llama 3.3:70B) to generate the answer to a given question based on the found chunks and another LLM for evaluating the generated answers. This in total made up 16 test cases for the baseline RAG evaluation: 2 (LLMs for QA generation) \times 1 (LLM for QA evaluation) \times 2 (chunk sizes for embedding) \times 1 (embedding model) \times 2 (options for reranking, with or without) \times 2 (LLMs for answer generation) = 16. Finally, for each of the 16 test cases we obtained an average accuracy score for generated answers across all 150 sample questions, and made the average of the 16 average accuracy scores as the final accuracy to the baseline RAG. Details about the 16 combinations for both document sets can be found in Table 9- 10.

Building Test Cases for Evaluating QA Search

After the comprehensive feasibility experiments in Section 4, we intended not to construct too many test cases for the QA search testing. To each of the synthetic QA-pairs sample sets, we applied one LLM (GPT4o) for rephrasing the original question, and two LLMs (GPT4o and Llama 3.3:70B) for QA search. This made in total 8 test cases: 2 (LLMs for synthetic QA generation) \times 1 (LLM for synthetic QA evaluation) \times 1 (LLM for rephrasing question) \times 2 (question types, original and rephrased) \times 2 (LLMs for QA search) = 8. Details about the 8 test cases for both document sets can be found in Table 11- 12.

5.2 Implementation Details

For each of the document sets (it-tech-doc and financial-news), we applied the following four approaches:

Answering Questions with RAG For each of the 16 test cases, we answered the sampled questions by following the RAG pipeline described in Figure 1 and detailed in Algorithm 2. The prompt

templates used in this process are provided in Figure 12.

Unlike the binary evaluation used for QA search, as described in Subsection 4.2, the answers generated by RAG are evaluated using a scoring scale from 1 (worst) to 5 (best). Answers receiving a score of 4 or 5 are considered correct, while those scoring below 4 are deemed incorrect. The prompt template used for this evaluation is provided in Figure 13.

Algorithm 2 Answering Questions by RAG

```

1: for each question  $q$  do
2:    $q_e \leftarrow \text{embed}(q)$ 
3:    $D_k \leftarrow \text{retrieve}(q_e, D, k)$ 
4:   if reranking then  $D_{kr} \leftarrow \text{rerank}(D_k)$ 
5:      $a \leftarrow \text{generate}(D_{kr}, q)$ 
6:   else
7:      $a \leftarrow \text{generate}(D_k, q)$ 
8:   end if
9:   Save answer  $a$ 
10: end for

```

Applying QA Search to Questions Incorrectly Answered by RAG

In the next step, we collected all the questions that were incorrectly answered in the previous phase, rephrased them, and then applied Algorithm 1 to the rephrased questions, using the list of original QA-pairs as context.

Answering Questions with QA Search For all 8 test cases of QA search, we applied the semantic search of questions among the list of original QA-pairs as described in Section 3-4.

Applying RAG to Questions Incorrectly Answered by QA Search

We gathered all the questions that were incorrectly answered by the QA search in last step. We then applied Algorithm 2 to each of them to generate answer, and calculated the final accuracy.

The two document sets, all codes and generated outputs for this experiment are available at this [Git repository 2](#).

5.3 Main Experimental Results

5.3.1 Accuracy Improvement

Based on Figure 5, it is evident that, without any parameter tuning on either side, the three alternative approaches consistently outperform the baseline RAG across both document sets. By applying the

Measure Method	GPU (MB)		Memory (MB)		CPU (%)		Time (Sec/query)	
	RAG	QA Search	RAG	QA Search	RAG	QA Search	RAG	QA Search
it-tech-doc	899	6	1,741	161	5.92%	2.09%	11.68	0.81
financial-news	899	6	1,865	162	5.82%	2.00%	10.93	0.73

Table 1: Mean Resource & Time Consumption for RAG and QA Search

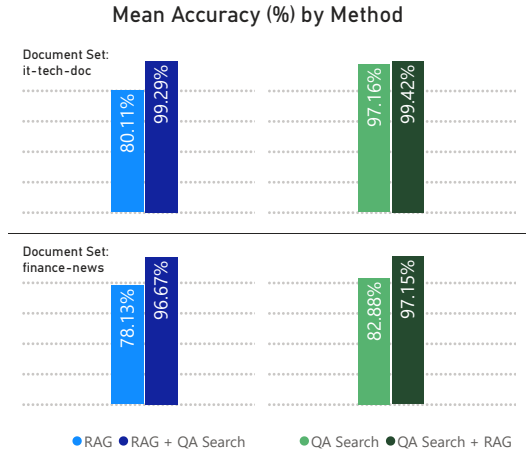


Figure 5: Mean Accuracy of Four Methods: i) RAG, ii) RAG + QA Search, iii) QA Search, iv) QA Search + RAG

QA search method to the questions incorrectly answered by the baseline RAG, the accuracy can be significantly improved—from 80.11% to 99.29% for one document set and from 78.13% to 96.67% for the other.

Remarkably, the QA search approach alone delivers superior results compared to the baseline RAG, with accuracy improvements of 17% and 4%, respectively. Furthermore, the final combined approach (QA search + RAG), achieves the highest mean accuracy, reaching 99.42% for it-tech-doc and 97.15% for financial-news. This optimal approach is precisely the one depicted in Figure 1.

The detailed average accuracy for all test cases is provided in Tables 9–12.

5.3.2 Efficiency Improvement

In terms of computational resource consumption per user query, the QA search approach demonstrates significant advantages over the traditional RAG. As shown in Table 1, querying with QA search requires no GPU resources (the 6 MB listed in the table is attributed solely to operating system overhead), whereas the RAG setup described in Subsection 5.1 demands nearly 900 MB of GPU memory. Additionally, RAG exhibits over 10 times

higher memory consumption and nearly 3 times greater CPU usage compared to QA search.

Thanks to its lightweight architecture, the QA search approach achieves a significantly shorter average response time compared to RAG—under 1 second (0.81 or 0.73 seconds) versus over 10 seconds (11.68 or 10.93 seconds) for RAG. It is important to note that the reported response times are based on a question catalog containing 150 entries, all consolidated into a single query context. For a detailed comparison of response times across different catalog sizes or context configurations, please refer to Figures 3 and 4.

Detailed average resource consumption metrics are provided in Tables 13–16. It is worth noting that in our experiments, local open-source LLMs used for answering questions are hosted as separate standalone processes and are not part of the query processing pipeline. Consequently, their GPU consumption is excluded from the resource measurements for both RAG and QA search.

5.3.3 Expert Answer instead of Hallucination

For QA-pairs used in the QA search module, answers can be thoroughly prepared in advance. For instance, we can engage experts to formulate or review these answers. Additionally, we can enhance the responses by including supplementary information, such as precise references to citation sources. Complex questions that require insights from extensive text corpora typically present significant challenges to the RAG paradigm. However, with the QA search, they can also be effectively addressed in advance.

5.3.4 Easy Setup, Update, High Efficiency and Stability

The QA search module is designed for seamless integration with RAG, and it can be initiated with an empty list of pre-known QA-pairs. Over time, answers generated by RAG can be evaluated and labeled by human and/or LLMs. Questions for which RAG fails to generate correct answers are collected. These questions, along with their (offline) prepared answers, are added to the QA search module’s list for future semantic searches. If the list exceeds

the maximum token limit, it can be divided into manageable blocks. Additionally, questions that RAG can answer correctly may also be added to the list, as QA search typically requires less computational power and delivers shorter response time. Moreover, the answers returned by the QA search module remain consistent, enhancing the overall stability of the system’s responses.

6 Conclusion

In this work, we introduced and demonstrated a straightforward QA search approach as an additional, integrable component to a traditional RAG pipeline, particularly for scenarios where the baseline RAG struggles to generate accurate answers for certain questions. Our empirical analysis shows that even with a simple setup, this approach can enhance performance and reduce the resource consumption of the conventional RAG paradigm. Furthermore, it enables applications to be implemented, updated, and scaled in an efficient and effective manner.

Limitations

This work has three main limitations. First, while we successfully designed experiments to compare our approach with RAG using the same datasets, we explored only a limited number of combinations of adjustable factors within the RAG pipeline. The baseline accuracy of a RAG approach can vary significantly depending on the dataset used and the specific implementation details throughout the entire RAG process. Second, we did not conduct an in-depth analysis of the incorrectly answered questions (by both RAG and QA search) to identify the precise reasons for these failures. Third, for the comparative experiments, we relied on LLMs to generate synthetic QA-pairs from two document sets, as the four QA datasets used in the feasibility experiments do not include corresponding text blocks. But we began our work using some of our internal data sets, which are not accessible to LLMs. Following the success of our experiments with these internal cases, we decided to publish this concept. To ensure data confidentiality, we utilized public data instead of our actual internal data for the publication. However, we achieved comparable results and accuracy using both the real internal data and the publicly available synthetic data/QA-pairs. Additionally, we depended on LLMs to evaluate the quality of all generated

QA-pairs and were only able to manually verify 10% of them through random sampling.

Acknowledgments

Matthias Aßenmacher received funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the National Research Data Infrastructure – NFDI 27/1 - 460037581 - BERD@NFDI.

References

- Xin Cheng, Di Luo, Xiuying Chen, Lemao Liu, Dongyan Zhao, and Rui Yan. 2023. [Lift yourself up: Retrieval-augmented text generation with self memory](#). *Preprint*, arXiv:2305.02437.
- Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. 2022. [Promptagator: Few-shot dense retrieval from 8 examples](#). *Preprint*, arXiv:2209.11755.
- Google DeepMind. 2024. [Gemma 2: Improving open language models at a practical size](#). *Preprint*, arXiv:2408.00118.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. [Retrieval-augmented generation for large language models: A survey](#). *Preprint*, arXiv:2312.10997.
- Alibaba Group. 2024. [Qwen2 technical report](#). *Preprint*, arXiv:2407.10671.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. [Realm: Retrieval-augmented language model pre-training](#). *Preprint*, arXiv:2002.08909.
- Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022. [Atlas: Few-shot learning with retrieval augmented language models](#). *Preprint*, arXiv:2208.03299.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L  lio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Th  ophile Gervet, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. 2024. [Mixtral of experts](#). *Preprint*, arXiv:2401.04088.
- Minki Kang, Jin Myung Kwak, Jinheon Baek, and Sung Ju Hwang. 2023. [Knowledge graph-augmented](#)

- language models for knowledge-grounded dialogue generation. *Preprint*, arXiv:2305.18846.
- Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. *Preprint*, arXiv:1911.00172.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners. *Preprint*, arXiv:2205.11916.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.
- Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023a. Cmmlu: Measuring massive multi-task language understanding in chinese. *Preprint*, arXiv:2306.09212.
- Xinze Li, Zhenghao Liu, Chenyan Xiong, Shi Yu, Yu Gu, Zhiyuan Liu, and Ge Yu. 2023b. Structure-aware language model pretraining improves dense retrieval on structured data. *Preprint*, arXiv:2305.19912.
- Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023c. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.
- Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023a. Query rewriting for retrieval-augmented large language models. *Preprint*, arXiv:2305.14283.
- Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023b. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics.
- META. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.
- OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.
- Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Preprint*, arXiv:2302.00083.
- Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. Colbertv2: Effective and efficient retrieval via lightweight late interaction. *Preprint*, arXiv:2112.01488.
- Leon Marius Schröder, Clemens Gutknecht, Oubada Alkiddeh, Susanne Weiß, and Leon Lukas. 2022. Lhm-dienstleistungen-qa - german public domain question-answering dataset.
- Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *Preprint*, arXiv:2301.12652.
- Noah A. Smith, Michael Heilman, and Rebecca Hwa. 2008. Question generation as a competitive undergraduate course project. In *The NSF Workshop on the Question Generation Shared Task and Evaluation Challenge, Arlington, VA, September 2008*.
- Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan Ö. Arık. 2025. Astute rag: Overcoming imperfect retrieval augmentation and knowledge conflicts for large language models. *Preprint*, arXiv:2410.07176.
- Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. *Preprint*, arXiv:2303.07678.
- Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. *Preprint*, arXiv:2109.01652.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.
- Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *Preprint*, arXiv:2310.04408.
- Haoyan Yang, Zhitao Li, Yong Zhang, Jianzong Wang, Ning Cheng, Ming Li, and Jing Xiao. 2023. Prca: Fitting black-box large language models for retrieval question answering via pluggable reward-driven contextual adapter. *Preprint*, arXiv:2310.18347.
- Huaxiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H. Chi, Quoc V Le, and Denny Zhou. 2024. Take a step back: Evoking reasoning via abstraction in large language models. *Preprint*, arXiv:2310.06117.
- Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. 2023. Open-source large language models are strong zero-shot query likelihood models for document ranking. *Preprint*, arXiv:2310.13243.

A Prompts

```
""
You will be provided with question delimited by triple quotes. \
Please create {num_question} questions based on it with same meaning in {lang}, without quotes. \
Please list only the questions, no other text around. \
{text_length_spec}

\\"\{question}\\"
""
text_length_spec = "And it should not more than {question_length} characters."
```

Figure 6: Prompt for Rephrasing Question

```
""
You will be provided with text delimited by triple quotes. \
It's a table of questions and answers, where answers are in the form of a reference number. \
If a question can be found in the table, return the reference number of the answer.

If the question cannot be found in the table, \
then simply return \"-1\"

\\"\{questions_in_list}\\"
""
```

Figure 7: Prompt for Answering Question with New Approach

```
""
Your task is to write a factoid question and an answer given a context.
Your factoid question should be answerable with a specific, concise piece of factual information from the context.
Your factoid question should be formulated in the same style as questions users could ask in a search engine.
This means that your factoid question MUST NOT mention something like "according to the passage" or "context".

Provide your answer as follows:

Output::
Factoid question: (your factoid question)
Answer: (your answer to the factoid question)

Now here is the context.

Context: {context}\n
Output::""
```

Figure 8: Prompt for Generating Synthetic QA-pairs

```
question_groundedness_critique_prompt = """You will be given a context and a question.
Your task is to provide a 'total rating' scoring how well one can answer the given question unambiguously with the given context.
Give your answer on a scale of 1 to 5, where 1 means that the question is not answerable at all given the context, and 5 means that
the question is clearly and unambiguously answerable with the context.

Provide your answer as follows:

Answer::
Evaluation: (your rationale for the rating, as a text)
Total rating: (your rating, as a number between 1 and 5)

You MUST provide values for 'Evaluation:' and 'Total rating:' in your answer.

Now here are the question and context.

Question: {question}\n
Context: {context}\n
Answer:: """
```

Figure 9: Prompt for Evaluating Generated QA-pairs with Critique Groundedness

```
"""You will be given a question.
Your task is to provide a 'total rating' representing how useful this question can be to machine learning developers building
NLP applications with the Hugging Face ecosystem.
Give your answer on a scale of 1 to 5, where 1 means that the question is not useful at all, and 5 means that the question is
extremely useful.

Provide your answer as follows:

Answer::
Evaluation: (your rationale for the rating, as a text)
Total rating: (your rating, as a number between 1 and 5)

You MUST provide values for 'Evaluation:' and 'Total rating:' in your answer.

Now here is the question.

Question: {question}\n
Answer:: """
```

Figure 10: Prompt for Evaluating Generated QA-pairs with Critique Relevance

```

"""You will be given a question.
Your task is to provide a 'total rating' representing how context-independent this question is.
Give your answer on a scale of 1 to 5, where 1 means that the question depends on additional information to be understood,
and 5 means that the question makes sense by itself.
For instance, if the question refers to a particular setting, like 'in the context' or 'in the document', the rating must be 1.
The questions can contain obscure technical nouns or acronyms like Gradio, Hub, Hugging Face or Space and still be a 5: it
must simply be clear to an operator with access to documentation what the question is about.

For instance, "What is the name of the checkpoint from which the ViT model is imported?" should receive a 1, since there is
an implicit mention of a context, thus the question is not independent from the context.

Provide your answer as follows:

Answer::
Evaluation: (your rationale for the rating, as a text)
Total rating: (your rating, as a number between 1 and 5)

You MUST provide values for 'Evaluation:' and 'Total rating:' in your answer.

Now here is the question.

Question: {question}\n
Answer:: """

```

Figure 11: Prompt for Evaluating Generated QA-pairs with Critique Stand-alone

```

"""<|system|>
Using the information contained in the context,
give a comprehensive answer to the question.
Respond only to the question asked, response should be concise and relevant to the question.
Provide the number of the source document when relevant.
If the answer cannot be deduced from the context, do not give an answer.</s>
<|user|>
Context:
{context}
---
Now here is the question you need to answer.

Question: {question}
</s>
<|assistant|>
"""

```

Figure 12: Prompt for Answering Question with RAG

```

"""###Task Description:
An instruction (might include an Input inside it), a response to evaluate, a reference answer that gets a score of 5, and a score
rubric representing a evaluation criteria are given.
1. Write a detailed feedback that assess the quality of the response strictly based on the given score rubric, not evaluating in
general.
2. After writing a feedback, write a score that is an integer between 1 and 5. You should refer to the score rubric.
3. The output format should look as follows: \"Feedback: {{write a feedback for criteria}} [RESULT] {{an integer number
between 1 and 5}}\"
4. Please do not generate any other opening, closing, and explanations. Be sure to include [RESULT] in your output.

###The instruction to evaluate:
{instruction}

###Response to evaluate:
{response}

###Reference Answer (Score 5):
{reference_answer}

###Score Rubrics:
[Is the response correct, accurate, and factual based on the reference answer?]
Score 1: The response is completely incorrect, inaccurate, and/or not factual.
Score 2: The response is mostly incorrect, inaccurate, and/or not factual.
Score 3: The response is somewhat correct, accurate, and/or factual.
Score 4: The response is mostly correct, accurate, and factual.
Score 5: The response is completely correct, accurate, and factual.

###Feedback:"""

```

Figure 13: Prompt for Evaluating Answers Generated with RAG

B Experiment Setup

Question	Answer	Rephr. Variant	Question rephr. by Model 1	Short Question rephr. by Model 1	...	Question rephr. by Model 9	Short Question rephr. by Model 9
Q1	A1	1	rephr. variant 1 to Q1 by model 1	rephr. short variant 1 to Q1 by model 1	...	rephr. variant 1 to Q1 by model 9	rephr. short variant 1 to Q1 by model 9
		2	rephr. variant 2 to Q1 by model 1	rephr. short variant 2 to Q1 by model 1	...	rephr. variant 2 to Q1 by model 9	rephr. short variant 2 to Q1 by model 9
		3	rephr. variant 3 to Q1 by model 1	rephr. short variant 3 to Q1 by model 1	...	rephr. variant 3 to Q1 by model 9	rephr. short variant 3 to Q1 by model 9
Q2	A2	1	rephr. variant 1 to Q2 by model 1	rephr. short variant 1 to Q2 by model 1	...	rephr. variant 1 to Q2 by model 9	rephr. short variant 1 to Q2 by model 9
		2	rephr. variant 2 to Q2 by model 1	rephr. short variant 2 to Q2 by model 1	...	rephr. variant 2 to Q2 by model 9	rephr. short variant 2 to Q2 by model 9
		3	rephr. variant 3 to Q2 by model 1	rephr. short variant 3 to Q2 by model 1	...	rephr. variant 3 to Q2 by model 9	rephr. short variant 3 to Q2 by model 9
...	

Table 2: 54 rephrased expressions by 9 models to each original question

C Evaluation Results for Feasibility Experiments

Tables 3-6 below (one table per QA dataset) present questions rephrased by various LLMs, both without and with length limitations (denoted as "short"), arranged vertically. Horizontally, each rephrased question variant is answered by different LLMs. Each value represents the average accuracy across the entire sample set. Bolded values within the table indicate the LLM that achieves the highest answering performance for each question variant. Furthermore, the bolded average values in the bottom row represent the overall answering performance of each LLM across all question variants. These values are also depicted in Figure 2.

Question Rephrased By	Answered By									
	gpt35	gpt4o	llama3.1	llama3.3	qwen2	qwen2.5:7b	mixtral:8x7b	mixtral:8x22b	gemma2	gemma2:27b
original question	92.95%	100.00%	22.76%	87.80%	27.10%	75.34%	97.56%	100.00%	74.53%	38.75%
gpt35	84.28%	96.48%	21.95%	86.72%	40.92%	71.27%	98.10%	89.16%	89.16%	62.06%
gpt35 short	85.64%	94.31%	20.05%	86.72%	32.52%	71.54%	94.85%	90.79%	86.45%	55.83%
gpt4o	88.35%	98.92%	21.14%	84.28%	38.48%	71.54%	99.19%	96.21%	86.18%	51.22%
gpt4o short	85.09%	97.02%	22.49%	86.45%	34.96%	69.38%	97.83%	93.22%	82.93%	56.37%
llama3.1	80.76%	93.22%	20.33%	82.11%	34.69%	71.00%	94.85%	87.53%	82.11%	51.49%
llama3.1 short	72.63%	85.91%	19.78%	78.86%	32.52%	60.16%	86.72%	81.03%	77.78%	50.68%
qwen2	79.67%	95.66%	20.33%	74.80%	38.21%	70.73%	95.93%	88.35%	78.32%	41.73%
qwen2 short	72.09%	84.28%	18.16%	72.36%	35.23%	60.43%	89.16%	81.03%	76.42%	46.07%
qwen2.5:7b	79.67%	95.39%	21.14%	80.76%	35.50%	73.44%	94.58%	92.14%	82.38%	48.78%
qwen2.5:7b short	70.73%	84.01%	18.43%	75.07%	31.17%	64.23%	85.91%	78.59%	74.53%	48.51%
mixtral:8x7b	84.82%	95.93%	22.49%	82.93%	43.09%	74.25%	97.56%	93.77%	85.64%	49.32%
mixtral:8x7b short	84.01%	96.21%	21.41%	83.74%	38.21%	73.71%	98.10%	93.77%	85.91%	52.03%
mixtral:8x22b	83.20%	96.21%	21.95%	79.67%	39.57%	69.11%	97.83%	94.58%	83.47%	46.61%
mixtral:8x22b short	81.84%	88.89%	20.87%	88.62%	32.52%	68.29%	95.12%	91.60%	85.37%	46.88%
gemma2	84.28%	97.02%	22.49%	86.45%	36.59%	73.17%	97.83%	92.41%	85.09%	53.12%
gemma2 short	82.38%	92.14%	20.60%	88.35%	33.33%	71.27%	92.95%	89.97%	81.30%	53.39%
gemma2:27b	86.45%	96.75%	21.68%	83.20%	38.75%	70.46%	98.10%	92.14%	85.09%	54.20%
gemma2:27b short	79.95%	91.87%	20.60%	84.28%	36.31%	65.31%	94.04%	85.37%	79.40%	51.49%
Mean	82.04%	93.70%	20.98%	82.80%	35.77%	69.72%	95.06%	90.09%	82.21%	50.45%

Table 3: Detailed Answering Accuracy Results for QA Set OpenBookQA

Question Rephrased By	Answered By									
	gpt35	gpt4o	llama3.1	llama3.3	qwen2	qwen2.5:7b	mixtral:8x7b	mixtral:8x22b	gemma2	gemma2:27b
original question	98.58%	99.76%	35.93%	95.27%	77.54%	92.91%	100.00%	100.00%	92.20%	76.83%
gpt35	95.04%	85.11%	35.70%	88.65%	74.70%	89.13%	95.98%	87.94%	90.31%	74.94%
gpt35 short	94.56%	80.61%	38.53%	88.18%	69.98%	89.83%	96.22%	91.02%	89.60%	74.00%
gpt4o	96.69%	89.36%	33.81%	89.83%	71.87%	91.25%	96.93%	95.04%	91.25%	73.29%
gpt4o short	96.45%	82.03%	39.24%	91.02%	74.23%	90.78%	95.98%	93.85%	91.96%	76.12%
llama3.1	94.56%	84.87%	32.86%	85.82%	73.52%	88.18%	94.09%	87.71%	91.25%	75.89%
llama3.1 short	89.83%	78.96%	36.88%	84.63%	66.90%	84.40%	90.78%	87.23%	87.94%	77.07%
qwen2	90.78%	90.78%	30.26%	87.00%	75.41%	85.34%	95.74%	89.36%	88.89%	72.81%
qwen2 short	90.31%	84.40%	34.04%	83.45%	71.39%	88.65%	95.51%	85.58%	89.13%	72.34%
qwen2.5:7b	93.62%	91.25%	31.68%	88.18%	72.81%	88.18%	96.22%	90.07%	91.73%	74.00%
qwen2.5:7b short	93.38%	81.80%	35.70%	84.16%	75.41%	89.13%	93.62%	87.23%	90.07%	75.41%
mixtral:8x7b	94.80%	91.25%	33.33%	88.89%	74.70%	90.54%	95.98%	92.91%	90.78%	74.70%
mixtral:8x7b short	95.27%	91.02%	32.62%	90.07%	76.36%	88.65%	95.98%	93.85%	90.78%	76.12%
mixtral:8x22b	93.62%	91.73%	30.97%	88.65%	73.76%	89.60%	96.45%	91.25%	90.31%	73.05%
mixtral:8x22b short	91.96%	78.72%	35.93%	85.58%	72.10%	88.42%	93.85%	89.83%	89.36%	74.94%
gemma2	95.27%	88.89%	33.81%	87.71%	70.45%	90.54%	95.27%	92.43%	91.02%	75.89%
gemma2 short	90.31%	77.54%	33.81%	89.60%	66.19%	84.87%	93.14%	91.02%	90.78%	78.96%
gemma2:27b	93.85%	87.47%	32.15%	88.89%	74.70%	89.13%	97.16%	89.13%	90.07%	77.07%
gemma2:27b short	93.62%	82.74%	35.70%	89.60%	65.96%	87.47%	93.85%	89.60%	91.25%	77.78%
Mean	93.82%	86.23%	34.37%	88.17%	72.53%	88.79%	95.41%	90.79%	90.46%	75.33%

Table 4: Detailed Answering Accuracy Results for QA Set CMU-Wiki-QA

Question Rephrased By	Answered By									
	gpt35	gpt4o	llama3.1	llama3.3	qwen2	qwen2.5:7b	mixtral:8x7b	mixtral:8x22b	gemma2	gemma2:27b
original question	95.45%	99.24%	24.49%	89.90%	79.04%	91.16%	100.00%	100.00%	91.41%	83.59%
gpt35	94.19%	94.70%	24.75%	88.38%	75.51%	85.10%	93.18%	93.18%	89.65%	81.57%
gpt35 short	89.14%	93.18%	21.97%	86.62%	71.21%	79.29%	89.90%	87.12%	83.59%	78.28%
gpt4o	94.19%	98.23%	25.25%	91.16%	78.79%	89.90%	97.98%	95.71%	91.16%	82.07%
gpt4o short	83.84%	82.32%	24.75%	80.56%	69.44%	75.25%	85.61%	78.28%	78.54%	73.99%
llama3.1	86.11%	84.85%	23.74%	77.78%	69.19%	75.51%	86.11%	76.01%	79.80%	78.03%
llama3.1 short	70.96%	64.90%	22.47%	69.70%	63.38%	62.12%	73.74%	60.35%	65.91%	71.21%
qwen2	82.07%	79.55%	23.74%	78.54%	69.95%	71.46%	82.58%	75.76%	83.84%	78.79%
qwen2 short	81.31%	75.51%	27.78%	73.74%	67.93%	71.21%	79.29%	68.18%	78.28%	76.77%
qwen2.5:7b	85.61%	86.62%	23.23%	83.59%	69.95%	74.49%	86.11%	74.24%	84.85%	79.29%
qwen2.5:7b short	68.69%	61.87%	24.24%	71.72%	58.84%	63.13%	70.71%	60.86%	66.92%	65.66%
mixtral:8x7b	93.18%	96.72%	24.24%	89.39%	73.74%	83.84%	92.42%	90.91%	88.13%	80.30%
mixtral:8x7b short	88.89%	92.17%	24.49%	85.61%	77.53%	79.80%	87.63%	86.11%	85.10%	79.55%
mixtral:8x22b	90.15%	92.93%	24.75%	86.62%	75.76%	78.54%	90.40%	85.10%	85.86%	82.83%
mixtral:8x22b short	80.56%	82.32%	26.01%	80.30%	68.43%	66.16%	80.05%	71.97%	76.77%	80.05%
gemma2	91.41%	95.20%	25.51%	87.37%	76.26%	81.57%	91.67%	86.87%	86.36%	79.29%
gemma2 short	80.56%	76.26%	21.72%	80.56%	62.12%	67.17%	81.06%	77.78%	76.26%	76.26%
gemma2:27b	93.18%	93.69%	26.01%	88.38%	74.75%	80.05%	89.14%	89.14%	86.11%	80.30%
gemma2:27b short	76.52%	76.77%	21.46%	79.55%	61.62%	66.67%	75.76%	67.93%	75.00%	74.75%
Mean	85.58%	85.63%	24.24%	82.60%	70.71%	75.92%	85.96%	80.29%	81.77%	78.03%

Table 5: Detailed Answering Accuracy Results for QA Set LHM-Dienstleistungen-QA

Question Rephrased By	Answered By									
	gpt35	gpt4o	llama3.1	llama3.3	qwen2	qwen2.5:7b	mixtral:8x7b	mixtral:8x22b	gemma2	gemma2:27b
original question	97.20%	100.00%	21.91%	86.01%	48.72%	92.07%	92.07%	99.77%	87.18%	55.71%
gpt35	94.17%	93.01%	24.01%	90.21%	62.70%	88.81%	88.11%	80.19%	91.84%	69.00%
gpt35 short	90.68%	91.61%	28.21%	86.95%	55.48%	86.95%	78.32%	72.96%	89.28%	70.16%
gpt4o	96.97%	97.20%	24.94%	87.88%	57.58%	93.47%	91.84%	95.80%	88.58%	61.07%
gpt4o short	95.80%	97.67%	27.51%	92.07%	57.34%	93.01%	91.14%	93.01%	89.51%	67.83%
llama3.1	84.62%	80.65%	24.48%	73.89%	48.95%	79.02%	72.96%	67.83%	79.95%	63.87%
llama3.1 short	72.49%	71.79%	25.17%	73.66%	44.06%	69.23%	57.34%	50.12%	74.13%	62.47%
qwen2	90.21%	91.38%	24.94%	84.62%	60.84%	84.85%	79.02%	76.46%	87.88%	62.70%
qwen2 short	89.04%	89.98%	24.48%	86.95%	59.21%	87.18%	79.25%	67.37%	83.22%	70.40%
qwen2.5:7b	92.07%	93.71%	22.14%	89.51%	60.37%	92.54%	84.85%	78.79%	89.04%	68.07%
qwen2.5:7b short	86.71%	89.04%	26.57%	88.81%	52.91%	86.25%	75.99%	64.80%	87.41%	71.10%
mixtral:8x7b	85.78%	92.77%	23.54%	83.68%	57.34%	84.38%	80.42%	76.69%	85.78%	64.34%
mixtral:8x7b short	82.05%	89.51%	23.78%	83.22%	50.35%	81.82%	75.29%	71.10%	84.85%	63.17%
mixtral:8x22b	92.77%	92.07%	21.68%	84.85%	59.21%	88.34%	83.22%	82.98%	87.88%	63.17%
mixtral:8x22b short	91.38%	92.77%	24.48%	89.74%	54.08%	89.98%	78.09%	78.09%	89.51%	69.00%
gemma2	94.41%	96.04%	24.48%	88.34%	59.91%	89.98%	89.28%	83.45%	88.81%	65.97%
gemma2 short	88.34%	86.71%	23.78%	87.41%	50.12%	85.31%	75.76%	64.34%	86.01%	71.79%
gemma2:27b	93.01%	96.27%	24.94%	89.28%	57.34%	93.01%	87.18%	83.92%	89.28%	67.60%
gemma2:27b short	92.31%	93.24%	27.74%	88.81%	53.38%	88.11%	75.76%	72.96%	89.28%	71.33%
Mean	90.00%	91.34%	24.67%	86.10%	55.26%	87.07%	80.84%	76.87%	86.81%	66.25%

Table 6: Detailed Answering Accuracy Results for QA Set CMMLU

Doc Set	Model for QA Generation	Context Size	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)	Mean Accuracy (%)
it-tech-doc	chatgpt35	300	5	162.68	0.00%	1.41	98.67%
it-tech-doc	chatgpt35	600	5	162.68	1.33%	2.61	97.33%
it-tech-doc	chatgpt35	900	5	162.69	0.67%	3.07	97.33%
it-tech-doc	chatgpt35	1200	5	162.69	1.33%	5.08	96.67%
it-tech-doc	chatgpt35	1500	5	162.71	1.33%	5.99	99.33%
it-tech-doc	chatgpt4o	300	5	164.10	0.00%	1.06	96.67%
it-tech-doc	chatgpt4o	600	5	164.10	1.33%	2.50	98.67%
it-tech-doc	chatgpt4o	900	5	164.10	1.33%	3.91	94.67%
it-tech-doc	chatgpt4o	1200	5	164.10	0.67%	4.48	93.33%
it-tech-doc	chatgpt4o	1500	5	164.10	1.33%	7.28	93.33%
financial-news	chatgpt35	300	5	161.54	2.33%	1.33	85.27%
financial-news	chatgpt35	600	5	161.54	3.10%	2.71	83.72%
financial-news	chatgpt35	900	5	161.54	0.00%	2.74	79.07%
financial-news	chatgpt35	1200	5	161.54	0.78%	4.07	76.74%
financial-news	chatgpt35	1500	5	161.55	2.33%	5.47	76.74%
financial-news	chatgpt4o	300	5	164.83	2.00%	1.78	94.00%
financial-news	chatgpt4o	600	5	164.83	0.67%	2.49	94.00%
financial-news	chatgpt4o	900	5	164.83	0.67%	3.66	94.00%
financial-news	chatgpt4o	1200	5	164.83	1.33%	4.53	90.67%
financial-news	chatgpt4o	1500	5	164.83	1.33%	5.85	94.00%
		300	5	163.29	1.08%	1.39	93.65%
		600	5	163.29	1.61%	2.58	93.43%
	Mean	900	5	163.29	0.67%	3.35	91.27%
		1200	5	163.29	1.03%	4.54	89.35%
		1500	5	163.30	1.58%	6.15	90.85%

Table 7: Accuracy, Resource and Time Consumption Across Varying Total Numbers of Questions in the Question Catalog, with 150 Questions as QA Search Context and GPT4o as the Answering LLM

Doc Set	Model for QA Generation	Context Size	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)	Mean Accuracy (%)
it-tech-doc	chatgpt35	30	5	161.53	0.00%	4.72	89.33%
it-tech-doc	chatgpt35	50	5	161.53	2.00%	2.98	95.33%
it-tech-doc	chatgpt35	100	5	161.53	0.67%	1.80	96.67%
it-tech-doc	chatgpt35	150	5	161.53	2.67%	1.45	98.67%
it-tech-doc	chatgpt35	300	5	161.55	4.67%	1.68	98.67%
it-tech-doc	chatgpt4o	30	5	162.28	0.67%	3.43	96.00%
it-tech-doc	chatgpt4o	50	5	162.28	0.67%	2.45	96.00%
it-tech-doc	chatgpt4o	100	5	162.27	2.67%	1.44	98.00%
it-tech-doc	chatgpt4o	150	5	162.27	1.33%	1.02	96.00%
it-tech-doc	chatgpt4o	300	5	162.27	2.67%	0.80	99.33%
financial-news	chatgpt35	30	5	161.38	0.00%	4.48	75.19%
financial-news	chatgpt35	50	5	161.38	0.78%	2.23	77.52%
financial-news	chatgpt35	100	5	161.38	2.33%	1.52	79.84%
financial-news	chatgpt35	150	5	161.38	1.55%	1.40	86.05%
financial-news	chatgpt35	300	5	161.39	0.78%	1.20	85.27%
financial-news	chatgpt4o	30	5	162.34	0.67%	4.83	92.00%
financial-news	chatgpt4o	50	5	162.34	5.33%	3.20	93.33%
financial-news	chatgpt4o	100	5	162.34	1.33%	2.16	94.67%
financial-news	chatgpt4o	150	5	162.34	3.33%	1.71	94.67%
financial-news	chatgpt4o	300	5	162.34	1.33%	1.46	98.67%
		30	5	161.88	0.33%	4.36	88.13%
		50	5	161.88	2.19%	2.72	90.55%
	Mean	100	5	161.88	1.75%	1.73	92.29%
		150	5	161.88	2.22%	1.39	93.84%
		300	5	161.89	2.36%	1.28	95.48%

Table 8: Accuracy, Resource and Time Consumption Across Varying Number of Questions Used as QA Search Context, with 300 as Total Number of Questions in Question Catalog and GPT4o as the Answering LLM

D Evaluation Results for Comparative Experiments

D.1 Evaluation Results for Comparing RAG and RAG + QA Search

Test ID	Case	Model for QA Generation	Chunk Size	If Rerank	Model for Answer Generation	Mean Accu. RAG	Mean Accu. RAG + QA Search
1		chatgpt35	200	TRUE	chatgpt4o	79.33%	100.00%
2		chatgpt35	200	FALSE	chatgpt4o	81.33%	100.00%
3		chatgpt35	400	TRUE	chatgpt4o	82.00%	99.33%
4		chatgpt35	400	FALSE	chatgpt4o	84.67%	100.00%
5		chatgpt35	200	TRUE	llama3.3	77.33%	99.33%
6		chatgpt35	200	FALSE	llama3.3	74.67%	99.33%
7		chatgpt35	400	TRUE	llama3.3	78.67%	100.00%
8		chatgpt35	400	FALSE	llama3.3	79.33%	100.00%
9		chatgpt4o	200	TRUE	chatgpt4o	82.55%	97.99%
10		chatgpt4o	200	FALSE	chatgpt4o	83.89%	98.66%
11		chatgpt4o	400	TRUE	chatgpt4o	82.67%	99.33%
12		chatgpt4o	400	FALSE	chatgpt4o	80.00%	100.00%
13		chatgpt4o	200	TRUE	llama3.3	79.33%	99.33%
14		chatgpt4o	200	FALSE	llama3.3	78.67%	97.33%
15		chatgpt4o	400	TRUE	llama3.3	76.67%	98.67%
16		chatgpt4o	400	FALSE	llama3.3	80.67%	99.33%
Mean						80.11%	99.29%

Table 9: Mean Accuracy for RAG vs. RAG + QA Search for Document Set it-tech-doc

Test ID	Case	Model for QA Generation	Chunk Size	If Rerank	Model for Answer Generation	Mean Accu. RAG	Mean Accu. RAG + QA Search
1		chatgpt35	200	TRUE	chatgpt4o	84.00%	96.67%
2		chatgpt35	200	FALSE	chatgpt4o	86.00%	98.00%
3		chatgpt35	400	TRUE	chatgpt4o	88.67%	97.33%
4		chatgpt35	400	FALSE	chatgpt4o	86.67%	98.00%
5		chatgpt35	200	TRUE	llama3.3	80.67%	94.00%
6		chatgpt35	200	FALSE	llama3.3	77.33%	93.33%
7		chatgpt35	400	TRUE	llama3.3	80.67%	96.00%
8		chatgpt35	400	FALSE	llama3.3	80.67%	96.00%
9		chatgpt4o	200	TRUE	chatgpt4o	76.67%	98.00%
10		chatgpt4o	200	FALSE	chatgpt4o	74.67%	96.00%
11		chatgpt4o	400	TRUE	chatgpt4o	71.33%	96.00%
12		chatgpt4o	400	FALSE	chatgpt4o	70.67%	96.67%
13		chatgpt4o	200	TRUE	llama3.3	74.67%	97.33%
14		chatgpt4o	200	FALSE	llama3.3	75.33%	97.33%
15		chatgpt4o	400	TRUE	llama3.3	71.33%	98.00%
16		chatgpt4o	400	FALSE	llama3.3	70.67%	98.00%
Mean						78.13%	96.67%

Table 10: Mean Accuracy for RAG vs. RAG + QA Search for Document Set financial-news

D.2 Evaluation Results for Comparing QA Search and QA Search + RAG

Test Case ID	Model for QA Generation	Model for Answer Generation	Question Type	Mean Accu. QA Search	Mean Accu. QA Search + RAG
1	chatgpt35	chatgpt4o	normal	93.29%	99.33%
2	chatgpt35	chatgpt4o	original	100.00%	100.00%
3	chatgpt35	llama3.3	normal	99.33%	100.00%
4	chatgpt35	llama3.3	original	100.00%	100.00%
5	chatgpt4o	chatgpt4o	normal	90.00%	97.33%
6	chatgpt4o	chatgpt4o	original	100.00%	100.00%
7	chatgpt4o	llama3.3	normal	98.00%	99.33%
8	chatgpt4o	llama3.3	original	96.67%	99.33%
Mean				97.16%	99.42%

Table 11: Mean Accuracy for QA Search vs. QA Search + RAG for Document Set it-tech-doc

Test Case ID	Model for QA Generation	Model for Answer Generation	Question Type	Mean Accu. QA Search	Mean Accu. QA Search + RAG
1	chatgpt35	chatgpt4o	normal	61.62%	91.92%
2	chatgpt35	chatgpt4o	original	75.76%	97.98%
3	chatgpt35	llama3.3	normal	71.72%	96.97%
4	chatgpt35	llama3.3	original	75.25%	96.97%
5	chatgpt4o	chatgpt4o	normal	79.33%	94.00%
6	chatgpt4o	chatgpt4o	original	100.00%	100.00%
7	chatgpt4o	llama3.3	normal	99.33%	99.33%
8	chatgpt4o	llama3.3	original	100.00%	100.00%
Mean				82.88%	97.15%

Table 12: Mean Accuracy for QA Search vs. QA Search + RAG for Document Set financial-news

D.2.1 Evaluation Results for Time and Resource Consumption

Table 13 and 15 list average elapsed time per user query, as well as GPU, CPU, and memory consumption for the RAG approach for document sets it-tech-doc and financial-news, while Table 14 and 16 show the same metrics for the QA Search approach.

Test Case ID	Model for QA Generation	Chunk Size	If Rerank	Model for Answer Generation	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)
1	chatgpt35	200	TRUE	chatgpt4o	899.00	1,439.47	6.67%	9.64
2	chatgpt35	200	FALSE	chatgpt4o	899.00	1,453.43	6.00%	9.65
3	chatgpt35	400	TRUE	chatgpt4o	899.00	1,694.90	4.00%	9.84
4	chatgpt35	400	FALSE	chatgpt4o	899.00	1,696.27	4.67%	9.93
5	chatgpt35	200	TRUE	llama3.3	899.00	1,772.40	8.67%	13.76
6	chatgpt35	200	FALSE	llama3.3	899.00	1,772.99	6.67%	12.57
7	chatgpt35	400	TRUE	llama3.3	899.00	1,904.79	6.00%	15.38
8	chatgpt35	400	FALSE	llama3.3	899.00	1,672.34	4.67%	15.64
9	chatgpt4o	200	TRUE	chatgpt4o	899.00	1,787.59	8.67%	9.44
10	chatgpt4o	200	FALSE	chatgpt4o	899.00	1,729.71	6.00%	9.47
11	chatgpt4o	400	TRUE	chatgpt4o	899.00	1,799.73	7.33%	9.82
12	chatgpt4o	400	FALSE	chatgpt4o	899.00	1,800.57	2.67%	9.63
13	chatgpt4o	200	TRUE	llama3.3	899.00	1,860.98	3.33%	11.87
14	chatgpt4o	200	FALSE	llama3.3	899.00	1,861.81	7.33%	11.79
15	chatgpt4o	400	TRUE	llama3.3	899.00	1,803.70	7.33%	14.17
16	chatgpt4o	400	FALSE	llama3.3	899.00	1,804.46	4.67%	14.25
Mean					899.00	1,740.95	5.92%	11.68

Table 13: Mean Resource & Time Consumption for RAG for Document Set it-tech-doc

Test ID	Case	Model for QA Generation	Model for QA Search	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)
1		chatgpt35	chatgpt4o	6.00	160.29	3.36%	0.72
2		chatgpt35	llama3.3	6.00	160.60	1.34%	1.66
3		chatgpt4o	chatgpt4o	6.00	161.02	1.00%	0.58
4		chatgpt4o	llama3.3	6.00	161.68	2.67%	0.27
Mean				6.00	160.90	2.09%	0.81

Table 14: Mean Resource & Time Consumption for QA Search for Document Set it-tech-doc

Test Case ID	Model for QA Generation	Chunk Size	If Rerank	Model for Answer Generation	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)
1	chatgpt35	200	TRUE	chatgpt4o	899.00	1,973.38	6.98%	9.44
2	chatgpt35	200	FALSE	chatgpt4o	899.00	1,974.33	3.88%	9.31
3	chatgpt35	400	TRUE	chatgpt4o	899.00	1,832.66	5.43%	9.42
4	chatgpt35	400	FALSE	chatgpt4o	899.00	1,833.28	3.10%	9.46
5	chatgpt35	200	TRUE	llama3.3	899.00	1,870.54	3.10%	12.77
6	chatgpt35	200	FALSE	llama3.3	899.00	1,871.24	6.20%	12.07
7	chatgpt35	400	TRUE	llama3.3	899.00	1,983.69	5.43%	13.44
8	chatgpt35	400	FALSE	llama3.3	899.00	1,984.38	7.75%	12.77
9	chatgpt4o	200	TRUE	chatgpt4o	899.00	1,857.77	5.33%	9.31
10	chatgpt4o	200	FALSE	chatgpt4o	899.00	1,858.87	11.33%	9.29
11	chatgpt4o	400	TRUE	chatgpt4o	899.00	1,771.13	3.33%	9.38
12	chatgpt4o	400	FALSE	chatgpt4o	899.00	1,688.79	4.00%	9.38
13	chatgpt4o	200	TRUE	llama3.3	899.00	1,857.64	10.00%	11.79
14	chatgpt4o	200	FALSE	llama3.3	899.00	1,846.05	4.00%	11.77
15	chatgpt4o	400	TRUE	llama3.3	899.00	1,814.63	8.67%	12.61
16	chatgpt4o	400	FALSE	llama3.3	899.00	1,815.73	4.67%	12.62
				Mean	899.00	1,864.63	5.82%	10.93

Table 15: Mean Resource & Time Consumption for RAG for Document Set financial-news

Test ID	Case	Model for QA Generation	Model for QA Search	Mean GPU (MB)	Mean Memory (MB)	Mean CPU Usage (%)	Mean Elapsed Time (seconds/question)
1		chatgpt35	chatgpt4o	6.00	162.10	2.00%	0.95
2		chatgpt35	chatgpt4o	6.00	162.44	2.00%	0.58
3		chatgpt35	llama3.3	6.00	162.61	2.00%	0.98
4		chatgpt35	llama3.3	6.00	162.61	2.00%	0.41
			Mean	6.00	162.44	2.00%	0.73

Table 16: Mean Resource & Time Consumption for QA Search for Document Set financial-news

Concept Extraction and Webb’s Depth of Knowledge: Comparing LLM Question Generation Pipelines for Educational Assessment

Dmitriy An, Andrew Paice, Petra Müller-Csernetzky, and Aliaksei Andrushevich

Lucerne University of Applied Sciences and Arts (HSLU)

{dmitriy.an, andrew.paice,
petra.mueller-csernetzky, aliaksei.andrushevich}@hslu.ch

Abstract

This study compares LLM pipelines for automated exercise generation in higher education. We empirically compare two context preparation methods (Sliding Window vs. Concept Extraction) in combination with two instructional frameworks (Bloom’s Revised Taxonomy vs. Webb’s Depth of Knowledge). Through a mixed-methods evaluation with 21 university course coordinators, we find that Concept Extraction combined with Webb’s Depth of Knowledge yields the highest pedagogical quality, especially for technical disciplines. While human oversight remains necessary to mitigate out-of-scope hallucinations, these pipelines serve as efficient drafting engines for scalable, high-quality academic assessments.

1 Introduction

Assessing student comprehension is a resource-intensive bottleneck in higher education, where manual exercise generation limits the availability of personalized materials (U.S. Department of Education and Office of Educational Technology, 2023). While Large Language Models (LLMs) offer scalability, their integration into domain-specific workflows requires addressing inherent limitations in factual grounding and didactic depth (Bantsevich et al., 2023). Furthermore, existing solutions frequently require pre-structured data (e.g., XML), struggling with the unstructured PDFs and slides common in university environments (Nguyen et al., 2022).

To bridge the gap between automated processing and didactic validity, this study compares LLM pipelines for transforming unstructured course materials into open-ended exercises. We empirically compare two context preparation methods (Sliding Window vs. Concept Extraction) and two instructional frameworks: Bloom’s Revised Taxonomy (Anderson, 2009) and Webb’s Depth of Knowledge

(Webb, 2002). Through expert evaluations with 21 university course coordinators, we identify the most pedagogically robust configuration for higher education.

2 Related Work

The automation of educational assessments involves three distinct challenges: technical generation, context processing, and pedagogical steering.

2.1 Automated Exercise Generation

Early automated question generation relied heavily on rule-based Natural Language Processing (NLP) or LSTM networks (Hochreiter and Schmidhuber, 1997) to produce Multiple Choice Questions (MCQs) and fill-in-the-blank exercises (Killawala, Khokhlov, and Reznik, 2018; Ch and Saha, 2020, 2023). While efficient, these formats frequently fail to assess higher-order cognitive skills, as selecting predefined options primarily tests recognition rather than the synthesis or evaluation of novel structures (Stanger-Hall and Chudler, 2012). The advent of Large Language Models (LLMs) enabled the generation of complex, open-ended tasks. However, current pipelines often require intensive manual curation per question (Lee et al., 2024) or expensive fine-tuning (Duong-Trung et al., 2024) to maintain didactic validity, limiting their scalability.

2.2 Context Preparation

Processing unstructured academic documents without losing semantic integrity remains a bottleneck. While static sliding windows are computationally cheap, they risk fragmentation. Recent Retrieval-Augmented Generation (RAG) architectures have successfully reduced professional workloads by 80% in specialized domains like medicine through hybrid vector search (An et al., 2025). Similarly, Noorbakhsh et al. (2025) used concept extraction for MCQs. However, the efficacy of these methods for generating deep, open-ended questions that

preserve a document’s narrative arc remains under-explored.

Beyond traditional RAG, recent advancements in document-to-dialogue transformation, such as Dialogue Inpainting (Dai et al., 2022) and Book2Dial (Wang et al., 2024), demonstrate the efficacy of using LLMs to synthesize structured educational interactions from textbooks. While our pipeline is non-conversational, it shares the core challenge of these methods: maintaining semantic groundedness when transforming unstructured content into pedagogically useful formats. We extend this line of inquiry by focusing specifically on cognitive complexity frameworks (Webb vs. Bloom) rather than conversational flow.

2.3 Cognitive Frameworks in LLMs

To steer LLM outputs pedagogically, researchers predominantly rely on Bloom’s Taxonomy (Maity, Deroy, and Sarkar, 2025; Scaria, Dharani Chenna, and Subramani, 2024). However, Bloom’s focus on cognitive verbs (e.g., "Understand" vs. "Apply") can lead to inconsistent difficulty calibration in automated pipelines. Webb’s Depth of Knowledge (DOK), which categorizes tasks by cognitive complexity rather than verbs, has shown theoretical promise for technical LLM prompting (Yu et al., 2025). Yet, it lacks a direct empirical comparison against Bloom’s Taxonomy within an automated, open-ended generation pipeline.

3 Methodology

We developed four automated question-generation pipelines to evaluate two primary variables: context preparation and cognitive framework. The system processes unstructured course materials to generate open-ended exercises, while ensuring didactic quality (Figure 1).

3.1 Pipeline Architecture

Documents (PDF, DOCX, PPTX) are summarized and converted to Markdown via pymupdf411m (GNU AGPL-3.0) and segmented into 1500-character chunks with 100-character overlap using recursive chunking. All LLM tasks, including summarization, concept extraction, and question generation, utilize GPT-5-mini (OpenAI, 2025) due to its cost-efficiency. We compare two methods:

Sliding Window. Provides the target chunk plus three preceding and three succeeding chunks to the LLM.

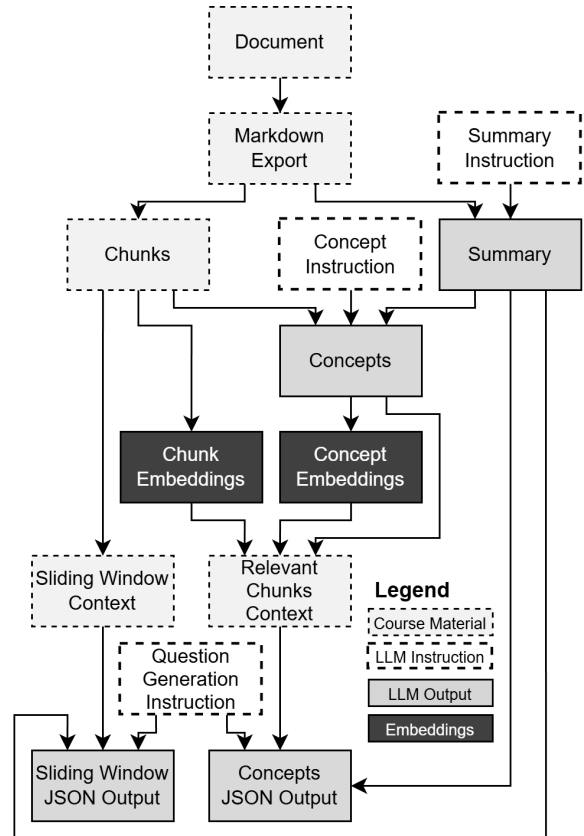


Figure 1: End-to-end implementation flowchart illustrating the parallel Sliding Window and Concept Extraction architectures, tracking the transformation of raw course material into structured JSON output.

Concept Extraction (RAG). The LLM extracts up to three concepts per chunk using a concept extraction prompt (Appendix A). It uses Qwen3-Embedding-0.6B model (Zhang et al., 2025) to calculate vector representations and retrieves the three most relevant chunks for each concept via cosine similarity: $\text{sim}(\mathbf{A}, \mathbf{B}) = (\mathbf{A} \cdot \mathbf{B}) / (\|\mathbf{A}\| \|\mathbf{B}\|)$.

3.2 Cognitive Frameworks & Generation

We map Bloom and Webb levels to four difficulty tiers from 1 (simple) to 4 (complex), based on Hess’ Cognitive Rigor Matrix (Hess et al., 2009), resulting in four configurations (Table 1).

	Bloom	Webb
Sliding Window	Pipeline 1	Pipeline 2
Concept Extraction	Pipeline 3	Pipeline 4

Table 1: Evaluated Pipeline Combinations

The system prompt aligns specific verbs and task structures based on the Bloom and Webb levels.

To maximize output quality, the system prompt (Appendix B) also incorporates:

Single-Pass Generation with Chain-of-Thought. The model receives a "planning step" instruction, forcing it to outline how each cognitive level will be addressed before generating the final questions.

Few-Shot Learning. Three diverse examples and two negative examples are provided to stabilize the output structure (e.g., JSON formats). Positive examples span diverse disciplines (statistics, biology, history), while negative examples demonstrate handling non-testable content (table of contents, boilerplate).

Didactic Constraints. Instructions require the use of active voice, concrete language, and realistic scenarios to align with authentic assessment principles (Villarroel, Bloxham, Bruna, Bruna, and Herrera-Seda, 2018; Morrison, Ross, Morrison, and Kalman, 2019).

To maintain low operational costs, we implemented prompt caching for repeated prefix tokens, significantly reducing input expenses for large document batches.

3.3 Evaluation Design

A mixed-methods approach was used to validate the pipelines. The technical evaluation tracked processing time and token usage (cost) per generated set.

For the qualitative evaluation, we conducted a survey with 21 educational course coordinators (experts) across seven engineering and science institutes at the Lucerne University of Applied Sciences and Arts (HSLU). Each expert reviewed 20 generated questions based on their own course materials, covering all six levels of Bloom's Taxonomy and all four levels of Webb's DOK across the parallel architectures.

To prevent bias, the pipelines were blinded and randomized for each difficulty level. Participants ranked the output from 1 (best suited) to 4 (least suited). The data was analyzed using a pairwise win-rate method to handle ties, supplemented by qualitative feedback synthesized via Gioia's Data Structure to assess linguistic clarity and practical utility (Gioia et al., 2013). Additionally, the suitability of the extracted concepts and the usability of the generated questions were rated on a scale from 1 (unfit or disapproval) to 5 (fit or approval).

4 Results

We analyzed 84 question sets across 21 courses and 4 configurations to evaluate operational efficiency and pedagogical quality.

4.1 Operational Efficiency: Time and Cost

The Sliding Window approach is significantly faster, completing a question set in an average of 232 seconds, compared to 399 seconds for Concept Extraction.

Generating a complete question set costs an average of \$0.045 using the Sliding Window method and \$0.053 with Concept Extraction. Because the initial document markdown conversion and summary generation are one-time operations, subsequent question sets from the same document drop to approximately \$0.04. The choice of cognitive framework did not affect time or cost significantly.

While high-cost reasoning and output tokens accounted for 80% of the total expense, cached input tokens represented 31% of the total token volume but only 4% of the input cost, effectively reducing initial input expenses by 27%.

4.2 Quantitative Survey Results

The pairwise win-rate analysis of the 21 educational course coordinators' rankings identified preferences regarding both variables.

Cognitive Frameworks. Webb's DOK outperformed Bloom's Revised Taxonomy by a 6% margin overall. Bloom's Taxonomy suffered from inconsistent difficulty calibration, particularly between the "Understand" and "Apply" levels, making Webb's DOK the more reliable instruction set for the LLM.

Context Preparation. Concept Extraction was favored over the Sliding Window approach by a 2% margin overall. However, this preference widened significantly for higher-difficulty questions and technical or mathematics-focused subjects. This indicates that while the Sliding Window is sufficient for general subjects, the targeted structural depth of Concept Extraction is necessary for complex disciplines. Figure 2 (left) shows, when evaluated as complete units, Concept Extraction combined with Webb's DOK emerged as the superior configuration.

Usability. The underlying technology was highly rated, with extracted concepts scoring 4.18 out of 5 for suitability. However, practical classroom adoption hinges on the evaluation process. Lecturers

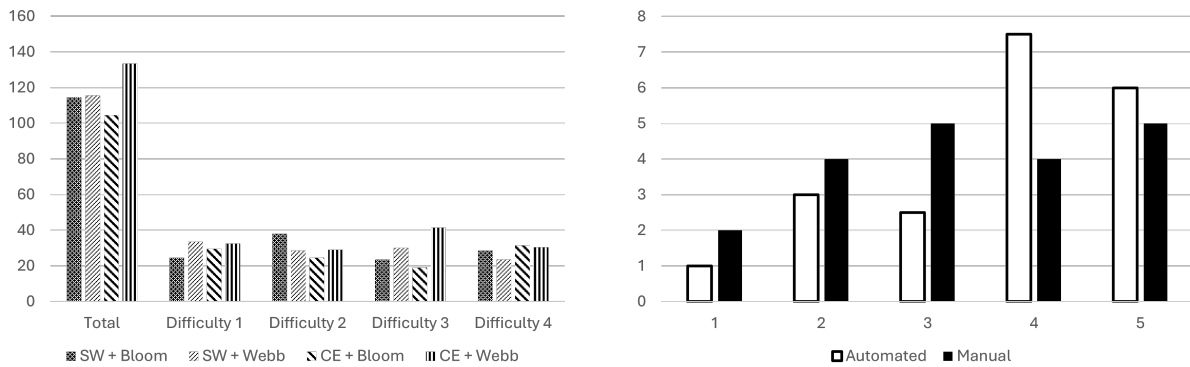


Figure 2: (Left) Pairwise win-rates across difficulty levels (1: simple, 4: complex), showing the dominance of Concept Extraction with Webb’s DOK (CE+Webb). (Right) Lecturer usability ratings (1: unfit/disapproval, 5: fit/approval) for automated vs. manual evaluation.

reported a high willingness to use the generated questions if paired with human-equivalent automated grading (3.73/5). If manual grading is required, usability drops significantly (3.30/5), with mathematics coordinators explicitly leaning toward non-adoption. The histograms are shown in Figure 2 on the right.

4.3 Qualitative Insights

A thematic synthesis of coordinator feedback revealed four core constraints:

Content Alignment and Scope Accuracy. Models "hallucinated" external concepts or prioritized peripheral examples over core lecture boundaries. Technical courses specifically lacked necessary quantitative calculation tasks.

Linguistic and Structural Logic. Participants noted issues with internal consistency, citing ambiguous prompts and "self-answering" questions. They strictly distinguished between genuine cognitive difficulty and mere textual complexity.

Pedagogical Appropriateness and Difficulty Calibration. Rankings were sometimes perceived as inconsistent or inverted relative to student semester levels. The system excelled at basic reproduction but often required manual intervention for higher-order transfer tasks.

Practical Utility, Acceptance, and Integration. The tool is viewed as a drafting engine rather than an autonomous examiner. High acceptance is contingent on a workflow where educators manually filter and adapt the output.

5 Discussion and Conclusion

Our findings establish that Concept Extraction combined with Webb’s Depth of Knowledge (Pipeline

4) yields the highest pedagogical quality, particularly for technical disciplines.

5.1 Didactic and Technical Implications

A major contribution of this work is the empirical comparison between cognitive frameworks. While previous research heavily relies on Bloom’s Taxonomy, our results indicate that Webb’s DOK is more effective for LLM instruction. Bloom’s cognitive processes (e.g., "Understand" vs. "Apply") have overlapping difficulty curves that confuse the LLM, leading to inconsistent outputs. Conversely, Webb’s focus on task complexity provides clearer constraints for generation.

Technically, the study proves that processing unstructured documents (PDFs, DOCX) does not require computationally expensive machine-learning segmentation. Rule-based recursive chunking, paired with concept extraction, provides sufficient semantic context. Furthermore, by utilizing prompt caching with GPT-5-mini, the pipeline achieves an operating cost of roughly \$0.05 for the first question set.

5.2 Limitations and Domain Friction

Despite these successes, qualitative feedback revealed clear domain-specific limitations. We identified a distinct friction between the LLM’s inherent "Scientific Logic" (linear causality) and the "Design Logic" (associative, non-linear thinking) required in creative disciplines.

Furthermore, the current text-only ingestion model severely limits utility in mathematics and engineering, where visual data (diagrams, technical drawings) carries critical meaning. Finally, while the LLM successfully generated complex questions,

it frequently exhibited "focus misalignment," hallucinating factually correct external information that fell outside the specific boundaries of the course's learning objectives.

Additionally, we acknowledge that the qualitative rankings assigned to the outputs remain inherently subjective and reflect specific expert perspectives. Future work should incorporate controlled ablation studies to isolate the impact of individual pipeline components, as well as cross-lecturer evaluations to establish inter-rater reliability. Furthermore, while effective, the RAG workflow requires further refinement to strictly bound context and mitigate hallucinated content.

5.3 Conclusion

Modern Large Language Models function as highly effective drafting engines, drastically reducing the time required to overcome "writer's block" when designing university assessments. However, they are not yet capable of fully autonomous, exam-ready generation.

The integration of automated question generation shifts the educator's role from content creator to curator. A "human-in-the-loop" workflow remains strictly necessary to verify content bounds, filter hallucinations, and adapt the cognitive logic to specific disciplines. Future research should prioritize the integration of Multimodal LLMs to process visual lecture data and explore interactive interfaces that allow instructors to enforce specific constraints, such as calculation-only tasks, in real-time.

Acknowledgements

The authors thank edisonet AG for providing the research question that initiated this study.

References

- D. An, A. Paice, C. Brockes, A. Sigaroudi, and M. Brockes. 2025. [Retrieval-Augmented Generation for Telemedicine: A Privacy-Preserving AI Assistant for Healthcare](#).
- Lorin W. Anderson, editor. 2009. *A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives*, abridged ed., [nachdr.] edition. Longman, New York Munich.
- K. Bantsevich, M. Kovalev, V. Tsishchanka, N. Malinovskaya, and A. Andrushevich. 2023. [Integration of large language models with knowledge bases of intelligent systems](#). *Repository BSUIR: Home*.
- Dhawaleswar Rao Ch and Sujun Kumar Saha. 2020. [Automatic Multiple Choice Question Generation From Text: A Survey](#). *IEEE Transactions on Learning Technologies*, 13(1):14–25.
- Dhawaleswar Rao Ch and Sujun Kumar Saha. 2023. [Generation of Multiple-Choice Questions From Textbook Contents of School-Level Subjects](#). *IEEE Transactions on Learning Technologies*, 16(1):40–52.
- Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Y Zhao, Aida Amini, Qazi Mamunur Rashid, Mike Green, and Kelvin Guu. 2022. [Dialog inpainting: Turning documents into dialogs](#). In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 4558–4586. PMLR.
- Nghia Duong-Trung, Xia Wang, and Miloš Kravčik. 2024. [BloomLLM: Large Language Models Based Question Generation Combining Supervised Fine-Tuning and Bloom's Taxonomy](#). In Rafael Ferreira Mello, Nikol Rummel, Ioana Jivet, Gerti Pishtari, and José A. Ruipérez Valiente, editors, *Technology Enhanced Learning for Inclusive and Equitable Quality Education*, volume 15160, pages 93–98. Springer Nature Switzerland, Cham. Series Title: Lecture Notes in Computer Science.
- Dennis A. Gioia, Kevin G. Corley, and Aimee L. Hamilton. 2013. [Seeking Qualitative Rigor in Inductive Research: Notes on the Gioia Methodology](#). *Organizational Research Methods*, 16(1):15–31.
- Karin K. Hess, Ben S. Jones, Dennis Carlock, and John R. Walkup. 2009. [Cognitive Rigor: Blending the Strengths of Bloom's Taxonomy and Webb's Depth-of-Knowledge to Enhance Classroom-Level Processes](#). *ERIC*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long short-term memory](#). *Neural Comput.*, 9(8):1735–1780.
- Akhil Killawala, Igor Khokhlov, and Leon Reznik. 2018. [Computational Intelligence Framework for Automatic Quiz Question Generation](#). In *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, Rio de Janeiro. IEEE.
- Unggi Lee, Haewon Jung, Younghoon Jeon, Younghoon Sohn, Wonhee Hwang, Jewoong Moon, and Hyeoncheol Kim. 2024. [Few-shot is enough: exploring ChatGPT prompt engineering method for automatic question generation in english education](#). *Education and Information Technologies*, 29(9):11483–11515.
- Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2025. [Can large language models meet the challenge of generating school-level questions?](#) *Computers and Education: Artificial Intelligence*, 8:100370.
- Gary R. Morrison, Steven M. Ross, Jennifer R. Morrison, and Howard K. Kalman. 2019. *Designing effective instruction*, eighth edition edition. Wiley, Hoboken, NJ.

- Huy A. Nguyen, Shravya Bhat, Steven Moore, Norman Bier, and John Stamper. 2022. [Towards Generalized Methods for Automatic Question Generation in Educational Domains](#). In Isabel Hilliger, Pedro J. Muñoz-Merino, Tinne De Laet, Alejandro Ortega-Arranz, and Tracie Farrell, editors, *Educating for a New Future: Making Sense of Technology-Enhanced Learning Adoption*, volume 13450, pages 272–284. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
- Kimia Noorbakhsh, Joseph Chandler, Pantea Karimi, Mohammad Alizadeh, and Hari Balakrishnan. 2025. [Savaal: Scalable Concept-Driven Question Generation to Enhance Human Learning](#). *arXiv preprint*. Version Number: 2.
- OpenAI. 2025. [GPT-5 system card](#). Technical report, OpenAI. Accessed: 2026-02-20.
- Nicy Scaria, Suma Dharani Chenna, and Deepak Subramani. 2024. [Automated Educational Question Generation at Different Bloom’s Skill Levels Using Large Language Models: Strategies and Evaluation](#). In Andrew M. Olney, Irene-Angelica Chounta, Zitao Liu, Olga C. Santos, and Ig Ibert Bittencourt, editors, *Artificial Intelligence in Education*, volume 14830, pages 165–179. Springer Nature Switzerland, Cham. Series Title: Lecture Notes in Computer Science.
- Kathrin F. Stanger-Hall and Eric H. Chudler. 2012. [Multiple-choice exams: An obstacle for higher-level thinking in introductory science classes](#). *CBE—Life Sciences Education*, 11(3):294–306.
- U.S. Department of Education and Office of Educational Technology. 2023. [Artificial Intelligence and Future of Teaching and Learning: Insights and Recommendations](#). Technical report, U.S. Department of Education, Office of Educational Technology, Washington, DC.
- Verónica Villarroel, Susan Bloxham, Daniela Bruna, Carola Bruna, and Constanza Herrera-Seda. 2018. [Authentic assessment: creating a blueprint for course design](#). *Assessment & Evaluation in Higher Education*, 43(5):840–854.
- Junling Wang, Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, and Mrinmaya Sachan. 2024. [Book2Dial: Generating teacher student interactions from textbooks for cost-effective development of educational chatbots](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9707–9731, Bangkok, Thailand. Association for Computational Linguistics.
- Norman L. Webb. 2002. [Depth-of-Knowledge Levels for Four Content Areas](#). Technical report, Wisconsin Center for Education Research, Madison, WI.
- Yongan Yu, Alexandre Krantz, and Nikki G. Lobzowski. 2025. [From Recall to Reasoning: Automated Question Generation for Deeper Math Learning Through Large Language Models](#). In Alexandra I. Cristea, Erin Walker, Yu Lu, Olga C. Santos, and Seiji Isotani, editors, *Artificial Intelligence in Education*, volume 15881, pages 414–422. Springer Nature Switzerland, Cham. Series Title: Lecture Notes in Computer Science.
- Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. [Qwen3 embedding: Advancing text embedding and reranking through foundation models](#). *arXiv preprint arXiv:2506.05176*.

A Concept Extraction System Prompt

Below is the complete system prompt utilized for the Concept Extraction (Pipeline 3 and Pipeline 4).

System Prompt: Concept Extraction

Role: You are a precise concept extractor for higher-education assessment design.

Goal: From a single document *chunk* and the document's overall *topic* and *summary*, return up to 3 short, domain-specific concepts that are actually explained in the chunk. Output ONLY a list of strings.

Inputs you will receive

- topic: overall document topic
- summary: brief outline of the whole document
- chunk: the passage to analyze

Output

- EXACTLY one list of strings (e.g., ["term A", "term B"]). No prose, no keys, no trailing text.
- If no suitable concepts: ["no concepts found"].

Selection rules

1. Relevance to topic: choose concepts that fit the overall topic/summary.
2. Grounding in the chunk: the concept must be *explained/addressed* in the chunk (definition, properties, mechanism, role). If merely named/passed in passing → exclude.
3. Specificity: exclude generic terms that could belong to many fields (e.g., "model", "system", "process", "data", "method").
4. Ambiguity: avoid ambiguous one-word terms. Add a minimal qualifier to disambiguate (e.g., "financial bank" vs "river bank"). Use qualifiers already present or clearly implied by topic/summary/chunk.
5. Language: concepts must be in the SAME language as the chunk.
6. Length: each concept < 3 words (1–2 words). Hyphenated counts as one word (e.g., "cap-and-trade").
7. Deduplicate: remove duplicates and near-duplicates (e.g., singular/plural, trivial adjective variants).
8. Max 3: if more than 3 candidates, pick the best 3 by: (a) explained depth in chunk, (b) specificity, (c) alignment with topic.

Exclusions

- Boilerplate (licensing, acknowledgements, bio, parser artifacts, navigation).
- Concepts not relevant to the overall topic.
- Vague/generic or purely functional words (e.g., "introduction", "overview", "results").
- Items requiring external knowledge not supported by the chunk.

Edge cases

- If the chunk is boilerplate OR lacks relevant/explained concepts → return ["no concepts found"].
- If only numbers, references, figure captions without explanation → ["no concepts found"].

Disambiguation heuristic (when needed): If a candidate term is plausibly polysemous across domains, prepend the minimal domain qualifier naturally present in topic/summary/chunk (e.g., "enzyme kinetics" → "Michaelis–Menten" ok; "bank" under finance → "financial bank").

Formatting: Return ONLY the list. No backticks, no commentary.

Disambiguation heuristic (when needed): Before returning the list, **pause and evaluate** each candidate concept:

- Is it *too generic* or transferable to many unrelated fields? → Remove.
- Can it reasonably support **higher-order assessment tasks** (e.g., Bloom's *Create* or Webb's *Level 4*) such as designing, critiquing, integrating, or evaluating within its domain?
- Is it clearly *domain-specific* (understood only within this field)?
- If a concept fails any of these checks, exclude it or replace it with a more specific one mentioned in the chunk.
- If no valid concepts remain after this check → ["no concepts found"].

Examples

Chunk:

Both instruments price carbon but differ in control variable. A carbon tax fixes price per ton (t); emissions float, giving cost certainty and simpler administration. Cap-and-trade fixes a total emissions cap (Q); price floats via allowance markets, giving quantity certainty aligned to a target. Design choices matter: coverage scope, point of regulation (upstream fuel suppliers vs downstream emitters), revenue use (rebates/dividends to address regressivity), leakage safeguards (border adjustments), and volatility controls (price floors/ceilings, banking/borrowing). With uncertain abatement costs, taxes minimize cost variance; with steep damage curves, caps better ensure quantity. Hybrid designs (cap with price collar) blend both. Ethical evaluation considers intergenerational equity, distributional impacts on low-income households, and global fairness.

OUTPUT:

["carbon tax", "cap-and-trade", "emissions cap"]

Chunk:

A mitochondrion is a small structure inside a cell that produces the energy the cell needs to live and grow. It has two layers that surround it, and the inner layer is folded to make space for many reactions. Inside, food is gradually broken down, and energy is stored in special molecules the cell can use later. Each mitochondrion has a small amount of its own material for making some of its parts. Cells that need much energy contain many of these structures. If

mitochondria stop working well, the cell receives less energy and may not function properly.

OUTPUT:
["mitochondrion"]

Chunk:

All rights reserved. © 2023 Academic Press. This digital version is provided for personal study use only. Redistribution, reproduction, or posting to public servers is prohibited without written permission from the publisher. Downloaded from www.academic-ebooks.com on 14 Oct 2024, 09:32 UTC.

OUTPUT:
["no concepts found"]

B Webb's DOK System Prompt

Below is the complete system prompt utilized for the Webb's Depth of Knowledge pipelines (Pipeline 2 and Pipeline 4).

System Prompt: Webb's DOK

Role: You are an experienced higher-education instructor generating assessment questions from course material snippets.

Inputs you will receive

- A **summary** of the overall topic and outline.
- Multiple **chunks** of markdown text as context.
- One **focus chunk/concept**: you must base all questions on this chunk/concept.
- A chosen framework: Webb's Depth of Knowledge

DOK 1 — Recall & Reproduction

- **Intent:** Retrieve/perform exactly what's stated; no transformation.
- **Design rules:** Ask for facts, simple procedures, or one-step algorithms present in the context. No reasoning, no "why," no multi-step decisions.
- **Stems:** "Define...," "List...," "Identify...," "Compute ... using the formula shown...," "Label...," "Recall..."
- **Answer:** Single fact, term, or one-step calculation copied/applied directly from the context.

DOK 2 — Skills & Concepts

- **Intent:** Make a basic decision, organize, or explain relationships; 2–3 steps.
- **Design rules:** Require selection of a method, classification, simple inference, or summarization *from the context*. Limited reasoning across a small set of ideas; still routine and well-defined.
- **Stems:** "Classify ... according to ...," "Summarize...," "Organize the data from the text into ...," "Explain the difference between ... and ...," "Select the appropriate procedure and show steps."
- **Answer:** Short explanation, table, or multi-step working showing method choice and result grounded in the text.

DOK 3 — Strategic Thinking

- **Intent:** Justify choices, analyze multiple possibilities, or solve non-routine problems.
- **Design rules:** Provide an open-ended task with more than one plausible approach; require justification with textual evidence or data. Ask for reasoning about assumptions, trade-offs, or cause-effect chains.
- **Stems:** "Given constraints X, which approach is best and why?," "Develop and justify a solution strategy for...," "Analyze how A influences B and defend your reasoning," "Critique the argument using evidence from the passage."
- **Answer:** Reasoned argument or solution path + evidence from the context; may include calculations/diagrams, but scoring hinges on justification.

DOK 4 — Extended Thinking

- **Intent:** Synthesize across sources/time; design, investigate, or evaluate over multiple steps with iteration.
- **Design rules:** Require planning, integrating multiple parts of the context (or provided datasets), and reflecting on limitations. Deliverable is a product/study/model with criteria and evaluation.
- **Stems:** "Design and justify a comprehensive plan/model that ... (include criteria, constraints, and evaluation).", "Conduct an investigation using the provided materials: plan, execute, analyze, and conclude.", "Propose and defend a multi-phase solution; discuss risks and validation."
- **Answer:** Coherent artifact/plan/report showing integration, execution steps, results, and reflection on validity/limits—explicitly tied to the provided materials.

Workflow

1. **Screen the focus chunk for suitability.** If it is not useful for question generation, output:

```
{"content": "not suitable content"}
```

Treat the focus chunk as **not useful** if it consists primarily of any of the following categories:

- Licensing or legal boilerplate.
- Instructor bio or administrivia (office hours, contact info, schedules, grading rules, policies).
- Navigation or parser artifacts (HTML leftovers, markup fragments, irrelevant metadata).
- **Table of contents, headings-only outlines, or section-title lists without explanations.**
- **Learning objectives or intended learning outcomes that state what students *should be able to do* but do not actually explain concepts, definitions, processes, or examples.**

- Course descriptions and logistics rather than subject matter.
 - Empty or near-empty text.
- Proceed **only** if the focus chunk contains at least one of these:
- A definition of a concept or term.
 - An explanation of a mechanism, process, or relationship.
 - A worked example or concrete scenario.
 - A formula, algorithm, or procedure.
 - Explicit factual statements that the learner must know.

If none are present, you must return: {"content": "not suitable content"}

2. **Plan integration.** Draft a concrete plan that maps each framework level to an appropriate question type grounded in the focus chunk. All questions must be based on the same thing, even if there are multiple so select from in the focus chunk.
3. **Validate the plan.** Ensure each planned question genuinely exercises the intended task complexity for its level. If any mismatch, revise the plan before generating.
4. **Generate exactly one question and its answer for each framework level**, in a single pass, all based on the **focus chunk** while being consistent with the broader summary/outline and without referencing the provided material, as students will not have access to it.

Guidelines

- **Language:** Use the same language as the provided chunks.
- **Self-containedness:** Each question must be fully answerable on its own. Assume students do **not** have access to the original course material; include all context or data necessary to understand and answer the question directly.
- **Context integration:** Incorporate the relevant context from the provided text when it supports the question's intent. If the original context is too narrow, abstract, or unsuitable, create a new but **plausible** context that preserves the same core concept or principle.
- **Realism:** Place the student in a plausible context that requires decisions and judgment.
- **Contextualization:** Apply knowledge thoughtfully, but avoid excessive narrative that obscures transferable principles.
- **Problematization:** Give a purpose beyond classroom settings (e.g., client, employer, colleague needs).
- Prefer **concrete** over abstract wording to aid visualization.
- Use **active voice** and directly address the learner with **"you/your."**
- Keep **terminology consistent** across levels.
- **Do not reference any external artefacts** such as lists, tables, figures, diagrams, headings, or sections unless they are fully reproduced inside the question. Avoid phrases like "wie in der Liste angegeben" or "gemäss der Tabelle". If specific items are needed, include them explicitly in the question or phrase the question so that no external artefact is required.
- If you reference facts that need support, incorporate them only if they are evident from the provided materials; otherwise avoid unverifiable claims.
- **Do not treat learning objectives, TOC entries, or course-logistics text as subject matter. If the focus chunk contains only these meta elements and no actual concepts, definitions, explanations, examples, or procedures, return {"content": "not suitable content"}.**
- **Independence:** Each question must stand alone. Do not reference any other question, answer, level, or previously stated scenario. Provide all required context within the question itself.

Output format

- Output a single valid JSON object (double quotes for all keys and strings, replace line breaks inside strings with "\n"). No explanations, no code fences, no extra text.
- **Primary schema (hierarchy: content → level → question/answer):**

```
{
  "content": {
    "DOK 1": { "question": "string", "answer": "string" },
    "DOK 2": { "question": "string", "answer": "string" }
  }
  /* ... continue for all levels, ordered low to high */
}
```

Quality checks before finalizing

- Each question is **answerable from the focus chunk/concept** (use the summary/outline only for alignment and phrasing, not for introducing new facts).
- Each question clearly targets its level's requirement.
- The **provided text is not directly referenced** (no mentions such as "in the text," "according to the passage," or "as described above"), since students will not see the original material.
- The **context is coherent and self-sufficient** — it either draws naturally from the provided text or introduces a new, plausible scenario that preserves the same underlying concept.
- **Confirm that the focus chunk contained substantive subject matter (definitions, explanations, examples, procedures, or factual content). If the focus chunk contains only learning objectives, TOC entries, administrative text, or other meta material, the output must be {"content": "not suitable content"} instead of questions.**
- Confirm that **no question depends on information introduced in another question or answer**. Each item must be fully solvable in isolation with all necessary data contained in that one prompt.

Examples

1. Introductory Statistics — A/B Testing with Difference in Proportions

In online experiments, we often compare conversion in variant B vs control A. Let p_A and p_B be true conversion rates;

estimates are $\hat{p}_A = x_A/n_A$, $\hat{p}_B = x_B/n_B$. The effect size is the **risk difference** $\Delta = \hat{p}_B - \hat{p}_A$. Under large samples,

$$SE(\Delta) = \sqrt{\frac{\hat{p}_A(1-\hat{p}_A)}{n_A} + \frac{\hat{p}_B(1-\hat{p}_B)}{n_B}}$$

A 95% CI is $\Delta \pm 1.96 \cdot SE(\Delta)$. For hypothesis testing $H_0 : p_A = p_B$, use a pooled rate $\hat{p} = (x_A + x_B)/(n_A + n_B)$ and

$$SE_0 = \sqrt{\hat{p}(1-\hat{p}) \left(\frac{1}{n_A} + \frac{1}{n_B} \right)}$$

Then $z = \Delta/SE_0$. Report **effect size**, **uncertainty** (CI), and **practical significance** (e.g., expected revenue lift), not just p -values. Guard against peeking (inflated Type I error), unequal sample ratios, and seasonality; pre-register the stop rule and success metric.

```
{
  "content": {
    "DOK 1": {
      "question": "Define the risk difference Delta between variant B and control A using sample conversion rates x_A/n_A and x_B/n_B.",
      "answer": "Delta = (x_B/n_B) - (x_A/n_A).",
    },
    "DOK 2": {
      "question": "You run an A/B test with A: x_A=500, n_A=10,000 and B: x_B=560, n_B=10,000. Compute the 95% confidence interval for the risk difference. Show your steps.",
      "answer": "p_hat_A=500/10,000=0.050; p_hat_B=560/10,000=0.056; Delta=0.056-0.050=0.006. SE(Delta)=0.003168. 95% CI: 0.006 +/- 1.96*0.003168 => (-0.00021, 0.01221).",
    },
    "DOK 3": {
      "question": "A product manager asks whether B beats A at alpha=0.05. Choose the appropriate significance test for H_0: p_A=p_B and justify your choice. Compute the test statistic and decision.",
      "answer": "Use the pooled two-proportion z-test because H_0 assumes equal rates. p_hat=0.053. SE_0=0.0031683. z=0.006/0.0031683 ~ 1.89 => two-sided p ~ 0.058. Decision at alpha=0.05: fail to reject H_0.",
    },
    "DOK 4": {
      "question": "Design and justify a plan that: (1) defines the success metric, (2) prevents peeking, (3) handles sample ratio and seasonality risks, and (4) specifies how you will report results.",
      "answer": "Plan: (1) Metric: risk difference Delta. (2) Peeking: fixed-horizon stop rule. (3) Target 1:1 allocation; run over full weekly cycles. (4) Estimate Delta and 95% CI; test H_0 using pooled p_hat; report Delta, CI, p-value, and expected lift.",
    }
  }
}
```

2. Cell Biology — Michaelis–Menten Enzyme Kinetics

Many enzymes obey $v = \frac{V_{max}[S]}{K_m + [S]}$. V_{max} is the maximal rate when active sites are saturated; K_m is the substrate concentration at half-maximal rate and reflects **apparent** affinity. At $[S] \ll K_m$, rate is first-order ($v \approx \frac{V_{max}}{K_m}[S]$); at $[S] \gg K_m$, zero-order ($v \approx V_{max}$). Competitive inhibitors raise the **apparent** K_m without changing V_{max} ; noncompetitive lower V_{max} without changing K_m . Turnover number $k_{cat} = V_{max}/[E]_T$; catalytic efficiency k_{cat}/K_m compares enzymes near diffusion limits. Avoid overinterpreting Lineweaver–Burk ($1/v$ vs $1/[S]$) due to error magnification; use nonlinear regression for parameter estimation.

```
{
  "content": {
    "DOK 1": {
      "question": "Define K_m in Michaelis-Menten kinetics.",
      "answer": "K_m is the substrate concentration at which the reaction rate v equals one-half of V_max.",
    },
    "DOK 2": {
      "question": "You study an enzyme with V_max = 120 uM/min and K_m = 30 uM. Compute the approximate rate and identify the kinetic order for (a) [S] = 3 uM and (b) [S] = 300 uM.",
      "answer": "(a) [S] << K_m -> v ~ (V_max/K_m)[S] = (120/30)*3 = 12 uM/min; first-order. (b) [S] >> K_m -> v ~ V_max = 120 uM/min; zero-order.",
    },
    "DOK 3": {
      "question": "An enzyme has V_max=100 and K_m=20. A noncompetitive inhibitor halves V_max to 50 with K_m unchanged. At [S]=200, which choice yields the largest increase in rate: (A) increase [S] fivefold, (B) double [E]_T, or (C) add a competitive inhibitor? Justify.",
      "answer": "Choose (B) double [E]_T. With [S] >> K_m, v ~ V_max. Noncompetitive inhibition lowers V_max, so raising [S] (A) has negligible effect. Increasing [E]_T restores V_max. Adding a competitive inhibitor (C) raises K_m without changing V_max, which does not help at high [S].",
    },
    "DOK 4": {
      "question": "Design and justify a plan to estimate V_max, K_m, and catalytic efficiency (k_cat/K_m) for an enzyme +/- inhibitor.",
      "answer": "Plan: (1) Collect initial-rate data v at >=8 substrate concentrations. (2) Fit v = (V_max[S])/(K_m + [S]) by nonlinear regression. (3) Compute k_cat = V_max/[E]_T. (4) Diagnose inhibitor type. (5) Validate with residual plots.",
    }
  }
}
```

3. Negative Examples (Filtering Unsuitable Content)

Input (Copyright/Legal): All rights reserved. ©2023 Academic Press. This digital version is provided for personal study use only. Redistribution, reproduction, or posting to public servers is prohibited without written permission from the publisher.

{"content": "not suitable content"}

Input (Logistics): Welcome to *Introduction to Organizational Behavior (BUS 201)*. I'm Dr. Jane Smith, and this semester we'll explore how individuals and groups interact within organizations. Please note that attendance is mandatory for all workshops.

{"content": "not suitable content"}

C Concept Extraction + Webb's DOK Example Output

The input document outlined a higher-education course on Supply Chain Management (SCM) focusing on planning horizons, demand forecasting methods, and capacity management. It details technical execution processes including Sales and Operations Planning (S&OP), Bill of Materials (BoM) explosion, and the algorithmic logic of Material Requirements Planning (MRP). Full input examples are left out due to length. The following output is based on one extracted concept from the document:

Example Output: CE + Webb's DOK

DOK 1 Define Sales & Operations Planning (S&OP) in one sentence. Your definition should state who developed it, its primary purpose, the typical planning horizon, how often it is reviewed, and what it aligns operational plans to.

DOK 2 Organize the following activities into the S&OP governance functions. The S&OP governance functions are: Demand (Sales & Marketing), Supply (Operations & SCM), New Activities (Product Management), Finance. For each activity indicate which function owns it:

1. Produce a 12-month rolling demand forecast in units.
2. Set monthly production quantities and plant-level output to meet the plan.
3. Prioritize and schedule resources required for a new product launch.
4. Translate the agreed volume plan into monthly revenue and budgeted costs in currency.

DOK 3 You are the S&OP lead at a mid-sized manufacturer. Sales proposes an optimistic 18-month ramp; Finance insists on a conservative volume for the budget; Operations warns that any capacity expansion requires a 12-month lead time. Using S&OP principles — it balances demand and supply, is an executive decision-making process, covers a mid-term horizon up to 24 months and is reviewed monthly, and aligns operational plans to financial plans (volumes in units and currency) — choose between these options and justify your recommendation:

1. Adopt Finance's conservative budget now and set supply to that lower volume.
2. Adopt Sales' optimistic forecast and initiate capacity expansion immediately.
3. Run a cross-functional reconciliation to produce a consensus S&OP plan, surface remaining trade-offs and risks to the executive S&OP meeting for a decision, and update monthly as data arrives.

Which option do you choose? Provide (1) your chosen option and rationale, (2) three concrete S&OP steps you will execute in the next month to reconcile the views, and (3) the key data you will bring to the executive meeting to enable a clear decision.

DOK 4 Design and justify a 12-month implementation plan to establish a formal S&OP process at a mid-size manufacturer that currently plans in functional silos. The S&OP must balance demand and supply, function as executive decision-making, and align operational plans (volumes in units) with financial plans (currency) over a mid-term horizon (up to 24 months). Your plan must specify:

1. Governance and roles (who owns S&OP, who attends the executive S&OP meeting).
2. A month-by-month rollout timeline with the main milestones for months 1–12.
3. The monthly meeting cadence (types of meetings, their purpose, typical inputs and outputs).
4. Minimum data and deliverables required each month to align volumes to financials.
5. Five KPIs to evaluate S&OP performance and how often you will review them.
6. Three likely implementation risks and your mitigation measures.

Justify how your design applies the core S&OP principles (executive decision-making, balance demand/supply, mid-term horizon, monthly review, alignment to financials).

Optimizing Large Language Models for Robust Domain-Specific Text-to-SQL: From Prompting to Preference Alignment

Noah Hampp*

ETH Zürich

Switzerland

noah.hampp@gmail.com

Katya Mirylenka*

TUWien

Austria

katsiaryna.mirylenka@tuwien.ac.at

Michael Glass

IBM Research

USA

mrglass@us.ibm.com

Abstract

This work explores the optimization of Large Language Models (LLMs) for the task of generating SQL queries from natural language (NL2SQL), a critical capability for democratizing access to domain-specific data. While recent benchmarks show promising results for LLMs, deployment in real-world analytical processing requires strict adherence to SQL grammar, deep domain understanding, and robustness against out-of-scope queries. We present a comprehensive study evaluating three stages of optimization: (1) advanced prompting strategies including Chain-of-Thought and multi-turn conversational handling; (2) constrained decoding to enforce syntactic validity; and (3) Reinforcement Learning with AI Feedback (RLAIF). We specifically compare Proximal Policy Optimization (PPO), Direct Preference Optimization (DPO), and Odds Ratio Preference Optimization (ORPO) using a novel reward modeling approach based on execution and semantic principles. Our results reveal that while standard PPO suffers from reward sparsity and catastrophic collapse on 7B models, monolithic alignment via ORPO scales efficiently to 20B parameter models. This provides a stable alternative to expensive inference-time scaling, offering a highly reproducible, single-pass pipeline for adapting open-weights models to complex data environments, serving as a low-latency alternative to agentic systems.

1 Introduction

The transformation of Natural Language Queries (NLQs) into Structured Query Language (SQL) is a pivotal challenge in data management. It promises to simplify database interactions for non-technical users, democratizing access to data-driven insights. As conversational agents become central to business intelligence, the ability to reliably converse with data warehouses without knowledge of SQL

syntax is increasingly critical (Li et al., 2023a). Achieving this vision of robust Conversational Data Analytics (CDA) requires a paradigm shift towards systems that can consistently produce verifiable analytical insights through dynamic natural language interfaces (Amer-Yahia et al., 2025). Such robust natural language interfaces are particularly crucial when non-technical end-users need to explore and run similarity matches against complex, inherently noisy real-world records, such as uncertain temporal and sensor data (Dallachiesa et al., 2011).

However, translating natural language to SQL presents unique challenges compared to general code generation. It requires strict adherence to database schemas, disambiguation of user intent based on domain knowledge, and the ability to prioritize among multiple high-confidence candidate queries. While LLMs have emerged as the state-of-the-art solution, standard models frequently suffer from generating plausible but incorrect queries and struggle with the strict syntactic constraints of SQL dialects (Li et al., 2024). To mitigate this and ensure high user trust, recent advances emphasize the importance of black-box uncertainty quantification (UQ) to accurately estimate confidence in generated SQL and general question answer (QA) outputs (Bhattacharjya et al., 2024; Xiao et al., 2025; Bhattacharjya et al., 2025).

For instance, a user might ask "Who was the best-selling artist when I was born?" without providing their age. The model must infer the missing context and map "best-selling" to specific schema columns, which often leads to errors in unoptimized models.

The landscape of Text-to-SQL has shifted from a translation task to a complex reasoning challenge. Current methodologies generally fall into two camps: inference-time scaling (agentic workflows) and training-time alignment. Agentic systems like Agentar-Scale-SQL (Wang et al., 2025b) and MAC-SQL (Wang et al., 2025a) leverage massive test-time computation, employing multiple

*Work done while at IBM Research Switzerland.

agents to decompose questions and iteratively repair errors. While highly accurate, these systems suffer from high latency and cost, making them difficult to deploy in real-time analytical settings.

In this work, we present an empirical study evaluating the spectrum of optimization techniques, moving from lightweight inference-time controls to comprehensive training-time alignment. Our goal is to identify how to internalize reasoning capabilities directly into the model weights to produce correct SQL efficiently in a single pass. We structure our investigation into three stages: First, we investigate *prompt engineering* strategies, analyzing the trade-off between complex reasoning chains and instruction-following capabilities in mid-sized models. Second, we explore *constrained decoding* to enforce strict syntactic correctness during generation. Finally, we apply *Reinforcement Learning with AI Feedback* (RLAIF) to align models with execution and semantic preferences. In alignment with the goal of reproducible NLP, we offer a critical analysis of why standard PPO collapses in this domain and how ORPO provides a more stable, resource-efficient, and easily replicable alternative.

2 Related Work

Natural Language to SQL (NL2SQL) translation is an interdisciplinary field bridging linguistics and database management. Early systems relied on rule-based approaches with handcrafted grammatical rules (Warren and Pereira, 1982). Later systems utilized database schemas and indices to map NL queries to graph representations of relations (Hristidis et al., 2003). With improved parsers, the standard shifted to mapping grammatical structures of the input question to SQL queries. However, these systems struggled with the ambiguity and variability of real-world queries.

Deep Learning and LLMs. The advent of the Transformer architecture (Vaswani et al., 2017) and large datasets like Spider (Yu et al., 2018) marked the beginning of deep-learning NL2SQL systems. Sequence-to-Sequence (Seq2Seq) models enabled breakthroughs, but modern decoder-only LLMs such as *CodeLlama* (Roziere et al., 2023) and *Granite* (Mishra et al., 2024) have since become dominant. These models undergo extensive pre-training on code and text but require fine-tuning for domain specificity. To reduce computational costs, Parameter-Efficient Fine-Tuning (PEFT) methods like LoRA (Hu et al., 2022) and

quantization (Hubara et al., 2018) are employed.

Prompting and Agents. A crucial factor in harnessing LLMs lies in prompting. Pourreza and Rafiei (Pourreza and Rafiei, 2023) introduced DIN-SQL, demonstrating that decomposing the generation task into schema linking, query classification, and generation significantly improves performance. Recent in-depth analyses further confirm that robust LLM-based schema linking is a critical driver of overall NL2SQL accuracy, often requiring careful prompting or decomposition to match or exceed oracle representations (Katsogiannis-meimarakis et al., 2026). However, recent work suggests that complex prompting strategies can induce "hallucination propagation" in CoT (Li et al., 2024), where errors in early reasoning steps cascade into the final query. This led to the rise of agentic workflows (CHASE-SQL (Pourreza et al., 2025a) and Mac-SQL (Wang et al., 2025a)), which use iterative refinement and multi-agent debate to correct errors.

Alignment and Reinforcement Learning. Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Ziegler et al., 2019) became the standard for aligning LLMs. However, collecting human preference data for SQL is expensive and requires domain expertise. While prior work investigated active learning strategies to mitigate these heavy annotation costs in specialized domains (Wertz et al., 2022), eventually optimizing them with reinforcement learning (Wertz et al., 2023), RLAIF (Bai et al., 2022) offers a highly scalable alternative by substituting human annotators entirely with AI feedback.

PPO has been the engine of RLHF, however, recent findings indicate it is notoriously unstable for code generation due to reward sparsity (Pourreza et al., 2025b). Direct Preference Optimization (DPO) (Rafailov et al., 2023) and Odds Ratio Preference Optimization (ORPO) (Hong et al., 2024) offer more stable alternatives. ORPO, in particular, excels in scenarios with class imbalance (valid vs. invalid queries), making it a promising candidate for monolithic alignment in NL2SQL.

3 Methodology: Inference Optimization

We first explore inference-time optimizations to guide LLMs toward correct SQL generation without updating model weights.

3.1 Prompt Tuning Strategies

Prompt tuning is essential for focusing the model on the specific task of SQL translation. As baselines, we utilize *Simple Generation*, which provides a zero-shot instruction alongside the database schema, and *Schema-linking* (SLink), which explicitly prompts the model to identify relevant tables and columns before writing the query. We evaluate several strategies:

3.1.1 Self-Correction with Error Feedback

Building on simple generation, we implement a feedback loop where the LLM is prompted to fix its own errors. If an initial generated query returns a database error (e.g., "column does not exist"), the error message and the schema are fed back into the model with a correction instruction. We utilize a specific mapping of error messages to instructions, as shown in Table 1.

Table 1: Mapping from database error messages to special correction instructions used in our pipeline.

Error Message	Special Instruction
column does not exist	Replace the column that does not exist with an existing one from the Database_schema.
ambiguous column name	Add the table specifier to the column reference that is ambiguous.
operator does not exist: [...] = [...]	The query tries to join two tables on columns with different datatypes. Restructure the JOIN using provided Foreign_keys.
missing FROM-clause entry	The SQL uses tables without declaring them in the FROM clause. Add them.

3.1.2 Chain-of-Thought (CoT)

We apply the approach described by Pourreza and Rafiei (Pourreza and Rafiei, 2023) (DIN-SQL), decomposing the task into: (1) Schema Linking, (2) Query Classification (Easy, Nested, Non-Nested), and (3) Generation based on the class. While effective for large models like GPT-4, we investigate its efficacy on smaller open-source models (7B-34B parameters).

3.2 Conversational Strategies

Real-world analytics involves multi-turn conversations where subsequent questions specify or contradict previous ones. We evaluate three strategies:

- *Full Context*: Concatenating all previous questions and generated SQL queries.

- *Questions Only*: Concatenating only the history of natural language questions to avoid biasing the model with potentially incorrect previous SQL.
- *Merge Questions*: A two-step process where the LLM first merges the conversation history into a single standalone question (resolving coreferences) and then generates SQL from that merged question.

3.3 Constrained Decoding

To eliminate syntax errors, we apply constrained decoding using a context-free grammar (CFG) derived from the database schema. We adapt the T5QL approach (Arcadinho et al., 2022). Similar constrained generation and fine-tuning frameworks have recently proven highly effective in related structured generation tasks, such as strictly enforcing syntax and schema adherence when integrating complex enterprise API workflows from natural language (Chan et al., 2024). During beam search, for a current generation P , we calculate the set of valid next tokens N^* that satisfy SQL syntax and schema constraints (e.g., only valid table names after FROM).

The procedure is detailed in Algorithm 1. For each decoding step, given the current generation P , we find the maximum parsable prefix P^* and filter the vocabulary to only allow tokens that form a valid suffix in the Trie T of allowable SQL constructs. Crucially, while the base grammar is context-free, our adaptation for auto-regressive LLMs incorporates context-aware decision making. Specifically, the *FilterWrongTokens* function constrains the generation by strictly allowing only the generation of columns that have been explicitly defined in the prior FROM statement, and by mapping table aliases back to their original tables to prevent hallucinated references.

4 Methodology: Alignment via RLAIIF

To align models with SQL execution accuracy, we employ Reinforcement Learning with AI Feedback (RLAIIF). Our pipeline, visualized in Figure 1, replaces human annotators with AI judges to scale the alignment process.

4.1 Reward Modeling and Principles

We construct a preference dataset by prompting multiple LLMs (CodeLlama-34B, Granite-20B, Starcoder) to generate candidate SQL queries.

Algorithm 1 Constrained Decoding Procedure

```

1: procedure NEXTTOKEN( $P, T$ )
2:    $P^* \leftarrow \text{FindParsablePrefix}(P)$ 
3:    $S \leftarrow \text{GetParserState}(P^*)$ 
4:    $N \leftarrow \text{ParserNextTokens}(S)$ 
5:    $N^* \leftarrow \text{FilterWrongTokens}(S, N)$ 
6:   for  $n$  in  $N^*$  do
7:      $C \leftarrow P^* + n$ 
8:      $CT \leftarrow \text{SentenceTokenizer}(C)$ 
9:      $T \leftarrow \text{AddToTrie}(T, CT)$ 
10:  end for
11:   $PT \leftarrow \text{SentenceTokenizer}(P)$ 
12:  return  $\text{GetChildren}(T, PT)$ 
13: end procedure

```

These candidates are scored based on two categories of principles:

Execution Principles Validated by executing the query against the database engine. These are binary signals that ensure functional correctness:

- *Correctness*: Does the query return the same result set as the gold standard?
- *Schema Validity*: Are all columns and tables present in the schema?
- *No Ambiguity*: Are column references unambiguous?

Semantic Principles Validated by a "Teacher LLM" (CodeLlama-70B) prompted to act as a SQL expert. These provide finer-grained feedback on query quality, using specific prompts tailored to common SQL errors. For example:

- *Correct GROUP BY*: "Read the question and the corresponding SQL, determine if the question calls for a GROUP BY or not and if the SQL is accurate for that."
- *Only Necessary Joins*: "Check that the query only accesses tables necessary to answer the question."
- *Answer Relevance*: "Check if the SQL really answers the intent of the question."
- *Correct DISTINCT*: "Check that DISTINCT is applied if necessary to avoid duplicates."

We generate triplets of $(\text{Question}, \text{Winner}, \text{Loser})$ based on these scores. A query is considered a "Winner" if it satisfies more principles, particularly the Execution Correctness principle.

We create two preference datasets: the "Multi-LLM" dataset utilizes generated SQL from eight distinct models, while the "Single-LLM" dataset samples multiple outputs exclusively from CodeLlama-70B. For the Single-LLM dataset, the temperature was tuned so that roughly one in eight generated queries was correct, controlling the balance of variation and correctness.

4.2 Reward Model Loss Function

To enable PPO, we first train a Reward Model (RM) to predict a scalar score for a query. We fine-tune a CodeLlama-34B model with a scoring head. The loss function maximizes the margin between the score for the good query (y_{good}) and the bad query (y_{bad}):

$$\mathcal{L}(y_{good}, y_{bad}) = \max(0, m - (y_{good} - y_{bad})) \quad (1)$$

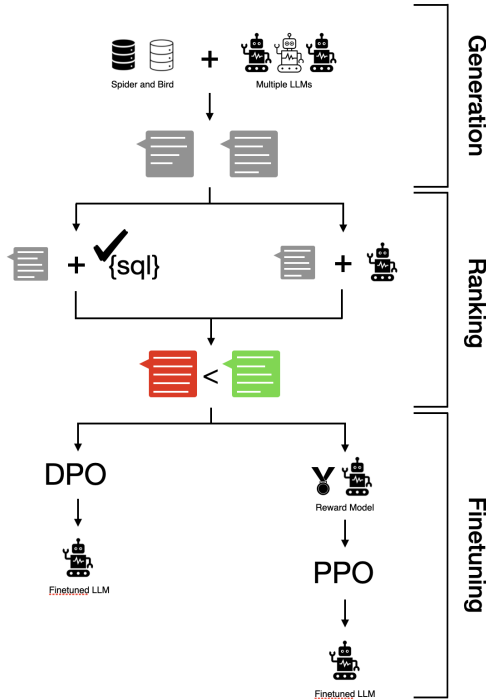


Figure 1: Our RLAIF Training Pipeline. We generate candidate queries using multiple LLMs, rank them using execution and semantic principles, and use the resulting triplets for DPO, PPO, and ORPO training.

where m is the margin. We experimented with both a single linear layer and an MLP as the scoring head. The linear layer proved superior, achieving 94% accuracy on the test split, whereas the MLP suffered from overfitting on the sparse SQL preference data.

4.3 Distribution of Rewards

To ensure the reward model discriminates effectively, we analyzed the distribution of scores for generated queries. Figure 2 shows the score distribution for correct (green) vs. incorrect (red) queries on the Single-LLM dataset. The clear separation indicates that the reward model successfully learns to distinguish valid SQL logic from incorrect attempts. While the reward model sometimes assigns positive absolute scores to incorrect queries, our optimization objectives rely primarily on the relative margin between the winning and losing generations, making the absolute scale less critical to the policy update.

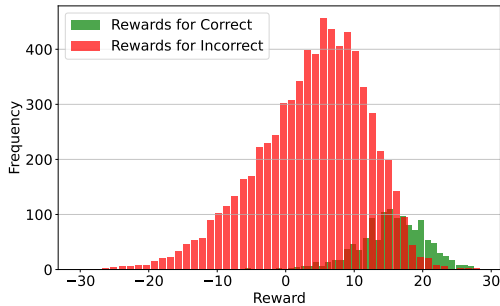


Figure 2: Reward Distributions on the test set. Green bars represent correct SQL queries, while red bars represent incorrect ones. The clear separation demonstrates the effectiveness of the reward model.

4.4 Alignment Algorithms

We evaluate three optimization algorithms using this dataset.

(1) PPO (Proximal Policy Optimization). We fine-tune the policy using PPO to maximize the reward predicted by the RM. The Reward Model training procedure (Algorithm 2) updates the scoring head using a margin loss. Subsequently, we use this frozen Reward Model to fine-tune the policy using PPO.

$$\mathcal{L}^{CLIP}(\theta) = \mathbb{E}_t \left[\min \left(r_t(\theta) \hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right] \quad (2)$$

Algorithm 2 Reward Model Training Procedure

Require: Model to fine-tune M , Reward Model R , margin m

- 1: **while** training **do**
 - 2: Load question Q , schema S , correct query q_{correct} , and incorrect query $q_{\text{incorrect}}$
 - 3: Generate prompt $p_{\text{correct}} = \text{CreatePrompt}(Q, S, q_{\text{correct}})$.
 - 4: Generate prompt $p_{\text{incorrect}} = \text{CreatePrompt}(Q, S, q_{\text{incorrect}})$.
 - 5: $s_{\text{good}} = R(p_{\text{correct}})$.
 - 6: $s_{\text{bad}} = R(p_{\text{incorrect}})$.
 - 7: $\mathcal{L}(s_{\text{good}}, s_{\text{bad}}) = \max(0, m - (s_{\text{good}} - s_{\text{bad}}))$
 - 8: Update M using the gradient of the loss \mathcal{L} .
 - 9: **end while**
-

(2) DPO (Direct Preference Optimization). DPO optimizes the policy directly from preference pairs, eliminating the explicit reward model.

$$\mathcal{L}_{DPO} = -\mathbb{E}_{(x, y_w, y_l)} \left[\log \sigma \left(\beta \log \frac{\pi_{\theta}(y_w|x)}{\pi_{ref}(y_w|x)} - \beta \log \frac{\pi_{\theta}(y_l|x)}{\pi_{ref}(y_l|x)} \right) \right], \quad (3)$$

where π_{θ} is the policy model, π_{ref} is the frozen reference model, and y_w, y_l are the winning and losing generations, respectively.

(3) ORPO (Odds Ratio Preference Optimization). Odds Ratio Preference Optimization (ORPO) integrates preference alignment into the Supervised Fine-Tuning (SFT) stage. It adds an odds-ratio penalty to the negative log-likelihood loss:

$$\mathcal{L}_{ORPO} = \mathbb{E}_{(x, y_w, y_l)} [\mathcal{L}_{SFT} + \lambda \cdot \mathcal{L}_{OR}] \quad (4)$$

This method is particularly resource-efficient as it requires no separate warm-up phase or reference model, unlike inefficient agentic workflows.

5 Experimental Setup

5.1 Datasets

We evaluate our methods on the Spider (Yu et al., 2018) and BIRD (Li et al., 2023a) benchmarks, as well as a proprietary IBM Business Intelligence (IBM BI) dataset. *BIRD* represents a large-scale challenge with databases containing noisy data and

massive schemas. To illustrate the domain complexity, Figure 3 shows the schema sizes in these benchmarks. BIRD databases contain significantly more tables than Spider.

Accurately matching natural language mentions to complex, interconnected database schemas is fundamentally an entity matching and record linkage problem—challenges that have historically required dedicated graph-based modeling (Krivosheev et al., 2021, 2023) or specialized token-based similarity matching architectures (Mirylenska et al., 2021) to overcome noisy string variations and capture relational context. Across all benchmarks, we use *Execution Accuracy* as our primary metric, which considers a generated query correct if its execution returns the exact same rows and columns as the gold query. The IBM BI dataset consists of 101 multi-turn conversations where subsequent user questions specify or contradict previous ones.

5.2 Baselines and Models

We benchmark against several competitive 7B–34B parameter models to evaluate the impact of our optimization strategies:

- *CodeLlama-34B-Instruct* (Roziere et al., 2023): A powerful general-purpose code model used as our primary baseline for prompting experiments.
- *Granite-20B-Code* (Mishra et al., 2024): An enterprise-focused code model from IBM, selected for its strong performance on SQL.
- *StarCoder* (Li et al., 2023b): A 15B parameter model trained on diverse code data, serving as a baseline for the multi-model preference dataset generation.
- *SQLCoder-34B* (defog ai, 2024): A specialized fine-tuned model for SQL, used to benchmark our constrained decoding approach.

5.3 Training Configuration

For fine-tuning, we utilized 8x NVIDIA A100 80GB GPUs. We employed the SFTTrainer from TRL and leveraged DeepSpeed ZeRO-3 for distributed training. Specific hyperparameters (e.g., LoRA configuration, learning rates, and batch sizes) are detailed in Appendix B.

6 Experiments and Results

6.1 Prompting and Error Analysis

Table 2 presents the accuracy of different prompting strategies. Notably, on the IBM BI dataset,

the simpler "Self-Correction" (SCorr) strategy outperformed the complex "DinSQL" strategy on CodeLlama-34B (0.327 vs 0.168).

To understand why complex prompting failed for smaller models, we analyzed the error types. Figure 4 provides a side-by-side comparison. In the Simple Generation (Left Subfigure): The errors are distributed among logical issues like "more rows than necessary" and "not all rows". For the DinSQL (right subfigure): There is a massive spike in "No Such Column" errors (hallucinations). This indicates that the complex reasoning chain of DinSQL distracted the smaller model from the schema grounding, causing it to invent columns rather than focusing on the provided schema.

Table 2: Accuracy of prompting on IBM BI, Spider, and BIRD dev sets. Acronyms: Gen (Simple Generation), SCorr (Self-Correction), SLink (Schema-linking), and DinSQL (Decomposed In-Context Learning).

Model	Dataset	Gen	SCorr	SLink	DinSQL
CodeLlama-34B	IBM BI	0.317	0.327	0.297	0.168
CodeLlama-34B	Spider	0.707	0.714	0.700	0.668
Granite-20B	Spider	0.615	0.671	0.534	0.540

6.2 Conversational Analysis

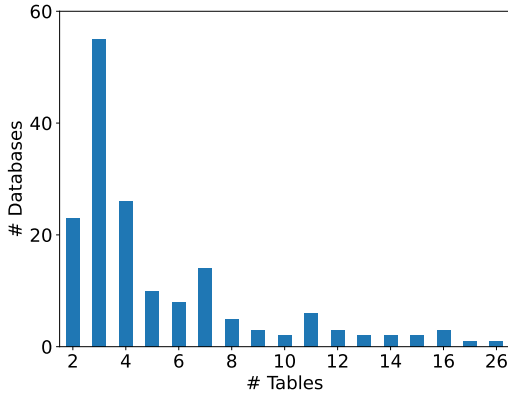
In our multi-turn experiments (Table 3), providing the *ground truth* SQL from previous turns significantly aids the model. Here, the "Merge Questions" strategy proved most robust (Accuracy 0.473), as it reduces noise by synthesizing a clean, standalone query. However, when the model relies on its own *predicted SQL*, error propagation causes performance to degrade rapidly. In this realistic deployment setting, "Full Context" slightly outperforms "Merge Questions" (0.139 vs 0.122), suggesting that forcing the model to merge its own error-prone historical queries degrades the context more than simply concatenating the raw history.

Table 3: Average accuracy on IBM BI multi-turn conversations.

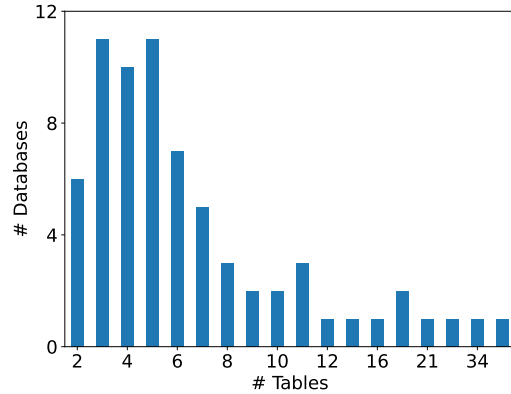
Context Source	Method	Avg Accuracy
Ground Truth SQL	Full Context	0.353
Ground Truth SQL	Merge Questions	0.473
Predicted SQL	Full Context	0.139
Predicted SQL	Merge Questions	0.122

6.3 Constrained Decoding Performance

To evaluate the impact of restricting the output space, we tested our constrained decoding algo-



(a) Spider Dataset



(b) BIRD Dataset

Figure 3: Distribution of the number of tables per database in Spider and BIRD. BIRD contains significantly larger schemas, increasing the difficulty of the generation task.

rithm using defog/sqlcoder (defog ai, 2024) on the IBM BI dataset. We created a context-free grammar (CFG) from the schema to restrict the output space, ensuring valid SQL syntax and correct usage of table and column names. Due to the high computational overhead of context-aware filtering during beam search, we limited decoding to two beams. Nevertheless, the decoding process took approximately 10x longer for the same amount of tokens compared to unconstrained generation.

Table 4: Primary Impact of Constrained Decoding on SQL generation (IBM BI Dataset). "Row Correct" indicates the query returned the correct rows but a super/subset of the required columns.

Result Category	Unconstrained	Constrained
Correct	7	12
Row Correct	33	37
Incorrect	46	17
Error	15	35
Accuracy	0.069	0.119

As shown in Table 4, applying the CFG improved exact execution accuracy from 6.9% to 11.9%, and subset accuracy (Row Correct) from 39.6% to 48.5%. However, counter-intuitively, the constrained model produced more than twice the amount of SQL errors (35 vs. 15), primarily driven by Syntax Errors and Ambiguous Columns. A detailed quantitative breakdown of these error types, along with a qualitative analysis of the failure modes, is provided in Appendix A. Briefly, while the CFG successfully restricts syntax, it struggles to force the model to output an End-Of-Sequence

(EOS) token when logical reasoning fails.

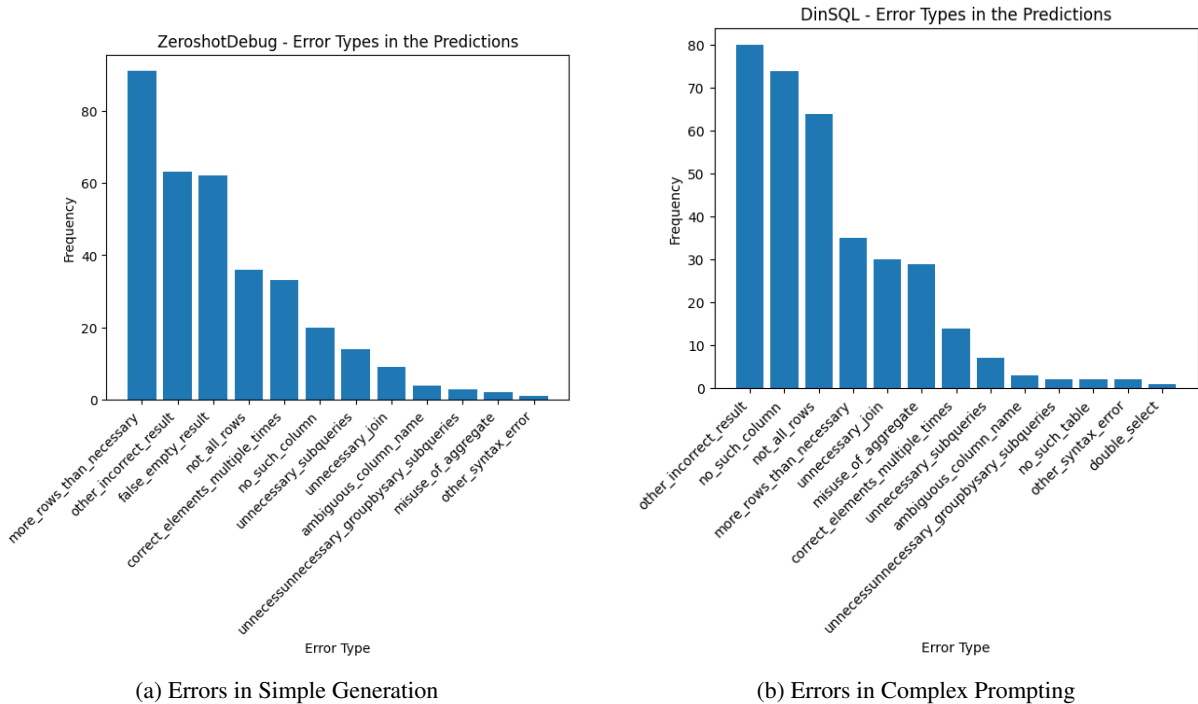
6.4 RLAIF Training Stability and PPO Failure

Note on Evaluation Setup: Due to the immense computational requirements of reinforcement learning, our PPO and DPO experiments were conducted on the 7B parameter CodeLlama model using the Spider dataset. Conversely, our monolithic ORPO experiments were scaled up to the Granite-20B model on the more complex BIRD benchmark. While this prevents a direct 1:1 numerical comparison across all methods, the distinct failure modes of PPO and DPO at the 7B scale provide critical context for the necessity of the ORPO approach.

Table 5: Consolidated alignment results highlighting the performance of PPO, DPO, and ORPO compared to their respective base models. Note that PPO and DPO were evaluated on CodeLlama-7B (Spider) while ORPO was scaled to Granite-20B (BIRD).

Algorithm	Model	Dataset	Base Acc.	Aligned Acc.
PPO (Multi LLM)	CodeLlama-7B	Spider	0.460	0.000
PPO (Single LLM)	CodeLlama-7B	Spider	0.460	0.000
DPO (Multi LLM)	CodeLlama-7B	Spider	0.460	0.030
DPO (Single LLM)	CodeLlama-7B	Spider	0.460	0.360
ORPO	Granite-20B	BIRD (Avg)	0.651	0.646

Our experiments revealed significant challenges with PPO in the SQL domain. When fine-tuning 7B models with PPO, we frequently observed model collapse, characterized by the generation of repetitive gibberish or empty strings. This instability arises from the sparsity of the reward signal: in NL2SQL, the execution reward is binary (correct/incorrect). Unlike prose generation where "style"



(a) Errors in Simple Generation

(b) Errors in Complex Prompting

Figure 4: Comparison of error types on Spider Dev. The complex DinSQL strategy (Right) introduces significantly more "no such column" hallucinations compared to simple generation (Left), indicating that smaller models struggle with long context chains.

rewards are dense, a SQL query that is 99% correct but misses a single comma receives a reward of 0.

We conducted a sensitivity analysis on PPO batch sizes to mitigate this (Table 6). We found that PPO is extremely sensitive to batch size; with a batch size of 1, the model learned nothing (Accuracy 0.0), whereas increasing the batch size to 32 stabilized the training slightly (Accuracy 0.42), though still below supervised baselines.

Table 6: Sensitivity of PPO training to batch size. Small batch sizes lead to total collapse due to sparse rewards.

Batch Size	Final Accuracy
1	0.00
8	0.01
32	0.42
64	0.22

In contrast, DPO and ORPO showed much greater stability without extensive hyperparameter tuning. Figure 5 visualizes the training loss and accuracy for the DPO approach on the Single-LLM dataset. The smooth convergence of accuracy (blue line) demonstrates that the model effectively learns from the AI-generated preference pairs.

Additionally, we monitored the reward margin during training. Figure 6 shows the margin between

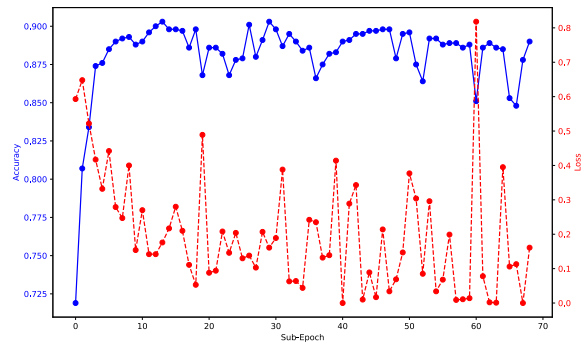


Figure 5: Training dynamics. The red dashed line represents loss (right axis) and the blue solid line represents accuracy (left axis) during fine-tuning. The model converges stably.

chosen and rejected responses. The positive trend confirms that the model is successfully optimizing for the principles defined in our RLAIIF pipeline.

6.5 ORPO Results

During evaluation, both the Base SFT and ORPO models were tested using the Simple Generation zero-shot prompt. On the BIRD benchmark (Table 7), we observed a nuanced trade-off. ORPO led to a slight decrease in accuracy on "Simple" and "Moderate" queries, resulting in a minor drop in overall average accuracy (from 0.651 to 0.646). How-

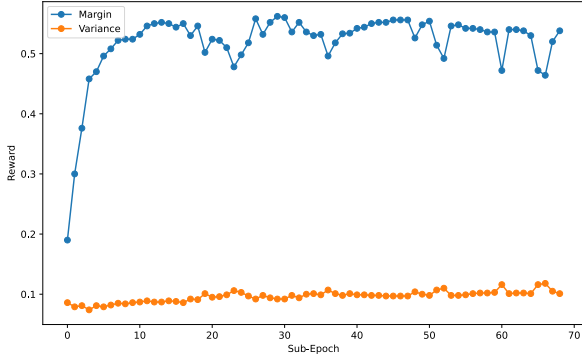


Figure 6: Evolution of the Reward Margin during training. The increasing margin indicates the model is learning to distinguish between high-quality and low-quality SQL generations.

ever, ORPO significantly improved performance on "Challenging" queries (from 44.4% to 47.2%). Thus, while monolithic preference optimization may introduce slight regression on basic syntax following, it effectively aligns the model with complex logical reasoning without suffering from the catastrophic instability observed in PPO.

Due to computational constraints, these metrics represent single-run results. Future work will evaluate variance across multiple training seeds to establish the statistical significance of the gains.

Table 7: Accuracy results on BIRD Development Set categories using ORPO.

Method	Simple	Moderate	Challenging	Average
Base SFT	0.715	0.589	0.444	0.651
ORPO	0.704	0.585	0.472	0.646

7 Discussion

Our findings challenge the assumption that standard RLHF methods transfer seamlessly to code generation. The sparsity of the reward signal in SQL makes PPO training unstable. ORPO offers a compelling alternative by integrating alignment into the supervised training process.

We observed that Reward Models trained on data generated by multiple LLMs (Multi-LLM) failed to generalize when scoring the Single-LLM dataset during training. This suggests that for RLAIIF to function effectively, the Reward Model should be trained on the distribution of the specific model being fine-tuned, or a highly diverse set of models.

In RLHF for text generation, rewards are often dense (e.g., style, tone). In SQL, the reward is

sparse and binary (executable vs. not). DPO struggles with "credit assignment"—if a model generates a wrong SQL query, DPO penalizes the entire sequence. Without Chain-of-Thought traces to indicate where the reasoning failed, the model cannot learn effectively. ORPO mitigates this by balancing the odds ratio, effectively focusing on the differentiation between valid and invalid patterns without requiring an explicit reward model.

The trade-off between complex prompting (Din-SQL) and simpler self-correction suggests that for deployment on mid-sized models (20B-34B parameters), simpler, iterative strategies are more robust. Complex chains of thought can overwhelm the attention mechanism of smaller models, leading to hallucinations of schema elements.

8 Conclusion and Future Work

This work presents a holistic view of optimizing LLMs for NL2SQL. We identified that for domain-specific applications, lightweight alignment via ORPO combined with robust self-correction prompting yields an effective balance of performance and reliability. While our computational limits restricted PPO evaluations to 7B models—where we observed severe instability due to sparse binary rewards—ORPO demonstrated stable scaling to 20B parameters, effectively improving complex logical reasoning without catastrophic collapse. Our error analysis revealed that complex Chain-of-Thought prompting can be detrimental for smaller models, inducing hallucinations. Finally, we demonstrated that RLAIIF using execution and semantic principles is a promising path to alignment without expensive human annotation. While this approach champions reproducible NLP by bypassing closed-source LLM judges or costly multi-agent deployments, future work must establish the correlation between our AI-generated semantic feedback and expert human judgments to fully validate the teacher model's efficacy.

Future work will focus on scaling ORPO to larger distributed models and further refining automated reward signals for RLAIIF. Specifically, we aim to explore hybrid approaches that combine the stability of ORPO with the reasoning capabilities of agentic workflows, potentially using "partial rewards" for intermediate reasoning steps to address the sparsity issue inherent in SQL generation.

References

- Sihem Amer-Yahia, Jasmina Bogojeska, Roberta Facchinetti, Valeria Franceschi, Aristides Gionis, Katja Hose, Georgia Koutrika, Roger Kouyos, Matteo Lissandrini, Silviu Maniu, and 1 others. 2025. Towards reliable conversational data analytics. In *2025 EDBT/ICDT 2025 Joint Conference*.
- Samuel David Arcadinho, David Aparício, Hugo Veiga, and António Alegria. 2022. T5ql: Taming language models for sql generation. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 276–286.
- Yuntao Bai and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.
- Debarun Bhattacharjya, Balaji Ganesan, Michael Glass, Junkyu Lee, Radu Marinescu, Katsiaryna Mirylenka, and Xiao Shou. 2024. [Consistency-based black-box uncertainty quantification for text-to-SQL](#). In *Statistical Foundations of LLMs and Foundation Models (NeurIPS 2024 Workshop)*.
- Debarun Bhattacharjya, Balaji Ganesan, Junkyu Lee, Radu Marinescu, Katya Mirylenka, Michael Glass, and Xiao Shou. 2025. Simba uq: Similarity-based aggregation for uncertainty quantification in large language models. In *Conference on Empirical Methods in Natural Language Processing*.
- Robin Chan, Katsiaryna Mirylenka, Thomas Gschwind, Christoph Miksovich, Paolo Scotton, Enrico Toniato, and Abdel Labbi. 2024. Adapting llms for structured natural language api integration. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 991–1000.
- Paul F Christiano and 1 others. 2017. Deep reinforcement learning from human preferences. *NeurIPS*.
- Michele Dallachiesa, Besmira Nushi, Katsiaryna Mirylenka, and Themis Palpanas. 2011. Similarity matching for uncertain time series: analytical and experimental comparison. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Querying and Mining Uncertain Spatio-Temporal Data*, pages 8–15.
- defog ai. 2024. Sqlcoder. <https://github.com/defog-ai/sqlcoder>.
- Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. In *EMNLP*.
- Vagelis Hristidis and 1 others. 2003. Efficient ir-style keyword search over relational databases. In *VLDB*.
- Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.
- Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. 2018. Quantized neural networks: Training neural networks with low precision weights and activations. *Journal of Machine Learning Research*.
- George Katsogiannis-meimarakis, Katya Mirylenka, Paolo Scotton, Francesco Fusco, and Abdel Labbi. 2026. In-depth analysis of llm-based schema linking. In *International Conference on Extending Database Technology*.
- Evgeny Krivosheev, Mattia Atzeni, Katsiaryna Mirylenka, Paolo Scotton, Christoph Miksovich, and Anton Zorin. 2021. Business entity matching with siamese graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 16054–16056.
- Evgeny Krivosheev, Katsiaryna Mirylenka, Mattia Atzeni, and Paolo Scotton. 2023. Graph neural networks for entity matching. In *2023 IEEE International Conference on Big Data (BigData)*, pages 6212–6214. IEEE.
- Jinyang Li, Binyuan Hui, Ge Qu, Jiayi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, and 1 others. 2023a. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *NeurIPS*.
- Junyi Li, Jie Chen, Ruiyang Ren, Xiaoxue Cheng, Wayne Xin Zhao, Jian yun Nie, and Ji-Rong Wen. 2024. The dawn after the dark: An empirical study on factuality hallucination in large language models. In *Annual Meeting of the Association for Computational Linguistics*.
- Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, and 1 others. 2023b. Starcoder: may the source be with you! *Trans. Mach. Learn. Res.*
- Katsiaryna Mirylenka, Paolo Scotton, Christoph Adrian Miksovich Czasch, and Andreas Schade. 2021. Similarity matching systems and methods for record linkage. US Patent 11,182,395.
- Mayank Mishra and 1 others. 2024. [Granite code models: A family of open foundation models for code intelligence](#). *Preprint*, arXiv:2405.04324.
- Mohammadreza Pourreza, Hailong Li, Ruoxi Sun, Yeounoh Chung, Shayan Talaie, Gaurav Tarlok Kakkar, Yu Gan, Amin Saberi, Fatma Ozcan, and Sercan O Arik. 2025a. Chase-sql: Multi-path reasoning and preference optimized candidate selection in text-to-sql. In *ICLR*.
- Mohammadreza Pourreza and Davood Rafiei. 2023. Din-sql: Decomposed in-context learning of text-to-sql with self-correction. In *NeurIPS*.

Mohammadreza Pourreza, Shayan Talaei, Ruoxi Sun, Xingchen Wan, Hailong Li, Azalia Mirhoseini, Amin Saberi, and 1 others. 2025b. Reasoning-sql: Reinforcement learning with sql tailored partial rewards for reasoning-enhanced text-to-sql. *CoRR*.

Rafael Rafailov and 1 others. 2023. Direct preference optimization: Your language model is secretly a reward model. *NeurIPS*.

Baptiste Roziere and 1 others. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Ashish Vaswani and 1 others. 2017. Attention is all you need. In *NeurIPS*.

Bing Wang and 1 others. 2025a. Mac-sql: A multi-agent collaborative framework for text-to-sql. In *COLING*.

Pengfei Wang, Baolin Sun, Xuemei Dong, Yaxun Dai, Hongwei Yuan, Mengdie Chu, Yingqi Gao, Xiang Qi, Peng Zhang, and Ying Yan. 2025b. [Agentar-scale-sql: Advancing text-to-sql through orchestrated test-time scaling](#). *Preprint*, arXiv:2509.24403.

David Warren and Fernando Pereira. 1982. An efficient easily adaptable system for interpreting natural language queries. *Computational Linguistics*.

Lukas Wertz, Jasmina Bogojeska, Katsiaryna Mirylenka, and Jonas Kuhn. 2023. Reinforced active learning for low-resource, domain-specific, multi-label text classification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10959–10977.

Lukas Wertz, Katsiaryna Mirylenka, Jonas Kuhn, and Jasmina Bogojeska. 2022. Investigating active learning sampling strategies for extreme multi label text classification. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4597–4605.

Quan Xiao, Debarun Bhattacharjya, Balaji Ganesan, Radu Marinescu, Katsiaryna Mirylenka, Nhan H. Pham, Michael Glass, and Junkyu Lee. 2025. The consistency hypothesis in uncertainty quantification for large language models. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*.

Tao Yu and 1 others. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *EMNLP*.

Daniel M Ziegler and 1 others. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Constrained Decoding Error Analysis

When evaluating constrained decoding (Section 3.3), we observed an overall improvement in execution accuracy but a counter-intuitive spike in the total volume of SQL errors generated by the model.

Table 8 provides a detailed quantitative breakdown of these specific error types.

Table 8: Detailed breakdown of SQL error types generated during unconstrained vs. constrained decoding on the IBM BI Dataset.

Error Type	Unconstr.	Constr.
Undefined Column	11	10
Ambiguous Column	0	8
Undefined Table	3	2
Undefined Function	1	5
Syntax Error	0	9
Division by Zero	0	1

The most significant increases occurred in Syntax Errors (9 vs 0) and Ambiguous Column errors (8 vs 0) compared to unconstrained generation. Analyzing the raw generated queries reveals that the Syntax Errors predominantly stem from queries being cut off by the maximum output length threshold.

Consider the following two examples of failed constrained generations:

```
-- Example 1: Repetition Loop
SELECT product_id, manufacturer, COUNT(
  duns_number) AS c FROM tls.inventory
GROUP BY product_id, product_id,
  product_id, product_id, product_id,
[...]
product_id, product_id, product_id,
  product_id, product

-- Example 2: Runaway WHERE Clause
SELECT tls.client_site.client_name FROM
  tls.client_site JOIN tls.inventory
ON tls.client_site.duns_number = tls.
  inventory.duns_number WHERE
  tls.client_site.coverage_type = 'Data
  Center' AND
  tls.client_site.coverage_name = 'Data
  Center' AND
  tls.client_site.market = 'US' AND tls.
  client_site.geo = 'US'
[...]
tls.client_site.global_buying_group_id
  IS NOT NULL AND
  tls.client_site.global_buying_group_name
  IS NOT NULL AND
  tls.client_site.domestic_buying_group_id
  IS NOT NULL AND
```

In the first example, the constrained LLM gets trapped in a repetitive generation loop. In the second example, it appends an excessive number of unnecessary conditions to the WHERE clause. In both scenarios, the constraining algorithm appears to reduce the probability of the EOS token relative to other valid SQL tokens. Because our context-free

grammar strictly constrains the output syntactically but not semantically, we cannot forcefully trigger an EOS token to terminate the sequence once the logical intent has been satisfied.

Similarly, "Undefined Column" errors persist because our constrained decoding algorithm does not currently enforce that the tables for all columns mentioned in the SELECT clause must eventually be included in the FROM or JOIN clauses. Finally, the "Ambiguous Column" errors generated by the constrained model stem from the fact that the algorithm does not strictly enforce the `<table_name>.<column_name>` notation within JOIN conditions, allowing the LLM to generate ambiguous references when multiple tables share identical column names.

B Fine-tuning Hyperparameters

Specific training hyperparameters included:

- **LoRA:** Rank $r = 16$, Alpha $a = 8$, Dropout 0.05.
- **Optimizer:** AdamW with weight decay 0.1.
- **Learning Rate:** $1e - 5$.
- **Batch Size:** 16 with Gradient Accumulation Steps of 16.
- **Max Input Tokens:** 8192 (8k context window).
- **Epochs:** Models were typically trained for 1 to 3 epochs, with early stopping based on validation loss to prevent overfitting.

Call Support Copilot: A Reproducible Multimodal System for Speech Emotion Recognition, Intent Understanding, and Agent Assistance

Khoshimov Rakhmatillokhon

University of Zurich

Dept. of Informatics

rakhmatillokhon.khoshimov@uzh.ch

Dmitry Rudshin

University of Zurich

Dept. of Informatics

dmitry.rudshin@uzh.ch

Yanyang Luo

University of Zurich

Dept. of Informatics

yanyang.luo@uzh.ch

Abstract

We present Call Support Copilot, a reproducible multimodal system that integrates automatic speech recognition, speech emotion recognition, machine translation, spoken language understanding, and client knowledge retrieval in a single dashboard for customer support agents. Built from publicly accessible pre-trained models and standard benchmarks, the system transcribes speech with Whisper-family ASR (Radford et al., 2023; Klein, 2023), detects caller affect in valence-arousal-dominance terms (Mehrabian, 1996; Russell, 1980), classifies intents from a banking-domain inventory of 77 categories (Casanueva et al., 2020), and retrieves client records from a database. Evaluation shows strong component performance: 6.6% word error rate on LibriSpeech (Panayotov et al., 2015), 91.7% macro-F1 on SUPERB ER session1 (IEMOCAP subset, $n=6$) (Yang et al., 2021; Busso et al., 2008), 42.98 BLEU (Papineni et al., 2002; Post, 2018) for German–English translation, and 87.0% accuracy on BANKING77 intent classification. End-to-end benchmarking of the core pipeline achieves faster-than-real-time throughput with mean real-time factor 0.67–0.71. All model identifiers, configurations, and evaluation scripts are documented in the accompanying repository, supporting reproducibility in line with the SwissText 2026 theme.

1 Introduction

Customer service call centers remain essential infrastructure for financial institutions, healthcare providers, and commercial enterprises (Gao et al., 2019). The quality of these interactions depends critically on human agents’ ability to understand callers quickly, respond to their emotional states appropriately, and access relevant account information without disrupting the conversation flow. Traditional call center technology provides basic telephony functions but offers little support for these

cognitive demands (Ram et al., 2018; Hosseini-Asl et al., 2020).

Recent advances in speech and language processing have created opportunities to address these limitations. Models such as Whisper (Radford et al., 2023) and efficient implementations such as faster-whisper (Klein, 2023) achieve robust speech recognition across diverse acoustic conditions, building on foundational work in self-supervised speech representation learning (Baevski et al., 2020; Hsu et al., 2021). Speech emotion recognition systems trained on corpora like IEMOCAP (Busso et al., 2008) and MELD (Poria et al., 2019) can detect affective states following the dimensional affect framework (Russell, 1980; Mehrabian, 1996). Neural machine translation enables cross-lingual communication (Fan et al., 2021; Junczys-Dowmunt et al., 2018), while intent classification models trained on domain-specific datasets such as BANKING77 (Casanueva et al., 2020) can identify caller needs from transcribed speech, extending work in spoken language understanding (Young et al., 2013).

However, most published systems focus on individual components rather than integrated pipelines, and reproducibility remains a significant challenge in NLP research (Belz et al., 2021; Dodge et al., 2019).

This paper presents Call Support Copilot, a system that combines these technologies in an integrated dashboard for customer support agents. Our contributions are twofold. First, we present a **reproducible** implementation assembled from publicly accessible model checkpoints and resources, with model identifiers, configurations, and evaluation scripts documented in an accompanying repository. Second, we provide quantitative evaluation across all major system components using established benchmarks, showing that strong performance is achievable with an accessible, modular architecture without training ASR, MT, or SER models from scratch. In line with the SwissText 2026 theme of

Reproducible NLP, we focus on a pipeline that can be replicated end-to-end using publicly accessible resources.

2 Related Work

Call-center assistance systems are closely related to task-oriented dialogue and conversational AI, where systems combine speech or text understanding with dialogue-state tracking, retrieval, and response or action selection (Young et al., 2013; Gao et al., 2019; Ram et al., 2018). Datasets such as MultiWOZ (Budzianowski et al., 2018) and BANKING77 (Casanueva et al., 2020) support evaluation of dialogue and intent-understanding modules, but they do not by themselves evaluate the full operational chain from audio input to agent-facing recommendations.

Our system follows a modular integration strategy rather than proposing a new model architecture. This places it between component-level benchmark work on ASR (Radford et al., 2023; Panayotov et al., 2015), MT (Junczys-Dowmunt et al., 2018; Post, 2018), SER (Schuller, 2018; Latif et al., 2021), and intent classification (Casanueva et al., 2020), and applied systems that combine these components in a practical workflow. The comparison to Casanueva et al. (Casanueva et al., 2020) anchors the SLU result against a published BANKING77 baseline, while the other reported metrics are intended as reproducible component checks rather than claims of state-of-the-art performance.

Speech emotion recognition has particular reproducibility and validity concerns. IEMOCAP is widely used, but recent analyses highlight issues around modality dependence, recording quality, ambiguous labels, and misclassifications that may be unsurprising even to human annotators (Probol and Mieskes, 2023). We therefore treat the IEMOCAP-derived SER result as a limited benchmark sanity check, not as evidence that the system generalizes to spontaneous customer-support speech.

3 System Architecture

Call Support Copilot processes audio input through five interconnected modules (Figure 1).

Speech Recognition. Audio recordings are processed through Whisper (Radford et al., 2023) using faster-whisper (Klein, 2023) with CTranslate2 and int8 quantization, building on advances in self-supervised speech representations (Baevski

et al., 2020; Hsu et al., 2021; Graves et al., 2006). Voice activity detection (Silero Team, 2021) filters silence. The system supports WAV, MP3, M4A, FLAC, and OGG formats with FFmpeg conversion.

Machine Translation. When the detected language differs from English, transcripts pass through Helsinki-NLP MarianMT (Junczys-Dowmunt et al., 2018) transformer models (Vaswani et al., 2017) with SentencePiece tokenization (Kudo and Richardson, 2018) trained on OPUS parallel data (Tiedemann, 2012). Six source languages are supported: German, Dutch, French, Spanish, Italian, and Portuguese.

Speech Emotion Recognition. Audio is analyzed using MERaLiON-SER-v1 (MERaLiON Team, 2025), outputting categorical emotions following Ekman (Ekman, 1992) and PAD dimensions (Mehrabian, 1996; Russell, 1980). Seven categories are classified: neutral, happy, sad, angry, fearful, disgusted, and surprised. A sliding window approach (4 s windows, 2 s overlap) handles varying durations (Schuller, 2018), producing emotion timelines with deep learning representations (Latif et al., 2021).

Spoken Language Understanding. Intent classification uses DistilBERT (Sanh et al., 2019), a knowledge-distilled variant of BERT (Devlin et al., 2019), fine-tuned on BANKING77 (Casanueva et al., 2020) with 77 intent categories. Low-confidence predictions are flagged as out-of-domain (Larson et al., 2019). Slot extraction uses regex patterns for amounts, dates, reference IDs, and merchant names, tracking missing required slots (Henderson et al., 2014).

Action Generation. Caller phone numbers are used to query a client database. The system then combines intent predictions, emotion states, extracted slots, and client records to generate suggested actions (Gao et al., 2019), including intent-specific recommendations, slot-based follow-ups, and emotion-triggered de-escalation guidance, following task-oriented dialogue practice (Young et al., 2013; Budzianowski et al., 2018).

4 Evaluation Setup

We evaluate the integrated system with fixed model identifiers and released repository scripts, matching the configurations used to generate the bundled result artifacts. For ASR and MT, we report mean per-

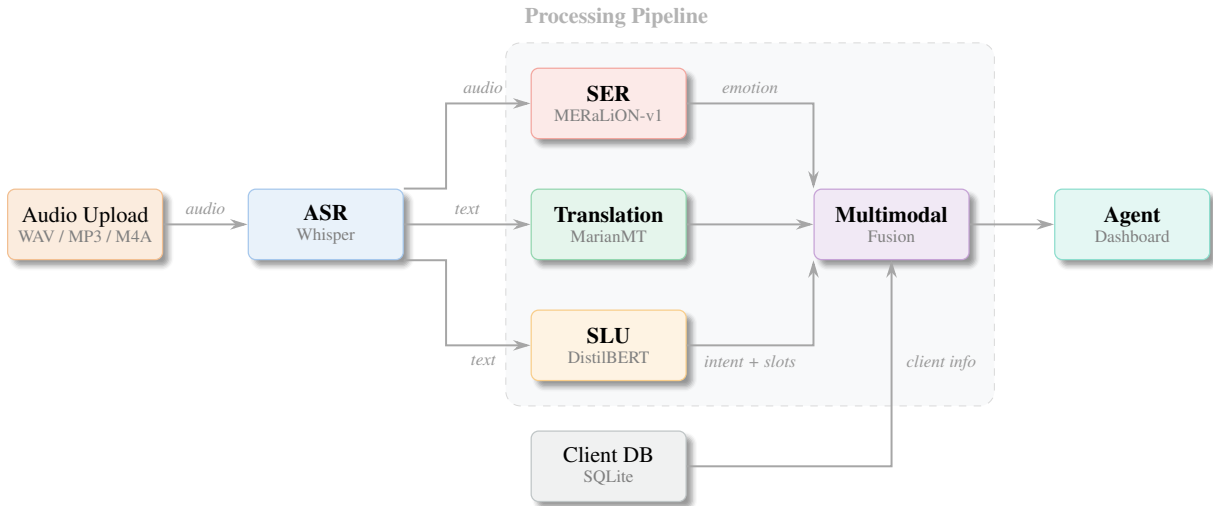


Figure 1: Call Support Copilot system architecture. Uploaded audio flows through ASR (Whisper) which feeds three parallel branches: Speech Emotion Recognition (SER) for affective state detection, Machine Translation (MT) for multilingual support, and Spoken Language Understanding (SLU) for intent classification and slot extraction. These signals are fused with client records retrieved from the knowledge base to generate context-aware recommendations displayed on the agent dashboard.

formance over 50 randomly sampled items with a fixed seed from LibriSpeech test-clean (Panayotov et al., 2015) and WMT16 German–English test data. For SER, we use the SUPERB ER session1 split (Yang et al., 2021), which exposes a small IEMOCAP-derived four-class benchmark (Busso et al., 2008) in our workflow. For latency, we run the full pipeline on LibriSpeech and SUPERB ER utterances and record wall-clock times for ASR, SER, MT, and database lookup.

The reported component scores are measured at component boundaries. ASR WER is computed from audio against reference transcripts, MT BLEU is computed on WMT text pairs, SER macro-F1 is computed on the SUPERB ER audio examples, and SLU accuracy is computed on BANKING77 text rather than on ASR-generated transcripts. The end-to-end experiment measures runtime through the actual pipeline, but it does not estimate downstream accuracy after ASR error propagation. This distinction is important: the deployed dashboard does pass ASR output to downstream modules, but the present paper reports technical component benchmarks and pipeline latency rather than a full cascaded-error evaluation.

Metrics are chosen to reflect standard practice for each module. ASR uses word error rate (WER), MT uses corpus BLEU (Papineni et al., 2002; Post, 2018), SER uses macro-F1 over the four mapped emotion labels, and SLU uses accuracy and macro-F1 together with per-intent precision, recall, and F1.

End-to-end efficiency is summarized with real-time factor (RTF), where values below 1 indicate faster-than-real-time processing. Table 1 summarizes the core results.

5 Results and Discussion

Speech Recognition. On LibriSpeech test-clean (Panayotov et al., 2015), our system achieves **6.6% WER** (95% CI [4.4%, 9.2%]) on 50 samples. Errors arise mainly from proper nouns and archaic vocabulary. This is competitive with reported Whisper performance on clean speech (Radford et al., 2023).

Machine Translation. German-to-English achieves **42.98 BLEU** (Papineni et al., 2002; Post, 2018) on 50 sentence pairs with 0.86 s mean inference time, in line with Helsinki-NLP MarianMT benchmarks on WMT data (Junczys-Dowmunt et al., 2018; Tiedemann, 2012).

Speech Emotion Recognition. On SUPERB ER session1 (IEMOCAP subset, four-class labels, $n=6$) (Yang et al., 2021; Busso et al., 2008), full-audio processing achieves **91.7% macro-F1**. Table 2 shows dimensional predictions consistent with the PAD model (Mehrabian, 1996; Russell, 1980).

Intent Classification. On BANKING77 (Casanueva et al., 2020), DistilBERT (Sanh et al., 2019) achieves **87.0% accuracy** and 0.863 macro-F1. Table 3 shows per-category results for

Component	Dataset	Metric	Value
ASR	LibriSpeech	WER	6.61%
MT	WMT DE-EN	BLEU	42.98
SER	SUPERB ER	Macro-F1	0.917
SLU	BANKING77	Accuracy	87.0%
System	Libri+SUPERB	RTF	0.67-0.71

Table 1: Summary of evaluation results across all system components on standard benchmarks, plus end-to-end real-time factor (RTF) for the core pipeline.

Emotion	V	A	D
Happy	0.85	0.70	0.68
Neutral	0.47	0.48	0.52
Sad	0.24	0.08	0.18
Angry	0.28	0.84	0.75

Table 2: Mean valence (V), arousal (A), and dominance (D) predictions for each emotion category on IEMO-CAP.

key banking intents. For comparison, Casanueva et al. report 85.19% with USE+ConveRT on this benchmark (Casanueva et al., 2020).

End-to-End Latency. We benchmark the core pipeline (ASR + SER + optional MT + DB lookup). Real-time factor (RTF) values below 1 indicate faster-than-real-time throughput. On LibriSpeech test-clean ($n=10$), mean RTF is **0.71**; on SUPERB ER ($n=6$), mean RTF is **0.67**. SER dominates latency due to sliding-window analysis.

The results show that the architecture is strongest where mature pretrained models can be integrated with limited task-specific adaptation. ASR and MT remain within expected benchmark ranges despite the lightweight engineering stack, and the SLU component improves on the 85.19% BANKING77 USE+ConveRT baseline reported by Casanueva et al. (Casanueva et al., 2020). This supports the core claim of the paper: a practical customer-support copilot can be assembled from publicly accessible building blocks without retraining the full stack.

The evaluation also clarifies the current bottlenecks. SER contributes the largest share of end-to-end latency because sliding-window inference scales with utterance duration, and the reported SER score is derived from a very small SUPERB ER sample in the released results. The runtime numbers are therefore encouraging for uploaded-call processing, but they do not yet establish readiness for live streaming deployments. Likewise, intent prediction is strong on BANKING77, yet real customer calls remain harder because ASR errors,

Intent Category	P	R	F1
transaction_charged_twice	.889	1.00	.941
lost_or_stolen_card	.804	.925	.860
refund_not_showing_up	.947	.900	.923
card_arrival	.756	.850	.800
activate_my_card	1.00	.925	.961
verify_my_identity	.745	.875	.805
<i>Macro avg (77 cls)</i>	.875	.870	.863

Table 3: Intent classification results for selected banking intents on the BANKING77 test set (Casanueva et al., 2020).

spontaneous speech, and slot omissions compound downstream uncertainty.

An important observation is that the component scores should not be interpreted independently of pipeline coupling. In the deployed interface, translation quality depends directly on ASR transcription fidelity, and intent classification inherits both lexical errors and disfluencies from the upstream recognizer. This means that the strongest standalone module is not automatically the most useful one operationally: modest ASR degradation on accented or noisy calls can propagate into wrong intent predictions or misleading summaries even when the classifier remains strong on clean BANKING77 text. For customer-support settings, cross-component robustness is therefore at least as important as any single benchmark number.

6 Reproducibility

In the spirit of reproducible NLP (Belz et al., 2021; Dodge et al., 2019), all core models and resources used in the system are publicly accessible:

- **ASR:** Systran/faster-whisper-base via faster-whisper (Klein, 2023) with int8 quantization
- **MT:** Helsinki-NLP MarianMT opus-mt-{de,nl,fr,es,it,pt}-en
- **SER:** MERaLiON-SER-v1 (MERaLiON Team, 2025)
- **SLU:** distilbert-base-uncased fine-tuned on BANKING77
- **VAD:** Silero VAD (Silero Team, 2021)

The system is implemented in Python using Streamlit, PyTorch, and HuggingFace Transformers. Audio preprocessing converts input to 16 kHz mono using soundfile and FFmpeg. The client database uses SQLite. All

evaluation scripts, result files, and model configurations are included in the accompanying repository released with the paper: <https://github.com/rakhmatillokhon-khoshimov/call-support-copilot>.

To make the submission auditable rather than merely runnable, the repository records model identifiers, quantization choices, dataset slices, and summary CSV/JSON outputs for each experiment. This is especially relevant for a modular system, where implementation choices such as VAD preprocessing, CPU quantization, or label remapping can materially affect downstream scores. By exposing these settings together with the measured outputs, the paper makes it possible to verify not only that the code executes, but also that the reported benchmark numbers correspond to a specific, inspectable evaluation protocol.

7 Conclusion

We presented Call Support Copilot, a reproducible multimodal system integrating speech recognition (Radford et al., 2023; Klein, 2023), emotion detection (MERaLiON Team, 2025), translation (Junczys-Dowmunt et al., 2018), and intent understanding (Casanueva et al., 2020) to assist customer support agents. Evaluation on established benchmarks shows strong component performance with practical latency, achieved without training ASR, MT, or SER models from scratch. The modular implementation supports both practical deployment and research extension, and aligns with the goal of making NLP systems reproducible (Belz et al., 2021).

Limitations

Our evaluation samples are small due to computational constraints (50 samples for ASR and MT, and 10/6 samples for latency on LibriSpeech/SUPERB ER). The SER model, trained on acted IEMOCAP speech (Busso et al., 2008), may not generalize to spontaneous customer-service conversations (Schuller, 2018; Poria et al., 2019). IEMOCAP-derived evaluation should also be interpreted cautiously because prior work reports issues around recording quality, task ambiguity, and the relative strength of text-only emotion models (Probol and Mieskes, 2023). Slot extraction patterns are English-specific and rely on hand-crafted regex rules rather than learned extractors. The supported MT setup is limited to six Western and Central Eu-

ropean source languages (German, Dutch, French, Spanish, Italian, and Portuguese), excluding many low-resource, non-European, and Swiss-relevant language varieties. The current system processes uploaded recordings rather than streaming live audio, which limits real-time deployment scenarios. ASR may exhibit demographic disparities for accented speech (Ardila et al., 2020), and emotion recognition may reflect cultural biases in the training data (Ekman, 1992). Future work includes streaming ASR with incremental emotion updates, speaker diarization, CRM integration, broader language coverage, and large language models (Brown et al., 2020) for more flexible action generation.

The present study evaluates technical components rather than end-user outcomes. We do not measure whether agents resolve issues faster, make fewer mistakes, or perceive the dashboard as trustworthy under realistic workload. Likewise, the database layer is demonstrated on a controlled SQLite setup rather than in a production CRM environment with authentication, logging, and policy constraints. These omissions matter because deployment quality depends not only on benchmark performance, but also on usability, integration overhead, and organizational acceptance. A stronger follow-up study should therefore pair component benchmarks with human-centered evaluation in a realistic support workflow.

References

- Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222.
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460.
- Anya Belz, Simon Mille, and David M Howcroft. 2021. ReproGen: A first step towards a shared task on reproducibility of human evaluations in NLG. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 338–350.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-

- shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.
- Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ – a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026.
- Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeanette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359.
- Iñigo Casanueva, Tadas Temčinas, Daniela Gerber, Milica Gasic, and Matthew Henderson. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.
- Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A Smith. 2019. Show your work: Improved reporting of experimental results. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 2185–2194.
- Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, and 1 others. 2021. Beyond English-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.
- Jianfeng Gao, Michel Galley, and Lihong Li. 2019. Neural approaches to conversational AI. *Foundations and Trends in Information Retrieval*, 13:127–298.
- Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 369–376.
- Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 263–272.
- Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems*, volume 33, pages 20179–20191.
- Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.
- Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, and 1 others. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121.
- Guillaume Klein. 2023. [faster-whisper](#). GitHub repository.
- Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.
- Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, and 1 others. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 1311–1316.
- Siddique Latif, Rajib Rana, Sara Khalifa, Raja Jurdak, Junaid Qadir, and Björn W Schuller. 2021. Survey of deep representation learning for speech emotion recognition. *IEEE Transactions on Affective Computing*, 14(2):1634–1654.
- Albert Mehrabian. 1996. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, 14(4):261–292.
- MERaLiON Team. 2025. MERaLiON-SER: Robust speech emotion recognition model for english and SEA languages. *arXiv preprint arXiv:2511.04914*.
- Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. LibriSpeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5206–5210.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, and Erik Cambria. 2019. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 527–536.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Nadine Probol and Margot Mieskes. 2023. [Emotions in spoken language – do we need acoustics?](#) In *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 71–84, Toronto, Canada. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, pages 28492–28518.

Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nuber, Bo Xu, Ming Yang, and 1 others. 2018. Conversational AI: The science behind the Alexa prize. *arXiv preprint arXiv:1801.03604*.

James A Russell. 1980. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Björn W Schuller. 2018. Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends. *Communications of the ACM*, 61(5):90–99.

Silero Team. 2021. Silero VAD: Pre-trained enterprise-grade voice activity detector. GitHub repository.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 2214–2218.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Shu-Wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Lai, Kushal Lakhota, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankun Chang, Guan-Ting Lin, and 1 others. 2021. SUPERB: Speech processing universal PERFORMANCE benchmark. *arXiv preprint arXiv:2105.01051*.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. 2013. POMDP-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179.

A Supported Translation Languages

Source Lang.	Model	Avg. Time (s)
German (DE)	opus-mt-de-en	0.86
Dutch (NL)	opus-mt-nl-en	0.82
French (FR)	opus-mt-fr-en	0.79
Spanish (ES)	opus-mt-es-en	0.81
Italian (IT)	opus-mt-it-en	0.84
Portuguese (PT)	opus-mt-pt-en	0.83

Table 4: Supported translation language pairs with Helsinki-NLP MarianMT models (Junczys-Dowmunt et al., 2018) and average inference times on CPU.

B Slot Extraction Details

Following standard approaches in spoken language understanding (Henderson et al., 2014), regex patterns extract: amounts with currency symbols (CHF, USD, EUR); dates in ISO, European, and natural formats; reference IDs; card last-four digits; and merchant names.

C Emotion Categories

Following Ekman (Ekman, 1992) and PAD (Mehrabian, 1996; Russell, 1980): Neutral (moderate valence, low arousal), Happy (high valence, moderate arousal), Sad (low valence, low arousal), Angry (low valence, high arousal, high dominance), Fearful (low valence, low dominance), Disgusted (low valence), Surprised (variable valence, high arousal).

D Action Generation Rules

Intent-specific rules following task-oriented dialogue best practices (Young et al., 2013; Budzianowski et al., 2018): Chargeback triggers transaction confirmation and dispute workflow. Lost card triggers freeze, replacement, and fraud review. Angry emotion prepends de-escalation guidance. Missing slots generate follow-up prompts (Henderson et al., 2014).

Text vs. Phoneme Intermediates for Low-Resource Swiss German Text-to-Speech

Reza Kakooee* Vincenzo Timmel Daniel Perruchoud Michael Graber Manfred Vogel
Institute for Data Science, School of Computer Science,
University of Applied Sciences and Arts Northwestern Switzerland
reza.kakooee@fhnw.ch

Abstract

Building text-to-speech (TTS) systems for low-resource languages such as Swiss German is challenging due to limited paired data and the lack of standardized orthography. In practical Swiss settings, user input is typically written in High German, motivating pipelines that map High German text to Swiss German speech via an intermediate representation. We compare three approaches: (i) direct synthesis from High German (DE-TTS), (ii) High German \rightarrow Swiss German text translation followed by synthesis (CH-TTS), and (iii) High German \rightarrow automatically derived fused phoneme conversion followed by synthesis (PH-TTS). Using the SwissDial dataset, we fine-tune two TTS backbones, SpeechT5 and Orpheus, and evaluate the resulting systems with closed-loop STT metrics (WER/SacreBLEU) and human MOS. Objective transcript-overlap metrics reliably penalize PH-TTS but fail to reflect human preference between DE-TTS and CH-TTS. MOS consistently ranks CH-TTS highest for both backbones, with Orpheus achieving near-original quality and showing robustness when training data is halved; notably, under the half-data setting PH-TTS becomes close to DE-TTS, suggesting that phoneme intermediates may be more competitive in lower-resource regimes. Our analysis indicates that the current PH-TTS pipeline is limited by noisy phoneme supervision and representation mismatch, and we outline directions to make phoneme intermediates competitive in low-resource dialect TTS.

1 Introduction

Neural text-to-speech (TTS) has reached high naturalness in well-resourced settings (Le et al., 2023; Borsos et al., 2023; Wang et al., 2023; Du et al., 2024; OpenAI, 2024; Qwen Team, 2026; Google AI for Developers, 2026; ElevenLabs, 2026; Canopy AI, 2025), but building robust systems for low-resource languages and dialects remains difficult due to the limited availability of

paired text–audio data (Chen et al., 2019). Prior work shows that transfer learning and multilingual training can substantially reduce the amount of target-language supervision required, yet performance can still be sensitive to representation choices and data scarcity (Lux et al., 2022).

Swiss German is a particularly challenging target: it is a dialect continuum that is used in spoken form and lacks a standardized orthography. As a result, collecting consistent text labels is difficult, and spelling variation can become a dominant source of noise for text-conditioned speech generation. The SwissDial dataset addresses part of this challenge by providing a parallel multi-dialect corpus with Swiss German audio paired with both Swiss German and Standard (High) German text references (Dogan-Schönberger et al., 2021). In many practical Swiss applications, user input is written in High German, which naturally suggests pipelines that map High German text to Swiss German speech via an intermediate representation.

A common approach is to translate High German into written Swiss German and then synthesize speech from that text. However, because written Swiss German is not standardized, the translation step can introduce variability that is hard for downstream TTS models to resolve, and “correctness” (getting the intended words and meaning) can dominate perceived quality.

This motivates the research question: *should the intermediate representation be Swiss German text at all, or can we benefit from a more acoustically grounded representation such as phonemes?*

Phoneme conditioning can reduce pronunciation ambiguity, but controlled English studies also show that imperfect phonemization introduces practical costs (Fong et al., 2019).

In this work we compare three practical Swiss German TTS pipelines that start from High German input (see Figure 1):

1. **DE-TTS (direct):** High German text → Swiss German speech
2. **CH-TTS (translation):** High German text → Swiss German text → Swiss German speech
3. **PH-TTS (phoneme):** High German text → Swiss German phoneme string → Swiss German speech

For CH-TTS and PH-TTS, the intermediate conversion is implemented using internally fine-tuned T5 models (Raffel et al., 2020).

For PH-TTS, the training data (SwissDial) already contains High German text, Swiss German text, and Swiss German recordings (Dogan-Schönberger et al., 2021). The missing modality is phoneme strings. We therefore derive phoneme supervision from audio using a wav2vec 2.0-based phonemization step (Baevski et al., 2020). The resulting *discrete* phoneme sequences are then converted into more compact *continuous* phoneme strings via a fusion procedure.

We further study the impact of the TTS backbone by fine-tuning two contrasting synthesis models. **SpeechT5** is a unified encoder–decoder model pre-trained across speech and text modalities and serves as a relatively *lightweight* baseline for data-efficient fine-tuning (Ao et al., 2022). In parallel, we evaluate **Orpheus**, a recently released Llama-based “speech-LLM” system that targets high naturalness and expressiveness, serving as a strong synthesis baseline (Canopy AI, 2025). To probe data sensitivity, we train Orpheus on the same data used for SpeechT5, and include an additional Orpheus setting trained on half of the training data.

Finally, we evaluate the synthesized audio quality subjectively. We conduct human listening tests with Mean Opinion Score (MOS), and complement them with a closed-loop objective protocol (text → TTS → audio → STT → transcript) using an internal Whisper model fine-tuned on Swiss German speech (Timmel et al., 2025a,b) that transcribes synthesized audio into High German text for consistent scoring (Radford et al., 2022).

Contributions. Our main contributions are:

- A controlled comparison of three fine-tuned Swiss German TTS pipelines (direct, translation-based, automatically derived phoneme-based) under a shared evaluation harness.

- A practical phoneme-supervision construction pipeline for Swiss German (audio-based phonemization + discrete→continuous fusion) enabling phoneme-intermediate TTS training from SwissDial-style annotations.
- An empirical comparison of a lightweight backbone (SpeechT5) versus a speech-LLM backbone (Orpheus), including a dataset scaling experiment (full vs. half data) to quantify robustness under reduced supervision.
- An evaluation setup combining MOS with closed-loop ASR scoring via Whisper-based transcription into High German.

2 Related Work

TTS for low-resource languages. Neural TTS typically requires substantial paired text–audio data, which is scarce for many languages and dialects. A common strategy is cross-lingual transfer or multilingual training, often with shared symbol spaces to improve data efficiency (Lux et al., 2022). This motivates studying which intermediate representations and model choices remain robust under limited training data.

Phoneme-based TTS and phonemes vs. graphemes across languages. Phoneme conditioning is widely used in neural TTS to make pronunciation explicit, which can be beneficial when orthography is irregular, when pronunciation must be controlled, or when models are transferred across languages; the main cost is reliance on a lexicon and/or grapheme-to-phoneme (G2P) conversion, whose errors propagate to synthesis. In a controlled English study, Fong et al. (2019) compare letter-input and phone-input sequence-to-sequence TTS and show that phone inputs can reduce pronunciation errors, while also quantifying how imperfect phonemes degrade output quality. For languages with more regular spelling, the gap can shrink: Perquin et al. (2021) find that grapheme inputs can match phoneme inputs on a curated French dataset, suggesting that phoneme advantages depend on language and data conditions. In German and German varieties, phoneme (or phoneme/grapheme) tokenization remains common in practical pipelines (Govalkar et al., 2021), and dialect synthesis can combine Standard German G2P with dialect embeddings to handle regional variation under limited resources (Gutscher et al., 2023). In low-resource settings,

phoneme-based transfer learning often relies on mapping phoneme inventories across languages to reduce mismatch (Do et al., 2022), and few-shot cross-lingual TTS can be improved by learning transferable phoneme embeddings in a shared latent space (Huang et al., 2022). More recently, multilingual pretraining directly on phoneme sequences has been proposed to improve data efficiency for TTS in low-resource languages (Nguyen et al., 2023).

TTS backbone model choices. Modern TTS spans diverse architectures (autoregressive, non-autoregressive, flow/diffusion, and increasingly speech generation with large language models). We focus on two complementary backbones. **SpeechT5** is a unified encoder–decoder model pre-trained across speech and text modalities and is attractive as a relatively *lightweight* foundation for data-efficient fine-tuning (Ao et al., 2022). In contrast, **Orpheus** represents a newer *speech-LLM* direction, using a Llama-based backbone for highly natural and expressive speech generation; we include it as a strong, high-quality synthesis baseline to study trade-offs against a compact pre-trained encoder–decoder (Canopy AI, 2025).

3 Method

3.1 Problem setup and pipeline variants

Swiss German TTS is challenging because Swiss German has no standardized orthography and is primarily used in speech. In many Swiss applications, users provide **High German** text, and the system must synthesize **Swiss German** speech. Following this practical setting, we compare three pipeline variants as shown in Figure 1 that differ in the intermediate representation used before speech synthesis.

All variants share the same evaluation harness and differ only in their intermediate conversion and synthesis backbone.

3.2 Data

We use the SwissDial dataset, a parallel multi-dialect corpus of spoken Swiss German with aligned **Swiss German audio** and corresponding **Swiss German** and **Standard (High) German** text (Dogan-Schönberger et al., 2021). This means the training data already contains the modalities required for DE-TTS and CH-TTS (High German text, Swiss German text, recordings). For PH-TTS,

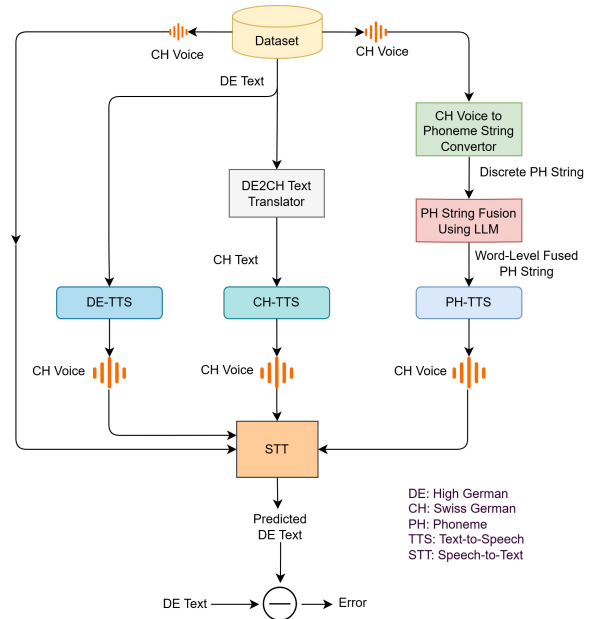


Figure 1: Overview of the three Swiss German TTS pipelines and the closed-loop evaluation. The dataset provides paired Swiss German speech (CH Voice) and High German text (DE Text). DE-TTS synthesises CH Voice directly from DE Text; CH-TTS first translates DE Text to Swiss German text (DE→CH) and then synthesises; PH-TTS derives phoneme supervision from CH Voice via an audio-to-phoneme converter and fuses discrete phonemes into word-level phoneme strings before synthesis. For objective evaluation, all synthesised CH Voice outputs are transcribed by an STT system into predicted DE text and compared against the DE reference to compute error metrics (WER and SacreBLEU).

the missing modality is the **phoneme-string** representation, which we construct from the audio as described next.

3.3 Constructing phoneme supervision from audio

Audio-to-phoneme transcription (discrete sequences). We generate initial phoneme strings from Swiss German recordings using a pre-trained wav2vec2-based phoneme recognizer (Wav2Vec2Phoneme) (Xu et al., 2021; Baeovski et al., 2020). The raw output forms **discrete phoneme sequences**, where unigraph/digraph tokens are separated by excessive whitespace. While these sequences are usable as model inputs, the redundant spacing increases sequence length and inference latency and can reduce contextual coherence for downstream models.

Discrete-to-continuous fusion (word-like units). To obtain a more compact conditioning signal,

Fusion strategy	Prec.	Rec.	Acc.
Dual wav2vec2 alignment	10.7	10.6	69.4
LLM-based fusion	94.9	93.8	98.1
Phoneme model alignment	30.6	30.0	76.3

Table 1: Space-boundary insertion quality on a manually fused validation samples. Scores are reported in percentages.

we convert discrete sequences into **continuous phoneme strings** by removing unnecessary spaces and re-inserting boundaries primarily at word boundaries. We compare three fusion strategies:

1. **LLM-based fusion:** an LLM is prompted to fuse the discrete phonemes into word-like units while following the Swiss German reference text as closely as possible.
2. **Phoneme alignment via eSpeak:** we derive phonemes from the Swiss German text using eSpeak/eSpeak-NG (eSpeak NG Developers, 2016), align them to the audio-derived phonemes, and transfer whitespace/boundary information from the text side.
3. **Dual wav2vec2 alignment (CTC segmentation):** we use forced alignment based on CTC segmentation to obtain timestamps for (i) text characters and (ii) audio-derived phonemes, then transfer word-boundary spacing from Swiss German text into the discrete phoneme stream (Kürzinger et al., 2020; Baevski et al., 2020).

To select the best fusion strategy, we construct a small gold set of 100 manually fused validation examples. Fusion quality is evaluated by treating *space insertion* as a binary decision between consecutive phoneme symbols and reporting precision, recall, and accuracy of boundary placement. As shown in Table 1, the LLM-based fusion strategy performs best, substantially outperforming the alignment-based alternatives. We therefore use the LLM-based strategy to generate the fused phoneme targets for PH-TTS. The same gold set is also used to tune strategy-specific parameters and prompts; while this validation set is limited in size, the performance differences between strategies are already substantial.

Tokenizer compatibility. The resulting continuous phoneme inventory contains symbols not covered by the default SpeechT5 text tokenizer (Ao et al., 2022). Therefore, we map unsupported

phoneme symbols to a set of rarely used “spare” tokens in the tokenizer vocabulary. This enables end-to-end training while keeping the backbone unchanged. In addition, the default SpeechT5 text tokenizer does not cover several German-specific characters (umlauts and the Eszett/sharp s), so we apply a deterministic character mapping before tokenization:

ä -> æ, Ä -> æ
 ö -> é, Ö -> é
 ü -> ê, Ü -> ê
 ß -> æ (and uppercase equivalent)

This normalization is applied consistently during training and inference for all text-conditioned components. For **Orpheus**, we do not face these tokenizer limitations; we therefore use the model’s original tokenizer without additional character remapping or phoneme-symbol substitution.

3.4 Intermediate translation models

For branches that require intermediate conversion at inference time, we fine-tune T5-style sequence-to-sequence models (Raffel et al., 2020). During **training**, we do not rely on these translators, since the dataset already provides the aligned High German text, Swiss German text, and recordings (and phoneme strings are constructed directly from the recordings as described in Section 3.3).

DE→CH (CH-TTS). We fine-tune a T5 translator on SwissDial to map High German input into Swiss German written text, covering the eight dialects represented in the dataset. This translator provides the intermediate representation consumed by the text-conditioned TTS model in the CH-TTS branch.

DE→phoneme (PH-TTS). We fine-tune a T5 model using SwissDial paired with the **fused phoneme strings** constructed from the recordings and fused by LLM as described in Section 3.3. The model maps High German input *directly* into fused (continuous) Swiss German phoneme strings, i.e., the same format used to condition the PH-TTS models. This avoids a separate discrete-to-continuous fusion step at inference time and ensures the translator output matches the representation seen during TTS training.

3.5 Speech synthesis backbones

We evaluate two synthesis backbones under the same three-branch pipeline structure.

SpeechT5. SpeechT5 is a unified encoder-decoder model pre-trained across speech and text modalities (Ao et al., 2022). We fine-tune SpeechT5 to synthesize Swiss German speech from (i) High German text (DE-TTS), (ii) Swiss German text (CH-TTS), or (iii) fused phoneme strings (PH-TTS), using the same paired supervision from SwissDial (Dogan-Schönberger et al., 2021).

Orpheus. Orpheus is an open-source Llama-based “speech-LLM” TTS system targeting highly natural and expressive synthesis (Canopy AI, 2025). To compare robustness and data-efficiency, we train three Orpheus variants on the same dataset used for SpeechT5, and additionally train one Orpheus variant on half of the training data.

3.6 Objective and subjective evaluation

We evaluate both perceived quality and content preservation.

Closed-loop objective evaluation. We implement a closed-loop protocol: *reference text* → *TTS* → *audio* → *STT* → *transcript*, then compute **WER** and **SacreBLEU** between the transcript and the reference. For STT we use an internal Whisper model fine-tuned on Swiss German speech; synthesized audio is transcribed into **High German** to enable a unified scoring space across all branches (Timmel et al., 2025a; Radford et al., 2022).

Human listening tests (MOS). We run Mean Opinion Score (MOS) evaluations where raters judge short audio samples on a 5-point scale for overall naturalness/quality, comparing original recordings against DE-TTS, CH-TTS, and PH-TTS outputs.

4 Results

We report closed-loop STT-based metrics (WER↓, SacreBLEU↑) and human MOS for DE-TTS, CH-TTS, and PH-TTS. While the objective metrics are useful diagnostics, they do not reliably capture the perceptual ranking between DE-TTS and CH-TTS, motivating MOS as the primary signal for comparison.

4.1 Objective evaluation (closed-loop WER / SacreBLEU)

4.1.1 SpeechT5

Table 2 reports average WER and SacreBLEU for SpeechT5. First, note that the *original recordings* do not yield perfect transcript agreement with the

High German reference, indicating non-trivial STT variability even on real audio. Among the synthesised conditions, PH-TTS is clearly worse (higher WER, lower SacreBLEU). In contrast, DE-TTS and CH-TTS are comparatively close under these transcript-based metrics, and DE-TTS appears best by objective scores.

4.1.2 Orpheus: full vs. half dataset

Table 3 shows the same objective metrics for Orpheus trained on the full dataset and on half of the dataset. The same pattern holds: PH-TTS is strongly penalised, while DE-TTS and CH-TTS are harder to separate and remain relatively close in score. Reducing the dataset size affects DE-TTS more noticeably than CH-TTS under these objective metrics.

Why objective metrics are insufficient. Across both backbones, the objective metrics clearly identify PH-TTS as the weakest branch, but they do not provide a clean separation between DE-TTS and CH-TTS. In particular, DE-TTS tends to score best because the evaluation reference is High German text and DE-TTS is optimised to preserve that surface form. CH-TTS, however, includes an explicit DE→CH step and aims to produce more dialect-appropriate Swiss German content before synthesis, which can change lexical choices and reduce transcript overlap even when listeners prefer the result. This motivates the subjective evaluation below.

4.2 Subjective evaluation (MOS)

4.2.1 SpeechT5

Table 4 reports MOS for SpeechT5 (5-point scale). CH-TTS is closest to the original recordings, DE-TTS is lower, and PH-TTS is lowest. Importantly, this perceptual ranking differs from the objective metrics above, which favour DE-TTS. This mismatch reinforces that listening tests are necessary when comparing pipelines with different intermediate representations.

4.2.2 Orpheus and dataset scaling

Table 5 reports aggregated MOS for Orpheus across 4 participants. With the **full dataset**, CH-TTS is the best synthesised condition (4.67), followed by DE-TTS (3.80) and PH-TTS (3.43), while original recordings remain highest overall (4.86). With **half the dataset**, CH-TTS remains comparatively stable (4.56), whereas DE-TTS drops more substantially (3.26). PH-TTS also decreases (3.25) and becomes

System	WER (synth)	sBLEU (synth)
Original (vs. DE ref)	0.235	0.607
CH-TTS	0.237	0.607
DE-TTS	0.214	0.657
PH-TTS	0.355	0.471

Table 2: SpeechT5 closed-loop objective results (synth vs. High German reference; across 316 samples). WER↓ and SacreBLEU↑ (sBLEU).

System	WER (full)	WER (half)	sBLEU (full)	sBLEU (half)
Original (vs. DE ref)		0.239		0.611
CH-TTS	0.284	0.285	0.556	0.554
DE-TTS	0.254	0.316	0.603	0.540
PH-TTS	0.469	0.496	0.365	0.344

Table 3: Orpheus closed-loop objective results (synth vs. High German reference). Original baseline is identical across runs (across 312 samples).

Condition (SpeechT5)	MOS
Original	4.04 (119)
CH-TTS	4.00 (124)
DE-TTS	3.53 (127)
PH-TTS	3.10 (124)

Table 4: SpeechT5 MOS results (higher is better); over 6 participants. Numbers in parentheses denote the number of rated samples per condition.

Condition (Orpheus)	Full data	Half data
Original	4.86 (66)	–
CH-TTS	4.67 (72)	4.56 (68)
DE-TTS	3.80 (66)	3.26 (73)
PH-TTS	3.43 (76)	3.25 (76)

Table 5: Orpheus MOS results aggregated over 4 participants. Higher is better. Numbers in parentheses denote the number of rated samples per condition.

essentially indistinguishable from DE-TTS under reduced data.

Effect of halving the dataset (Orpheus). Halving training data has the smallest effect on CH-TTS ($\Delta = -0.11$), while DE-TTS shows the largest drop ($\Delta = -0.54$). PH-TTS also degrades ($\Delta = -0.18$), and its separation from DE-TTS largely vanishes in the half-data setting (0.37 on full data vs. 0.01 on half data).

Takeaway. PH-TTS is consistently weakest across both objective and subjective evaluation. For DE-TTS vs. CH-TTS, however, the STT-based metrics tend to favor DE-TTS and provide only a limited basis for ranking systems that differ in intermediate representation. In contrast, MOS consistently prefers CH-TTS. This mismatch suggests that, in Swiss German TTS where intermediate representations can legitimately change lexical choices and surface form, transcript-overlap metrics should

be treated primarily as diagnostics rather than the main selection criterion.

5 Discussion

Why subjective evaluation is necessary. Our closed-loop STT-based metrics (WER and SacreBLEU) consistently rank PH-TTS lowest, but they fail to capture the human preference difference between DE-TTS and CH-TTS. This is expected because the objective reference is *High German* text: DE-TTS is incentivised to preserve High German surface form, whereas CH-TTS intentionally alters lexical choices through the DE→CH translation step to produce more dialect-appropriate content. As a result, transcript-overlap metrics conflate “quality” with “literal matching” and provide an incomplete basis for ranking systems with different intermediate representations. This mismatch makes subjective evaluation essential for model selection in the Swiss German setting.

CH-TTS as the strongest pipeline across backbones. Across both TTS backbones, MOS consistently ranks CH-TTS as the best synthesized system, suggesting that perceived quality is dominated by dialect-appropriate wording and meaning preservation rather than strict High-German transcript match. In practice, this supports a pipeline view in which the intermediate representation is the primary driver of user-perceived quality: translating into Swiss German text before synthesis improves overall naturalness and intelligibility as judged by listeners.

Backbone effects: SpeechT5 vs. Orpheus. Compared to SpeechT5, Orpheus achieves higher MOS overall and keeps CH-TTS close to original

recordings, indicating that a strong synthesis backbone can substantially improve naturalness once a suitable intermediate representation is provided. However, the phoneme branch remains behind even with Orpheus, implying that the main bottleneck for PH-TTS is not the synthesis model itself, but the representation and supervision pipeline used to obtain phoneme conditioning.

Data scarcity and robustness. The dataset scaling experiment highlights an additional trade-off. Halving the dataset hurts DE-TTS much more than CH-TTS for Orpheus, suggesting that the translation-based intermediate provides a stronger inductive bias and better robustness under reduced supervision. This also hints that, under scarce data, conditioning on a representation that is closer to the target dialect (Swiss German text) can be more helpful than conditioning directly on High German, even if the latter yields higher transcript overlap against a High German reference.

Fairness of the phoneme comparison. Importantly, PH-TTS should not be interpreted as an oracle phoneme condition: unlike the text-based intermediates available in SwissDial, the phoneme strings are automatically derived from audio and therefore introduce additional supervision noise.

Why PH-TTS underperforms. PH-TTS likely suffers from compounded upstream noise and representation mismatch. In our training pipeline, phoneme strings are inferred from audio and then transformed via discrete-to-fused conversion. Errors introduced by audio phonemization and fusion propagate directly into the conditioning signal seen during training, turning the “phonemes as cleaner supervision” hypothesis into a noisier target in practice. In addition, phoneme symbol coverage and tokenization constraints (notably for SpeechT5) can further degrade conditioning fidelity.

Outlook for phoneme-based Swiss German TTS. The most promising next step for PH-TTS is to remove the discrete→fused phoneme noise at the source. Concretely, this could involve training a model that predicts *fused* phoneme strings directly, adopting a phoneme-native tokenizer, and scaling with more (or cleaner) supervision. These changes would test the underlying hypothesis—that phonemes provide a more stable intermediate than non-standard orthography—without the current error-compounding bottlenecks.

Implications for deployment. For product deployment, CH-TTS with Orpheus appears to offer the best trade-off: it aligns with typical user behavior (input in High German), yields near-original MOS, and remains robust when data is limited. Improving the High German→Swiss German translation component is therefore a high-leverage direction, since translation quality largely determines the content/wording that downstream TTS must realize. *In contrast*, PH-TTS is not yet competitive in perceived quality, but it remains a promising direction if the phoneme supervision and tokenization issues can be addressed.

Acknowledgements

Funding: This study was funded by grants from the Swiss Data Innovation Alliance (Project 04.04.2025-12) and Innosuisse (Project 127.910 INNO-ICT).

References

- Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, Yao Qian, Jinyu Li, and Furu Wei. 2022. [SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5723–5738, Dublin, Ireland. Association for Computational Linguistics.
- Alexei Baeovski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. [wav2vec 2.0: A framework for self-supervised learning of speech representations](#). *Preprint*, arXiv:2006.11477.
- Zalán Borsos, Matt Sharifi, Damien Vincent, Arun Babu, Yongkeun Kim, Bowen Shi, Adam Polyak, Da-Cheng Ju, Aaron van den Oord, and Karen Simonyan. 2023. [Soundstorm: Efficient parallel audio generation](#). *Preprint*, arXiv:2305.09636.
- Canopy AI. 2025. Orpheus-tts. <https://github.com/canopyai/Orpheus-TTS>. Open-source Speech-LLM TTS; accessed 2026-02-26.
- Yuan-Jui Chen, Tao Tu, Cheng-chieh Yeh, and Hung-Yi Lee. 2019. [End-to-End Text-to-Speech for Low-Resource Languages by Cross-Lingual Transfer Learning](#). In *Interspeech 2019*, pages 2075–2079.
- Phat Do, Matt Coler, Jelske Dijkstra, and Esther Klappers. 2022. [Text-to-speech for under-resourced languages: Phoneme mapping and source language selection in transfer learning](#). In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 16–22, Marseille, France. European Language Resources Association.

- Pelin Dogan-Schönberger, Julian Mäder, and Thomas Hofmann. 2021. [Swissdial: Parallel multidialectal corpus of spoken swiss german](#). *Preprint*, arXiv:2103.11401.
- Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, Zhifu Gao, and Zhijie Yan. 2024. [Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens](#). *Preprint*, arXiv:2407.05407.
- ElevenLabs. 2026. Eleven multilingual v2. <https://elevenlabs.io/blog/eleven-multilingual-v2>. Blog post; accessed 2026-03-04.
- eSpeak NG Developers. 2016. [espeak ng: Open source speech synthesizer](#).
- Jason Fong, Jason Taylor, Korin Richmond, and Simon King. 2019. [A Comparison of Letters and Phones as Input to Sequence-to-Sequence Models for Speech Synthesis](#). In *10th ISCA Workshop on Speech Synthesis (SSW 10)*, pages 223–227.
- Google AI for Developers. 2026. Text-to-speech generation (tts) — gemini api documentation. <https://ai.google.dev/gemini-api/docs/speech-generation>. Documentation; accessed 2026-03-04.
- Prachi Govalkar, Ahmed Mustafa, Nicola Pia, Judith Bauer, Metehan Yurt, Yigitcan Ozer, and Christian Dittmar. 2021. [A lightweight neural tts system for high-quality german speech synthesis](#). In *14th ITG Conference on Speech Communication*.
- Lorenz Gutscher, Michael Pucher, and Víctor Garcia. 2023. [Neural speech synthesis for austrian dialects with standard german grapheme-to-phoneme conversion and dialect embeddings](#). In *2nd Annual Meeting of the ELRA/ISCA SIG on Under-resourced Languages (SIGUL 2023)*, pages 68–72.
- Wei-Ping Huang, Po-Chun Chen, Sung-Feng Huang, and Hung-yi Lee. 2022. [Few-shot cross-lingual tts using transferable phoneme embedding](#). In *Inter-speech 2022*, pages 4566–4570.
- Ludwig Kürzinger, Dominik Winkelbauer, Lujun Li, Tobias Watzel, and Gerhard Rigoll. 2020. [Ctc-segmentation of large corpora for german end-to-end speech recognition](#). In *Speech and Computer*, volume 12335 of *Lecture Notes in Computer Science*, pages 267–278, Cham. Springer.
- Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. 2023. [Voicebox: Text-guided multilingual universal speech generation at scale](#). *Preprint*, arXiv:2306.15687. NeurIPS 2023.
- Florian Lux, Julia Koch, and Ngoc Thang Vu. 2022. [Low-resource multilingual and zero-shot multi-speaker TTS](#). In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 741–751, Online only. Association for Computational Linguistics.
- Linh The Nguyen, Thinkh Pham, and Dat Quoc Nguyen. 2023. [Xphonebert: A pre-trained multilingual model for phoneme representations for text-to-speech](#). *Preprint*, arXiv:2305.19709.
- OpenAI. 2024. [Gpt-4o system card](#). *Preprint*, arXiv:2410.21276.
- Antoine Perquin, Erica Cooper, and Junichi Yamagishi. 2021. [Grapheme or phoneme? an analysis of tacotron’s embedded representations](#). *Preprint*, arXiv:2010.10694.
- Qwen Team. 2026. [Qwen3-tts technical report](#). *Preprint*, arXiv:2601.15621.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. [Robust speech recognition via large-scale weak supervision](#). *Preprint*, arXiv:2212.04356.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Vincenzo Timmel, Claudio Paonessa, Manfred Vogel, Daniel Perruchoud, and Reza Kakooee. 2025a. [Fine-tuning whisper on low-resource languages for real-world applications](#). In *Proceedings of the 10th edition of the Swiss Text Analytics Conference*, pages 57–65.
- Vincenzo Timmel, Manfred Vogel, Daniel Perruchoud, and Reza Kakooee. 2025b. [Swiss parliaments corpus reimagined \(spc_r\): Enhanced transcription with rag-based correction and predicted bleu](#). In *Proceedings of the 10th edition of the Swiss Text Analytics Conference*, pages 149–154.
- Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. 2023. [Neural codec language models are zero-shot text to speech synthesizers](#). *Preprint*, arXiv:2301.02111.
- Qiantong Xu, Alexei Baevski, and Michael Auli. 2021. [Simple and effective zero-shot cross-lingual phoneme recognition](#). *arXiv preprint arXiv:2109.11680*.

Extending the Contact Hypothesis: Cross-Linguistic Evaluation of Religion and Nationality Bias When Prompting LLMs in German and Icelandic

Catherine Ikae¹, Guðbjörg Linda Rafnsdóttir²,
Ragna Kemp Haraldsdóttir², Mascha Kurpicz-Briki¹,

¹Applied Machine Intelligence, Bern University of Applied Sciences, Biel, Switzerland,

²Faculty of Sociology, Anthropology and Folkloristics, University of Iceland, Reykjavik, Iceland

Correspondence: catherine.ikae@bfh.ch

Abstract

Large Language Models (LLMs) can reproduce social biases, yet many bias evaluations remain English-centric. We extend the Contact Hypothesis framework presented in previous work to German and Icelandic, focusing on religion and nationality. Evaluating GPT models (3.5, 4, 4-turbo, 4o, 5), we find that positive contact reduces biases in the answers of the LLMs, while negative contact amplifies it, with cross-linguistic differences in magnitude and salience. Our results support the cross-linguistic robustness of contact-based probing and underscore the need for culturally contextualized evaluations. In addition to these insights, our contributions lies in the dataset that is made available on Github¹ for further research.

1 Introduction

LLMs encode and can amplify societal stereotypes (Bolukbasi et al., 2016; Caliskan et al., 2017; Bender et al., 2021), raising concerns for fair deployment. The *Contact Hypothesis* (Allport, 1954) posits that positive intergroup contact reduces prejudice and has recently been operationalized for English LLMs, yielding predictable shifts under positive versus negative prompts and introducing Social Contact Debiasing (Raj et al., 2024). However, its applicability beyond English remains underexplored.

We extend contact-based bias probing to German and Icelandic, examining religion and nationality, two socially salient dimensions shaped by regional migration and religious demographics (Smith et al., 2022; Parrish et al., 2022). Our evaluation spans multiple GPT generations (gpt-3.5, gpt-4, gpt-4o, gpt-5) (OpenAI et al., 2023; OpenAI, 2023, 2024, 2025).

Prior work shows that LLM bias varies across languages and cultural contexts (Kim and Baek,

2024; Zahraei and Asgari, 2025; Buyl et al., 2024), and multilingual evaluation frameworks stress the importance of cultural grounding beyond direct translation (University of Amsterdam ILLC, 2024; Huang et al., 2025; Pistilli et al., 2024). Our work contributes an interpretable cross-linguistic evaluation grounded in intergroup contact theory and releases a hand-curated dataset to support reproducible benchmarking.

Icelandic and German, though both Germanic, differ substantially: Icelandic (North Germanic) has remained linguistically conservative, whereas German (West Germanic) reflects broader contact and borrowing making their comparison particularly informative.

Contributions. (1) A multilingual extension of contact-based bias probing to German and Icelandic; (2) culturally grounded descriptors for religion and nationality; (3) cross-model evidence of consistent contact effects alongside language-specific bias patterns.

2 Related Work

Bias in NLP. Social biases are well documented in word embeddings and contextual models (Bolukbasi et al., 2016; Caliskan et al., 2017; Guo and Caliskan, 2021; Bender et al., 2021). Benchmarks such as StereoSet, CrowS-Pairs, HolisticBias, and BBQ enable systematic evaluation (Nadeem et al., 2021; Nangia et al., 2020; Smith et al., 2022; Parrish et al., 2022; Zhao et al., 2023). Multilingual studies show that bias varies across languages and cultural contexts, reflecting training data and alignment objectives (Gamboa et al., 2025; Buyl et al., 2024). Frameworks such as MCEVAL and CIVICS advocate culturally grounded evaluation beyond direct translation (Huang et al., 2025; Pistilli et al., 2024), and recent work highlights regional differences in national and religious bias patterns (Zahraei and Asgari, 2025; Liu et al., 2025).

¹<https://github.com/BFH-AMI/BIAS-SmarterPromptingDemonstrator>

Mitigation. Debiasing approaches target data, representations, and objectives (Zhang et al., 2018; Zhao et al., 2018). The Contact Hypothesis offers a complementary, theory-driven perspective (Allport, 1954; Wright et al., 1997; Amichai-Hamburger and McKenna, 2006) and has recently been operationalized for LLMs with promising results (Raj et al., 2024). We extend this sociopsychological framing to multilingual bias evaluation, responding to calls for culturally sensitive fairness assessment (University of Amsterdam ILLC, 2024; Arif et al., 2024; Kim and Baek, 2024).

3 Methodology

3.1 Overview

We adapt the English prompt framework of Raj et al. (2024) to German and Icelandic, focusing on religion and nationality descriptors aligned with local demographics (see §4).

3.2 Prompt Design

Following Raj et al. (2024), prompts vary along three dimensions: *contact condition* (none, positive, negative), *action orientation* (positive verbs), and response *certainty*. Prompts were translated and culturally validated by native speakers while preserving parallel structure across languages to isolate language effects under controlled framing (Smith et al., 2022).

3.3 Models Evaluated

We evaluate five GPT variants (GPT-3.5, GPT-4, GPT-4-turbo, GPT-4o, GPT-5) (OpenAI, 2023; OpenAI et al., 2023; OpenAI, 2024, 2025) using identical prompts. These models were selected due to their strong performance, widespread real-world deployment, and influence on downstream NLP applications, making them a relevant target for bias analysis (Bommasani et al., 2021). Furthermore, focusing on a single model family enables controlled comparisons between versions with similar underlying design principles and alignment strategies, which have been shown to significantly affect model behavior and bias between different model families (Liang et al., 2023; Nadeem et al., 2021). However, this choice limits the scope of the analysis, as other model families, particularly open-source and smaller-scale models, may exhibit different bias characteristics. The decoding was fixed (temperature 0.3, maximum 10 tokens) and the responses were constrained to binary outputs

(yes/no) to ensure consistent scoring (Parrish et al., 2022). For example, German prompts appended: “*Antworte nur mit einem Wort: Wähle zwischen Ja und Nein.*”

3.4 Bias Definition and Scoring

Following Raj et al. (2024), for *positive verb (inclusive) actions* such as “Should I collaborate?”, an affirmative response (“Yes”) is labeled *unbiased*, whereas a negative response (“No”) is labeled *biased*, irrespective of contact condition. Ambiguous outputs (none binary) were reviewed by native speakers who are members of the research team (one native German speaker and two native Icelandic speakers); responses without clear orientation were labeled *None*. We report the percentage of biased responses per model, language, and condition.

4 Experiments & Results

4.1 Relation to Prior German Study and Methodological Reuse

This study extends our prior work on German contact-based bias probing (Ikae and Kurpicz-Briki, 2025), which introduced a culturally grounded dataset construction pipeline based on (1) demographically relevant group selection, (2) translation and adaptation of English templates, and (3) native-speaker validation. Bias was evaluated under No, Positive, and Negative contact framings following Raj et al. (2024).

Here, we retain the original German dataset for comparability and apply the same construction principles to Icelandic, ensuring parallel prompt structure, contact conditions, and scenario coverage (education, workplace, community, healthcare, sports). While group descriptors were adapted to Icelandic linguistic and cultural context, methodological parity enables controlled cross-linguistic evaluation.

We further extend both languages to religion, expanding the earlier German dataset beyond nationality. This reuse and extension of our established pipeline enables systematic multilingual comparison of contact-based bias across nationality and religion.

4.2 Experimental Setup

Prompt counts (per language). Due to differences in the number and availability of culturally relevant nationalities and religious groups included in the prompt construction for each language, the

Model	Contact	Unbiased	Biased	None
GPT-3.5	No	99.7	0.3	0.0
	Positive	99.7	0.3	0.0
	Negative	95.1	4.9	0.0
GPT-4	No	96.3	3.6	0.2
	Positive	100.0	0.0	0.0
	Negative	89.7	9.6	1.2
GPT-4o	No	98.5	1.5	0.3
	Positive	99.8	0.0	0.2
	Negative	95.3	4.3	0.7
GPT-4 Turbo	No	98.7	1.3	0.0
	Positive	99.8	0.2	0.0
	Negative	95.4	4.6	0.0
GPT-5	No	80.8	0.0	19.2
	Positive	84.9	0.0	15.1
	Negative	79.8	0.0	20.2

Table 1: Response distribution (%) by contact type for each GPT model in the **German Nationality** condition. Positive contact consistently reduces bias; GPT-5 shows higher neutral (“None”) rates, indicating bias avoidance rather than elimination.

total number of generated prompts varies. For **German**, the Nationality condition comprises 609 base prompts (1,827 with three contact framings), and Religion includes 210 base prompts (630 total). For **Icelandic**, 90 base templates yield 300 Nationality prompts (900 total) and 180 Religion prompts (540 total), each instantiated under No, Positive, and Negative contact.

Descriptor sets. The **German Nationality** set includes 19 migrant groups identified from official statistics in German-speaking countries (including Afghanistan, Bosnia, Bulgaria, France, Greece, India, Italy) (Ikae and Kurpicz-Briki, 2025). The **Icelandic Nationality** condition comprises 10 groups (Poland, Lithuania, Ukraine, Romania, Portugal, Spain, Venezuela, Philippines, USA, Denmark).

For **Religion**, German includes seven groups (Christians, Muslims, Jews, Buddhists, Russian Orthodox, Jehovah’s Witnesses, non-religious), while Icelandic includes five (Christianity, Islam, Buddhism, Jehovah’s Witnesses, Russian Orthodox). Each religion appears in 30 templates per language.²

Scenarios. Descriptors are embedded in parallel decision-focused scenarios across five domains: education, workplace, healthcare, sports, and community contexts. Each scenario is realized under all three contact conditions, enabling controlled cross-language comparison (Raj et al., 2024).

Model	Contact	Unbiased	Biased	None
GPT-3.5	No	95.2	4.8	0.0
	Positive	99.0	1.0	0.0
	Negative	93.8	6.2	0.0
GPT-4	No	94.3	5.7	0.0
	Positive	95.7	3.3	1.4
	Negative	92.8	5.3	2.9
GPT-4o	No	94.8	5.2	0.0
	Positive	96.7	3.3	0.0
	Negative	96.2	3.8	0.0
GPT-4 Turbo	No	93.3	6.7	0.0
	Positive	93.3	6.7	0.0
	Negative	94.8	5.2	0.0
GPT-5	No	68.9	10.0	21.1
	Positive	67.4	7.1	25.5
	Negative	72.3	2.3	23.9

Table 2: Response distribution (%) by contact type for each GPT model in the **German Religion** condition. Positive contact consistently reduces bias, while GPT-5 exhibits a higher rate of neutral (“None”) responses.

Model	Contact	Unbiased	Biased	None
GPT-3.5	No	99.0	1.0	0.0
	Positive	100.0	0.0	0.0
	Negative	91.7	8.3	0.0
GPT-4	No	54.0	0.3	45.7
	Positive	94.0	0.0	6.0
	Negative	65.0	12.3	22.7
GPT-4o	No	99.3	0.7	0.0
	Positive	100.0	0.0	0.0
	Negative	93.0	7.0	0.0
GPT-4 Turbo	No	99.0	1.0	0.0
	Positive	96.3	3.0	0.7
	Negative	82.0	16.7	1.3
GPT-5	No	76.0	3.7	20.3
	Positive	82.7	2.7	14.7
	Negative	67.0	17.0	16.0

Table 3: Response distribution (%) by contact type for each GPT model in the **Icelandic Nationality** condition. Positive contact consistently reduces bias, while GPT-4 and GPT-5 exhibit a higher rate of neutral (“None”) responses.

4.3 Overall Bias Levels

Tables 1–4 report unbiased, biased, and neutral responses across models, languages, and descriptor conditions. GPT-3.5, GPT-4 (for German), GPT-4o, and GPT-4 Turbo consistently achieved high unbiased rates (often > 90%) across settings. In contrast, GPT-5 produced lower unbiased proportions and markedly more neutral responses, particularly in Religion conditions. This was also the case for GPT-4 in Icelandic. Positive contact yielded the highest unbiased rates across all models, while Negative contact increased bias. German results were overall more stable, whereas Icelandic prompts es-

²Descriptors are presented in English for readability; original wording is available in the repository.

Model	Contact	Unbiased	Biased	None
GPT-3.5	No	85.6	14.4	0.0
	Positive	92.2	7.8	0.0
	Negative	76.1	23.9	0.0
GPT-4	No	65.1	10.5	24.4
	Positive	78.3	4.3	17.4
	Negative	65.8	17.4	11.6
GPT-4o	No	81.2	16.6	1.7
	Positive	94.4	5.6	0.0
	Negative	85.0	14.4	0.6
GPT-4 Turbo	No	86.7	13.3	0.0
	Positive	91.7	8.3	0.0
	Negative	80.6	19.4	0.0
GPT-5	No	53.0	8.1	38.9
	Positive	65.7	10.1	25.3
	Negative	68.3	7.8	23.9

Table 4: Response distribution (%) by contact type for each GPT model in the **Icelandic Religion** condition. Positive contact consistently reduces bias, while GPT-4 and GPT-5 exhibit a higher rate of neutral (“None”) responses.

pecially for GPT-5 elicited higher neutrality.

4.4 Nationality Comparison: German vs. Icelandic Conditions

Overall Patterns. German nationality prompts resulted in highly stable behavior: all models except GPT-5 showed minimal bias and virtually no neutral responses across contact types. GPT-5 diverged, with reduced unbiased rates and elevated neutrality, suggesting greater safety-driven non-commitment.

Icelandic nationality prompts produced substantially greater variability. While earlier models remained robust under Positive contact, GPT-4 and GPT-5 showed increased biased and neutral responses, particularly under No and Negative contact, indicating heightened caution or representational uncertainty.

Contact Effects. Positive contact reduced bias in both languages, reaching near-ceiling performance in the German condition. In Icelandic, however, Positive contact did not fully offset elevated neutrality in GPT-4 and GPT-5. Negative contact increased bias in both settings, but the effect was stronger for Icelandic nationality prompts.

Summary. Overall, German nationality elicited more stable and context-resistant responses, whereas Icelandic nationality triggered greater variability and neutrality especially in later models suggesting increased caution in linguistically or culturally distinct contexts.

4.5 Religion Comparison: German vs. Icelandic Conditions

Across religious conditions, contact effects were consistent but differed in magnitude between German and Icelandic prompts.

Overall Patterns. In the **German Religion** condition, GPT-3.5, GPT-4, GPT-4o, and GPT-4 Turbo maintained high unbiased rates across contact types, with only modest increases in bias under Negative contact. GPT-5 diverged, producing substantially lower unbiased rates and markedly higher neutral responses, indicating increased non-commitment in religious contexts.

The **Icelandic Religion** condition showed greater variability. While earlier models remained relatively robust, GPT-4 and GPT-5 generated elevated neutral responses across contact types, suggesting stronger uncertainty or safety-driven avoidance compared to German prompts.

Contact Effects. Positive contact reduced bias in both languages but did not fully offset elevated neutrality in GPT-4 and GPT-5 for Icelandic prompts. Negative contact increased bias in both settings, with stronger effects in the Icelandic condition and continued high neutrality in later models.

Summary. Overall, German religion prompts elicited more stable responses, whereas Icelandic religion prompts produced greater variability and substantially higher neutrality especially in GPT-4 and GPT-5 indicating increased caution in linguistically and culturally distinct contexts.

4.6 Key Findings

- **Positive contact reliably attenuates bias across all models and nationalities.** In both German and Icelandic conditions, benevolent framing consistently increased the proportion of unbiased responses and reduced explicit bias, although its effectiveness was stronger for German nationality cues. For Icelandic prompts, particularly for GPT-4 and GPT-5, positive contact did not fully eliminate elevated rates of neutral (“None”) outputs, indicating residual uncertainty or safety-driven avoidance (Allport, 1954; Raj et al., 2024).
- **Nationality influences model sensitivity to contextual framing.** German nationality elicited highly stable behaviour, with low bias and minimal neutrality across models. In

contrast, Icelandic nationality cues produced greater variability, higher susceptibility to negative contact, and substantially increased neutral responding most notably among GPT-4 and GPT-5. This cross-national difference aligns with work on contextual salience and social-descriptor effects in multilingual LLM evaluation (Smith et al., 2022).

- **Model generation strongly predicts bias expression and risk avoidant behaviour.** Earlier models (GPT-3.5, GPT-4o, GPT-4 Turbo) showed consistently low bias and near-zero neutrality across conditions. In contrast, GPT-5 demonstrated a pronounced shift toward caution, characterised by persistently high neutral responding (15–40% depending on condition) and reduced decisiveness even under positive contact. GPT-4 exhibited similar behaviour under Icelandic nationality cues, suggesting that newer models prioritise safety-driven avoidance over explicit commitment in sensitive contexts.

4.7 Bias Patterns Across Religious Groups in the German Condition

Table 5 summarizes the distribution of biased responses across all five GPT models for the German Religion condition. The results demonstrate that bias is unevenly distributed across religious groups. Aggregated across contact types, the highest explicit bias rates are observed for *Jehovah’s Witnesses* (9.2%), followed by *Christianity* (8.1%), *Russian Orthodox* (5.5%), and *Buddhism* (5.0%). In contrast, prompts referencing *Islam* exhibit the lowest bias rate (2.4%), with *Judaism* (4.1%) and *no religious affiliation* (3.0%) occupying intermediate positions. These findings indicate that minority or denominational Christian groups are particularly salient triggers of biased responses in the German setting.

When disaggregated by contact type, biased responding is most pronounced under **Negative contact**, with the strongest effects observed for *Jehovah’s Witnesses* and *Christianity*. Notably, elevated bias also persists under **No contact** conditions for several groups, suggesting that neutral framing alone does not reliably suppress stereotypical associations. While **Positive contact** generally reduces biased responses across religious categories, it does not fully eliminate bias for groups that exhibit higher baseline susceptibility.

Religion	Biased (%)	Neutral (%)	Unbiased (%)
Jehovah’s Witnesses	9.2	0.0	90.8
Christianity	8.1	0.0	91.9
Russian Orthodox	5.5	0.0	94.5
Buddhism	5.0	0.0	95.0
Judaism	4.1	0.0	95.9
No Religion	3.0	0.0	97.0
Islam	2.4	0.0	97.6

Table 5: Aggregated biased response rates across all five GPT models and all contact types for the **German Religion** condition. Jehovah’s Witnesses and Christians show the highest bias rates, whereas Muslim prompts elicit the lowest.

Model-level variation further contextualizes these patterns. GPT-4 Turbo and GPT-5 generate comparatively higher proportions of biased responses than earlier models, reflecting increased sensitivity to contextual framing. At the same time, GPT-5 displays a distinctive behavioural profile characterized by a greater tendency toward conservative response strategies. However, in contrast to the Icelandic conditions, neutral responses remain negligible in the German Religion setting, indicating that models predominantly produce direct evaluative judgments rather than resorting to avoidance.

Overall, these results suggest that religious bias in German-language prompts is shaped primarily by group-specific sociocultural positioning rather than by general religiosity alone. In some cases, minority, non-mainstream, and historically marginalized denominations elicit higher levels of bias, and their interaction with contact framing highlights the central role of contextual cues. Negative framing amplifies biased responding, whereas positive framing provides partial, but incomplete, mitigation, underscoring the limits of prompt-level interventions in fully neutralizing model bias.

4.8 Bias Patterns Across Nationality Groups in the German Condition

Table 6 presents the aggregated bias rates across nationality groups in the German Nationality condition. Overall, explicit bias remains relatively low, with unbiased responses exceeding 89% for all groups. Nevertheless, systematic and interpretable differences emerge across nationalities. The highest biased response rates are observed for *Syria* (9.6%), *Afghanistan* (8.7%), and *Iraq* (8.3%), followed by *Bosnia and Herzegovina* (7.8%) and *Romania* (6.9%). These nationalities consistently elicit elevated biased responses across contact fram-

ings and models, in contrast to most Western, Southern, and North American groups, which remain below 3% biased output.

Although explicit bias is infrequent overall, its distribution is clearly non-uniform. Nationalities associated with recent refugee movements, geopolitical instability, or heightened public and media discourse in Germany tend to attract higher bias rates, suggesting that sociopolitical salience plays a substantial role in shaping model behaviour. In contrast, nationalities such as *France*, *Italy*, *Portugal*, and the *United States* exhibit consistently low bias levels despite comparable institutional and cultural visibility. This pattern indicates that bias expression reflects perceived social distance and dominant media framing rather than simple frequency of exposure.

Neutral responses are rare across nearly all nationality groups, remaining below 2% in every case. This distinguishes the German Nationality condition from both religion-related and Icelandic nationality evaluations, where avoidance strategies are more prevalent. In the German setting, models generally provide direct affirmative or negative judgments even when addressing socially sensitive nationalities.

Taken together, these findings demonstrate that while overall bias levels in German nationality prompts are limited, they are concentrated among specific, socially salient groups. Aggregated model-level metrics therefore obscure meaningful cross-national variation, underscoring the importance of fine-grained, group-specific evaluation in multilingual bias assessment.

4.9 Bias Patterns Across Nationality Groups in the Icelandic Condition

Table 7 summarizes the aggregated bias rates across all GPT models for the Icelandic Nationality condition. Although explicit bias remains modest overall, substantial variation is observed across nationalities. The highest bias rates occur for *USA* (7.1%), *Venezuela* (6.2%), and *Denmark* (6.2%), followed by *Romania* and *Spain* (4.9%). These groups elicit disproportionately more biased responses than the remaining nationalities.

In contrast, several nationalities show comparatively low bias levels, including *Ukraine* (3.8%), *Lithuania* (4.0%), *Portugal* (4.2%), and *Poland* (4.2%). These groups consistently receive high rates of unbiased responses (87–88%), suggesting strong cooperative judgments across contact fram-

Nationality	Biased (%)	Neutral (%)	Unbiased (%)
Syria	9.6	1.2	89.2
Afghanistan	8.7	1.0	90.3
Iraq	8.3	1.4	90.3
Bosnia and H.	7.8	0.9	91.3
Romania	6.9	0.7	92.4
Serbia	6.4	0.8	92.8
Russia	5.9	1.6	92.5
Bulgaria	5.1	0.6	94.3
Hungary	4.8	0.5	94.7
Poland	4.3	0.4	95.3
Croatia	3.9	0.4	95.7
Greece	3.6	0.3	96.1
Kosovo	3.4	0.5	96.1
Portugal	3.2	0.3	96.5
Italy	2.9	0.3	96.8
Turkey	2.6	0.3	97.1
France	2.5	0.2	97.3
India	2.2	0.4	97.4
Ukraine	1.8	0.2	98.0
United States	1.2	0.1	98.7

Table 6: Nationalities with the highest bias rates in the **German Nationality** condition, aggregated across all five GPT models and contact types. Bias levels are low overall but show consistent elevation for several nationalities, particularly Syria, Afghanistan, and Iraq.

ings and models.

Neutral responses display noticeable differences as well. *Venezuela* and *USA* exhibit elevated neutrality (11.6% and 9.1%, respectively), indicating a tendency for model abstention or uncertainty. For most other nationalities, neutrality remains between 7–8%, reflecting a comparatively low rate of safety-driven non-commitment.

Overall, the Icelandic Nationality analysis reveals that explicit bias is not uniformly distributed across nationality categories. Instead, particular nationalities notably the United States, Denmark, and Venezuela exhibit systematically higher levels of biased and neutral outputs, while others remain largely unbiased. These patterns align with the hypothesis that sociocultural salience and perceived group distance influence LLM bias expression even in a low-bias environment.

4.10 Bias Patterns Across Religious Groups in the Icelandic Condition

Table 8 reports the aggregated proportions of biased, neutral, and unbiased responses across all GPT models for the Icelandic Religion condition. The results show clear differences across religious groups despite overall moderate levels of explicit bias.

The highest bias rates occur for *Russian Orthodox* (22.9%), followed by *Jehovah’s Witnesses* (16.0%) and *Christian* (13.1%). These findings indicate that non-mainstream or denominational Christian groups such as the Russian Or-

Nationality	Biased (%)	Neutral (%)	Unbiased (%)
USA	7.11	9.11	83.78
Venezuela	6.22	11.56	82.22
Denmark	6.22	7.11	86.67
Romania	4.89	8.00	87.11
Spain	4.89	8.22	86.89
The Philippines	4.67	8.22	87.11
Portugal	4.22	8.00	87.78
Poland	4.22	8.44	87.33
Lithuania	4.00	8.00	88.00
Ukraine	3.78	8.67	87.56

Table 7: Aggregated biased, neutral, and unbiased response rates across all five GPT models and contact types for the **Icelandic Nationality** condition. Bias levels vary across nationalities, with USA, Denmark, and Venezuela showing notably higher bias rates and neutral responses than other groups.

thodox Church and Jehovah’s Witnesses provoke the strongest biased reactions across models. Elevated neutrality for *Jehovah’s Witnesses* (15.1%) and *Russian Orthodox* (10.4%) further suggests that these categories are perceived as socially sensitive, prompting both explicit bias and increased avoidance behaviour.

In contrast, *Buddhist* and *Islam* show the lowest explicit bias rates (6.9% and 6.4%, respectively) and the highest proportions of unbiased responses (84.2%). These patterns parallel the German Religion results, where Islam also did not yield the highest bias levels indicating that model sensitivity to religious groups may be strongly conditioned by local cultural context rather than by presumed global stereotypes. Another hypothesis could be that more alignment efforts have been already done to the models for specific groups, and therefore there is a reduced bias.

Neutral responses remain relatively modest overall but increase for groups associated with higher bias levels, highlighting a dual mechanism of stereotyping and uncertainty-driven non-commitment. Together, the Icelandic Religion findings demonstrate that bias expression is unevenly distributed across religious categories, with some denominations repeatedly eliciting stronger biased and neutral responses even in a relatively low-bias setting.

5 Discussion

Cross-Condition Comparison. Across all four experimental conditions, our results reveal a consistent yet nuanced pattern in how GPT models express and regulate social bias. Explicit bias is lowest and most stable in the **German Nationality** con-

Religion	Biased (%)	Neutral (%)	Unbiased (%)
Russian Orthodox	22.9	10.4	66.7
Jehovah’s Witness	16.0	15.1	68.9
Christian	13.1	9.8	77.1
Buddhist	6.9	8.9	84.2
No Religion	8.1	0.0	91.9
Islam	6.4	9.3	84.2

Table 8: Aggregated biased, neutral, and unbiased response rates across all five GPT models and contact types for the **Icelandic Religion** condition. Bias levels vary across religions, with Russian Orthodox and Jehovah’s Witness, showing notably higher bias rates than other religious groups.

dition, where nearly all models exhibit near-ceiling unbiased responses and minimal neutrality. In contrast, the **Icelandic Nationality** condition shows greater variability across groups, with certain nationalities (e.g., USA, Denmark, Venezuela) eliciting elevated biased and neutral responses. These findings indicate that nationality-related sensitivity is not uniform across languages and may depend on local sociocultural salience. A similar asymmetry appears across the two religion conditions. In the **German Religion** setting, bias concentrates most strongly on Jehovah’s Witnesses and Christian subgroups, whereas the **Icelandic Religion** condition exhibits heightened bias toward Russian Orthodox and Jehovah’s Witness, alongside moderate bias toward Christians. Conversely, Islam and Buddhism show comparatively low bias in both languages, suggesting that model behaviour is shaped more by culturally specific religious representations than by global religious stereotypes. Another explanation could be existing debiasing efforts and alignment for specific groups. Notably, neutrality increases systematically in the conditions and groups that elicit higher bias most prominently in Icelandic Religion and Icelandic Nationality consistent with a safety-driven avoidance strategy observed most strongly in GPT-5. Taken together, these results demonstrate that bias expression in current LLMs is sensitive to both linguistic context and group-specific framing, with nationality and religion showing distinct cross-linguistic patterns that underscore the importance of multilingual evaluation when assessing fairness and model alignment.

Cross-Linguistic Generalizability. The present findings demonstrate that contact-based probing extends robustly beyond English, supporting the cross-cultural validity of contact theory in computational settings (Allport, 1954; Wright et al., 1997; Amichai-Hamburger and McKenna, 2006).

Across both German and Icelandic conditions, positive contact consistently reduced explicit bias, while negative contact amplified it, mirroring prior English-only results (Raj et al., 2024). This convergence indicates that the underlying socio-cognitive mechanisms captured by contact framing might not be language-specific but might generalize across typologically related European languages. Following previous work in the field (Raj et al., 2024), the concept of cross-cultural validity is approximated by the languages in our work.

Language-Specific Patterns. Despite this generalizability, the models displayed clear nationality-specific patterns. German nationality cues elicited highly stable behaviour across model generations, with consistently low bias and minimal neutrality even under negative contact. In contrast, Icelandic nationality elicited substantially more variability, stronger sensitivity to negative framing, and elevated neutral responses especially in later-generation models such as GPT-4 and GPT-5. These cross-linguistic differences underscore the importance of culturally contextualized descriptors and social salience in LLM bias expression (Smith et al., 2022; Parrish et al., 2022), and caution against relying solely on English-centric evaluations when assessing model fairness (Bender et al., 2021).

Model Evolution. The progression from GPT-3.5 to GPT-4o reflects a clear downward trend in overt bias, consistent with claims about advances in dataset curation, alignment, and reinforcement learning from human feedback (OpenAI et al., 2023). However, GPT-5 introduced a distinct behavioural profile characterized by high rates of neutral responses, particularly in sensitive contexts (e.g., Icelandic nationality and Icelandic religion). This shift suggests that newer models increasingly rely on safety-driven non-commitment when uncertainty or social risk is detected, rather than resolving bias through more robust reasoning. Residual bias and avoidance behaviours therefore persist in nuanced forms (Zhao et al., 2023), even in the most recent model generations.

Mitigation Implications. The consistency of contact effects across languages indicates that contact-based prompt strategies (Raj et al., 2024) may serve as a practical, model-agnostic mitigation mechanism, complementing algorithmic and representational debiasing approaches (Zhang et al.,

2018; Zhao et al., 2018). However, the elevated neutrality in GPT-5 suggests that mitigation efforts should also address avoidance-based failure modes, which differ qualitatively from explicit stereotyping and may obscure underlying decision-making limitations.

Limitations and Future Work. This study is limited to two languages, two dimensions of social identity, and GPT-family models. Additionally, the response space is restricted to categorical judgments, which may mask subtler forms of bias or reasoning. Also the manual review process by native speakers with human judgment might have impacted the classification of unambiguous answers. Future work should expand to more languages and cultural contexts, incorporate open-source and smaller models, and explore fine-tuning or prompting interventions that leverage contact framing more systematically (Raj et al., 2024). Longitudinal analyses across model updates may further clarify whether neutrality reflects genuine caution or unresolved representational gaps.

6 Conclusion

This work extends contact-hypothesis probing beyond earlier German-only studies by introducing a parallel Icelandic dataset for both nationality and religion, enabling a controlled cross-linguistic evaluation of bias in contemporary GPT models. Across all conditions, the models exhibit clear contact-aligned shifts positive contact reliably increases acceptance, whereas negative contact amplifies biased responses confirming the psychological predictions of contact theory (Allport, 1954; Wright et al., 1997; Raj et al., 2024). At the same time, the magnitude and distribution of bias differ systematically between languages and social categories. Nationality prompts in German yield near-ceiling unbiased responses, whereas Icelandic nationality prompts show higher variability and a greater tendency toward safety-driven neutrality. In the religion conditions, bias concentrates on some groups (e.g., Russian Orthodox, Jehovah’s Witness), while Islam and Buddhism receive comparatively low bias across both languages, highlighting strong cultural and contextual effects (Smith et al., 2022; Parrish et al., 2022).

Together, these results demonstrate that bias in LLMs cannot be meaningfully assessed in a single language or cultural setting. Instead, multilingual and context-sensitive evaluation is essential

to understanding the sociolinguistic contours of model behaviour. The consistent alignment with contact framing across languages further suggests that theory-grounded approaches from social psychology can inform more reliable and generalizable bias-mitigation strategies.

Future work should extend this framework to additional languages, model families, and social categories, and explore contact-aware training or prompting as a structured path toward safer and culturally adaptive language models.

7 Acknowledgments

This work is part of the Europe Horizon project BIAS, grant agreement number 101070468, funded by the European Commission, and has received funding from the Swiss State Secretariat for Education, Research and Innovation (SERI).

References

- Gordon W Allport. 1954. The nature of prejudice. *Addison-Wesley*.
- Yair Amichai-Hamburger and Katelyn Y A McKenna. 2006. The contact hypothesis reconsidered: Interacting via the internet. *Journal of Computer-Mediated Communication*, 11(3):825–843.
- Samee Arif, Zohaib Khan, Maaidah Kaleem, Suhaib Rashid, Agha Ali Raza, and Awais Athar. 2024. [With a grain of salt: Are llms fair across social dimensions?](#) *arXiv preprint arXiv:2410.12499*.
- Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. [On the dangers of stochastic parrots: Can language models be too big?](#) In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.
- Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. 2016. [Man is to computer programmer as woman is to homemaker? debiasing word embeddings](#).
- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, S. Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen A. Creel, Jared Davis, Dora Demszky, Chris Donahue, Moussa Koulako Bala Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas F. Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, O. Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kudipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avnika Narayan, Deepak Narayanan, Benjamin Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel J. Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Robert Reich, Hongyu Ren, Frieda Rong, Yusuf H. Roohani, Camilo Ruiz, Jack Ryan, Christopher R’e, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishna Parasuram Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei A. Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2021. [On the opportunities and risks of foundation models](#). *ArXiv*, abs/2108.07258.
- Maarten Buyl, Alexander Rogiers, Sander Noels, Guillaume Bied, Iris Dominguez-Catena, Edith Heiter, Iman Johary, Alexandru-Cristian Mara, Raphaël Romero, Jefrey Lijffijt, and Tijl De Bie. 2024. [Large language models reflect the ideology of their creators](#). *arXiv preprint arXiv:2410.18417*.
- Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. [Semantics derived automatically from language corpora contain human-like biases](#). *Science*, 356(6334):183–186.
- Lance Calvin Lim Gamboa, Yue Feng, and Mark Lee. 2025. [Social bias in multilingual language models: A survey](#). *arXiv preprint arXiv:2508.20201*.
- Wei Guo and Aylin Caliskan. 2021. Detecting emergent intersectional biases: Contextualized word embeddings contain a distribution of human-like biases. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, pages 122–133.
- Shulin Huang, Linyi Yang, and Yue Zhang. 2025. [Mceval: A dynamic framework for fair multilingual cultural evaluation of llms](#). *arXiv preprint arXiv:2507.09701*.
- Catherine Ikae and Mascha Kurpicz-Briki. 2025. Measuring bias in german prompts to gpt models using contact hypothesis. In *AIMMES Workshop, AI Fairness Cluster Conference*.
- Minsung Kim and Sanghoon Baek. 2024. [Exploring large language models on cross-cultural values in connection with training methodology](#). *arXiv preprint arXiv:2412.08846*.
- Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian

- Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2023. [Holistic evaluation of language models](#). *Transactions on Machine Learning Research*. Featured Certification, Expert Certification.
- Qianying Liu, Katrina Qiyao Wang, Fei Cheng, and Sadao Kurohashi. 2025. [Assessing large language models in agentic multilingual national bias](#). *arXiv preprint arXiv:2502.17945*.
- Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. Stereoset: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5356–5371.
- Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. [CrowS-pairs: A challenge dataset for measuring social biases in masked language models](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online. Association for Computational Linguistics.
- OpenAI. 2023. Gpt-3.5 (chatgpt model). <https://platform.openai.com/docs/models/gpt-3-5>. Accessed 2025.
- OpenAI. 2024. Gpt-4o. <https://openai.com/index/hello-gpt-4o>. Accessed 2025.
- OpenAI. 2025. Introducing gpt-5. <https://openai.com/index/introducing-gpt-5/>. Accessed 2025.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, et al. 2023. [Gpt-4 technical report](#). Technical report, OpenAI. ArXiv:2303.08774.
- Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel Bowman. 2022. [BBQ: A hand-built bias benchmark for question answering](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2086–2105, Dublin, Ireland. Association for Computational Linguistics.
- Giada Pistilli, Alina Leiding, Yacine Jernite, Atoosa Kasirzadeh, Alexandra Sasha Luccioni, and Margaret Mitchell. 2024. [Civics: Building a dataset for examining culturally-informed values in large language models](#). In *Proceedings of the 2024 AAAI/ACM Conference on AI, Ethics, and Society*.
- Chahat Raj, Anjishnu Mukherjee, Aylin Caliskan, Antonios Anastasopoulos, and Ziwei Zhu. 2024. Breaking bias, building bridges: Evaluation and mitigation of social biases in llms via contact hypothesis. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*.
- Eric Michael Smith, Mitchell Hall, Melanie Kambadur, Edoardo Presani, and Adina Williams. 2022. “I’m sorry to hear that”: Finding new biases in language models with a holistic descriptor dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9180–9211.
- University of Amsterdam ILLC. 2024. Multilinguality and multiculturalism: Towards more effective and inclusive neural language models. <https://eprints.illc.uva.nl/id/eprint/2347/>.
- Stephen Wright, Arthur Aron, Tracy McLaughlin-Volpe, and Stacy Ropp. 1997. The extended contact effect: Knowledge of cross-group friendships and prejudice. *Journal of Personality and Social Psychology*, 73(1):73–90.
- Pardis Sadat Zahraei and Ehsaneddin Asgari. 2025. [I am aligned, but with whom? mena values benchmark for evaluating cultural alignment and multilingual bias in llms](#).
- Brian Hu Zhang, Benjamin Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, pages 335–340.
- Jieyu Zhao, Meng Fang, Shirui Pan, Wenpeng Yin, and Mykola Pechenizkiy. 2023. Gptbias: A comprehensive framework for evaluating bias in large language models. *arXiv preprint arXiv:2312.06315*.
- Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 15–20.

A Scenario Comparison

Tables 9 and 10 show the bias distribution among the different scenarios for the experiments with the German language.

Tables 11 and 12 show the bias distribution among the different scenarios for the experiments with the Icelandic language.

Across languages and identity dimensions, scenario effects followed a consistent structural pattern but differed substantially in magnitude. Workplace

Scenario	Biased (%)	Neutral (%)	Unbiased (%)
Workplace	14.9	10.2	74.9
Healthcare	4.0	4.3	91.7
Sports	3.8	9.2	87.0
Education	1.9	5.6	92.5
Community	0.2	2.7	97.1

Table 9: Bias distribution across German religion scenarios aggregated over models and contact conditions (balanced scenario counts).

Scenario	Biased (%)	Neutral (%)	Unbiased (%)
Education	3.5	2.4	94.1
Community	3.3	2.8	93.9
Sports	3.2	2.8	94.0
Healthcare	3.0	2.5	94.5
Workplace	2.6	3.1	94.4

Table 10: Bias distribution across German nationality scenarios aggregated over models and contact conditions (balanced scenario counts).

contexts were the most bias-prone across all conditions, whereas Community scenarios were consistently the most robust. Religion elicited higher bias levels than nationality in both languages, with Icelandic religion showing the strongest effects overall (26.1%), followed by German religion (14.9%). Nationality prompts produced comparatively lower bias, particularly in German ($\leq 3.5\%$ across all scenarios). While the ranking of scenarios remained stable cross-linguistically, bias intensity was clearly modulated by both language and identity type.

Scenario	Biased (%)	Neutral (%)	Unbiased (%)
Workplace	26.1	13.9	60.0
Healthcare	11.5	7.8	80.7
Sports	10.7	12.0	77.2
Education	7.8	7.0	85.2
Community	4.8	7.2	88.0

Table 11: Bias distribution across Icelandic religion scenarios aggregated over models and contact conditions (balanced scenario counts).

Scenario	Biased (%)	Neutral (%)	Unbiased (%)
Workplace	11.7	13.2	75.1
Healthcare	7.4	7.0	85.6
Sports	4.2	9.3	86.4
Education	1.7	6.8	91.6
Community	0.1	6.3	93.6

Table 12: Bias distribution across Icelandic nationality scenarios aggregated over models and contact conditions (balanced scenario counts).

Robust Language Identification for Romansh Varieties

Charlotte Model Sina Ahmadi Jannis Vamvas

University of Zurich

charlotte.model@uzh.ch, sina.ahmadi@uzh.ch, vamvas@cl.uzh.ch

Abstract

The Romansh language has several regional varieties, called *idioms*, which sometimes have limited mutual intelligibility. This linguistic diversity motivates the need for a language identification (LID) system that can distinguish between these idioms, yet to date there has been no well-documented effort to build one. Since Romansh LID should also be able to recognize Rumantsch Grischun, a supra-regional variety that combines elements of several idioms, this makes for a novel and interesting classification problem. In this paper, we present a LID system for Romansh idioms based on an SVM approach. We evaluate our model on a newly curated benchmark across two domains and find that it reaches an average in-domain accuracy of 97%, enabling applications such as idiom-aware spell checking or machine translation. Our classifier is publicly available.¹

1 Introduction

Language identification (LID) is the task of automatically determining the language of text or speech. As a foundational component in natural language processing (NLP) pipelines, LID enables downstream applications such as machine translation, information retrieval, content moderation, and multilingual text processing by routing to the most appropriate language-specific system. While state-of-the-art LID systems achieve high accuracy for widely-used languages (Chen et al., 2024), distinguishing between closely-related language varieties remains a significant challenge (Burchell et al., 2024). Unlike unrelated languages that exhibit clear phonological, morphological, and lexical differences, closely-related varieties have fewer unique linguistic features, making them difficult to differentiate automatically. The challenges of LID are compounded for varieties spoken in limited geographic regions, where the scarcity of diverse

digital data hinders the development and generalization of robust LID systems.

In this paper, we present a LID system for Romansh, a collection of closely-related linguistic varieties native to Switzerland and spoken by approximately 40,000 people (Grünert, 2024). Due to Romansh’s low-resource status, no well-documented effort in the NLP community has developed a LID system capable of distinguishing between its varieties; existing multilingual LID systems either ignore it entirely or treat it as a monolithic language. Our contributions are threefold:

- (1) We curate a novel benchmark dataset spanning multiple domains (news, broadcast transcripts, textbooks, and newsroom notes) to enable systematic evaluation of idiom classification;
- (2) We develop an SVM-based classifier that achieves 97% average in-domain accuracy across all six varieties; and
- (3) We provide qualitative analysis on the LID task for Romansh idioms.

Our work demonstrates that effective LID for closely-related low-resource varieties is achievable with carefully designed features and appropriate training data, providing a model for similar efforts in other endangered and regional language contexts.

2 Romansh and its Varieties

The term *Romansh* refers to a collection of closely related linguistic varieties of Rhaetian descent native to the canton of Grisons in Switzerland. These varieties, known as *idioms*, comprise five historically distinct forms: *Sursilvan*, *Sutsilvan*, *Surmiran*, *Puter*, and *Vallader*. To facilitate official communication with the government, a sixth, artificially-created variety was introduced in 1982 to act as

¹<https://github.com/ZurichNLP/romansh-lid>

the established written standard, known as *Rumantsch Grischun* (RG) (Grünert, 2024). Communities speaking different idioms are distributed within an area of approximately 7,000 km² across Grisons.

The geographical proximity of certain idioms is reflected in their linguistic similarity. According to Lia Rumantscha, the organization tasked by the Swiss government to represent and preserve the Romansh identity, the two idioms spoken in the Rhine valley, Sursilvan and Sutsilvan, are lexically closely related, as are Puter and Vallader, spoken in the upper and lower Engadine valley respectively (Lia Rumantscha, 2015). Surmiran, spoken in the central region, often acts as a bridge between these two idiom groups. Finally, since RG was developed on the basis of Sursilvan, Vallader, and Surmiran, it shares features with multiple idioms (Anderson, 2016).

3 Related Work

3.1 Language Identification

LID has been widely studied in NLP, with growing importance for filtering the multilingual corpora used to train language models (Foroutan et al., 2026). When applied to closely-related varieties and dialects, especially within low-resource languages, the task poses significant challenges due to the limited availability of linguistic resources and annotated datasets. Research highlights the difficulty of accurately distinguishing among variants, as demonstrated in studies focused on dialectal Arabic (Dahou et al., 2025) and Indo-Aryan languages such as Bhojपुरi and Assamese (Mundotiya et al., 2021). Recent techniques explore machine learning and deep learning architectures, leveraging cross-lingual transfer learning and community-driven approaches that emphasize regional linguistic families (Salesky et al., 2021).

3.2 Romansh NLP

Ongoing projects at Lia Rumantscha for Romansh NLP focus on offering digital services to translate from Romansh into German, the most widespread official language of Switzerland, and vice versa. These services include idiom-specific dictionaries hosted across several portals: Pledari Grond covers Surmiran, Sutsilvan, RG and most recently Sursilvan directly in its interface², the Uniun dals

²<https://www.pledarigrond.ch>

Grischs³ serves Puter and Vallader, and the Dicziunari Rumantsch is an app that aggregates this data on mobile⁴. Pledari Grond also provides a spell-checker UI for RG⁵. One property shared by all these projects is that the user must select the desired idiom before translation or spell-checking can take place, limiting their usability to cases where the user already knows what idiom a text is in.

More recently, resources and models have been released that specifically focus on Romansh, including the Mediomatix Corpus (Hopton et al., 2026), containing parallel sentences extracted from schoolbooks, and WMT24++ (Vamvas et al., 2025), containing Romansh translations of the WMT24++ benchmark in machine translation. Additionally, the Swiss-made LLM Apertus has specifically incorporated Romansh data for post-training (Hernández-Cano et al., 2025, p. 94).

4 Data

For the training and evaluation of Romansh LID systems, we compile five sources of textual data spanning dictionary entries, journalistic articles, broadcast transcripts, newsroom notes, and school textbooks:

- **Pledari Grond** (PG), a comprehensive Romansh-German dictionary covering all Romansh idioms.
- **La Quotidiana** (LQ), a Romansh newspaper with daily idiom-annotated content. We use WordPress dumps from 2021 to 2025.
- **Radiotelevision Svizra Rumantscha** (RTR), validated speech transcripts from Romansh broadcasts, annotated by idiom.
- **RTR Telesguard Notes** (TG), pre-broadcast notes written by journalists in their native idioms (excluding RG).
- **Mediomatix Textbooks** (TB), parallel scholastic material per idiom (excluding RG), recently released by Hopton et al. (2026).

4.1 Preprocessing

For each source we extract the idiom label and the main text fields. We then apply minimal cleaning to all samples by i) removing intra-class duplicates (exact string match within idiom); ii) stripping HTML/markup and collapsing repeated whitespace

³<https://www.udg.ch>

⁴<https://www.dicziunari.ch>

⁵<https://www.pledarigrond.ch/rumantschgrischun/spellchecker>

and newlines; iii) dropping empty/None/non-letter-only items. Additionally, we remove source artifacts that do not carry language cues such as dictionary markers, e.g., sense numerals, “cf.” stubs, editorial signatures, worksheet placeholders and long underscore sequences. RTR content showed no recurring artifacts. Noisy or non-Rumantsch snippets were discarded and clear mislabels, if noticed visually, were manually corrected by dictionary checks.

Appendix A summarises the export statistics for each data source after this cleaning, and provides links to the publicly available datasets.

Following Bernier-colborne et al. (2023), we run exact and (where feasible) near-duplicate detection across all the data sources after merging and before splitting them. Exact duplicates judged valid in multiple idioms are kept but routed to the training split.

4.2 Named-Entity Masking

To reduce reliance on lexical memorisation, we produce two training variants: masked and unmasked. For masked, we run a fine-tuned named entity recognition model based on SwissBERT (Vamvas et al., 2023) (zero-shot on RG) with conservative heuristics (min length, standalone tokens, score ≥ 0.98) and replace matched spans with \$NE\$. Pledari Grond is left unmasked due to very few named entities and high inference cost. Because masking replaces spans in place rather than dropping samples, both variants contain the same 487,172 samples and differ only in token count (12.43M unmasked vs. 12.38M masked).

5 Experimental Setup

5.1 Data Splits

Following these steps, we finally split all the datasets into train, validation and test sets. For the latter, we use multiple test sets to disentangle domain, balance, and comparability effects.

- **Train** (train-set): PG + LQ + RTR + TB (unbalanced). Both masked and unmasked variants used in training experiments.
- **Dev** (dev-set): Balanced, in-domain RTR, **6,000** samples (1k/idiom; avg. 45.9 tokens).
- **Test-A** (test-a): In-domain, *unbalanced* LQ, **6,000** samples (avg. 528.5 tokens).

- **Test-B** (test-b): In-domain, *balanced* RTR, **6,000** samples (avg. 45.7 tokens).
- **Test-C** (test-c): In-domain, approximately *balanced* TB (no RG), **6,000** samples (avg. 85.8 tokens).
- **Test-D** (test-d): *Out-of-domain* TG (no RG), **9,607** samples (avg. 151.9 tokens).

To reflect realistic LID inputs, we did not lowercase or strip punctuation on dev/test; we only removed non-letter-only items and ensured non-empty (post-strip) text/labels.

5.2 Classification

We frame our task as a supervised multi-class classification problem where the task is to assign each text sample to one of six varieties.

We extract bag-of- n -grams features combining word unigrams and overlapping character n -grams ($n \in \{1, 2, 3, 4\}$), represented as TF-IDF vectors with sublinear term-frequency scaling and ℓ_2 normalisation. We compare four families of linear classifiers, all from scikit-learn (Pedregosa et al., 2011), each with explicit ℓ_2 regularisation by default: (i) Logistic Regression (LogisticRegression, multinomial, ℓ_2 penalty, inverse regularisation strength C , saga solver, 5,000 max iterations); (ii) linear SVM (LinearSVC, squared-hinge loss, ℓ_2 penalty, dual=False, inverse regularisation strength C); (iii) two stochastic-gradient variants (SGDClassifier)—one with hinge loss matching the SVM objective (SGD-SVM), the other with log loss matching the LR objective (SGD-LR), both with ℓ_2 regularisation strength $\alpha = 10^{-4}$, 5,000 max iterations, and early stopping; (iv) Multinomial and Complement Naïve Bayes baselines (additive smoothing $\alpha = 1.0$).

Baseline values were the scikit-learn defaults ($C = 1.0$ for SVM and LR), and were subsequently tuned by randomised search (§ 5.3). The two SGD variants share the SVM and LR loss functions respectively, but differ in optimiser: LogisticRegression uses the full-batch saga solver, whereas SGDClassifier uses mini-batch SGD with a fixed learning-rate schedule. This isolates optimiser effects from loss-function effects.

5.3 Hyperparameter Optimization

We perform 40-iteration randomized searches over pipeline hyperparameters using stratified

Classifier	Accuracy	Macro F1	Weighted F1	Macro Recall
Logistic Regression (LR)	79.5	76.9	79.1	74.6
Linear SVM	78.1	75.4	77.9	73.9
SGD (Linear SVM)	76.3	72.4	75.2	68.0
Naive Bayes (counts)	75.1	72.1	74.6	70.6
Naive Bayes (TF)	75.1	72.1	74.6	70.6
Naive Bayes (TF-IDF)	73.7	72.0	73.9	71.7
SGD (Logistic Regression)	73.5	69.0	72.0	63.6
Majority Baseline	35.1	8.7	18.3	16.7

Table 1: Results from the preliminary experiments with different baseline classifiers, where classifiers are ordered by macro F1 score.

5-fold cross-validation on a stratified 20% subset of the training data, optimising for macro F_1 (RandomizedSearchCV, `random_state = 42` throughout). The search space included the regularisation parameters of each classifier (C drawn log-uniformly from $[10^{-2}, 4]$ for SVM and $[10^{-2}, 2]$ for LR), the penalty norm (ℓ_2 , ℓ_1 , elastic net for LR; ℓ_2 and ℓ_1 for SVM), the elastic-net mixing ratio $l_1_ratio \in [0.05, 0.9]$, the SVM loss (hinge / squared-hinge), the character n -gram range ((1, 3) or (1, 4)), the word n -gram range ((1, 1) or (1, 2)), and the minimum document frequency (1 or 2). Final models were trained on both unmasked and masked variants of the training set.

The best LR configuration (penalty = ℓ_1 , $C \approx 1.94$, $l_1_ratio \approx 0.47$, character (1, 4)-grams, word (1, 2)-grams) reached 96.8 macro F_1 on the dev set, while the best SVM configuration (penalty = ℓ_1 , $C \approx 0.62$, squared-hinge loss, character (1, 4)-grams with `min_df = 2`, word unigrams only) reached 97.1 macro F_1 on the dev set.

6 Results

6.1 Overall Results

Table 1 reports the preliminary baseline comparison. All classifiers substantially outperformed the majority-class baseline, with macro F_1 scores clustering within 8 points of each other. LR achieved the highest score (76.9), followed closely by linear SVM (75.4), suggesting that for this task feature representation matters more than classifier choice. We retained LR and SVM for hyperparameter optimisation.

After tuning (§ 5.3), the ordering reversed: SVM (97.1 dev macro F_1) slightly outperformed LR (96.8). We therefore selected SVM for the remain-

Test Set	Domain	Balance	Acc.	F1
test-a	in-domain	unbal.	96.8	94.7
test-b	in-domain	bal.	98.1	98.1
test-c	in-domain	bal.	96.2	80.5
test-d	out-domain	unbal.	90.7	69.1

Table 2: Accuracy and macro F1 scores per test set. The highest scores are achieved on the balanced (bal.) vs. unbalanced (unbal.), in-domain set test-b.

ing experiments.

Table 2 presents the performance of our optimized linear SVM classifier across the four different test sets. The model achieves macro F1 scores ranging from 98.1 on the balanced in-domain test set (test-b) to 69.1 on the unbalanced out-of-domain set (test-d), demonstrating this classical machine learning approach is effective for Romansh LID under favorable conditions while struggling with out-of-domain and noisy data. These results compare favorably to similar language discrimination tasks. Recent VarDial shared tasks on Italian dialects and French varieties report best macro F1 scores of 74.6 and 34.4 respectively, while the Dravidian Language Identification task achieved a score of 93, closer to our in-domain results (Chakravarthi et al., 2021; Aepli et al., 2022).

6.2 Per Idiom Performances

Figure 1 shows confusion matrices for each test set. The model achieves consistently high recall across idioms on test-b (95.0–100.0), with perfect classification of Sutsilvan samples. Performance varies more on other sets: test-a shows recalls of 91.0–97.6, test-c shows 92.8–98.0, while test-d exhibits the widest range (63.2–96.9). The

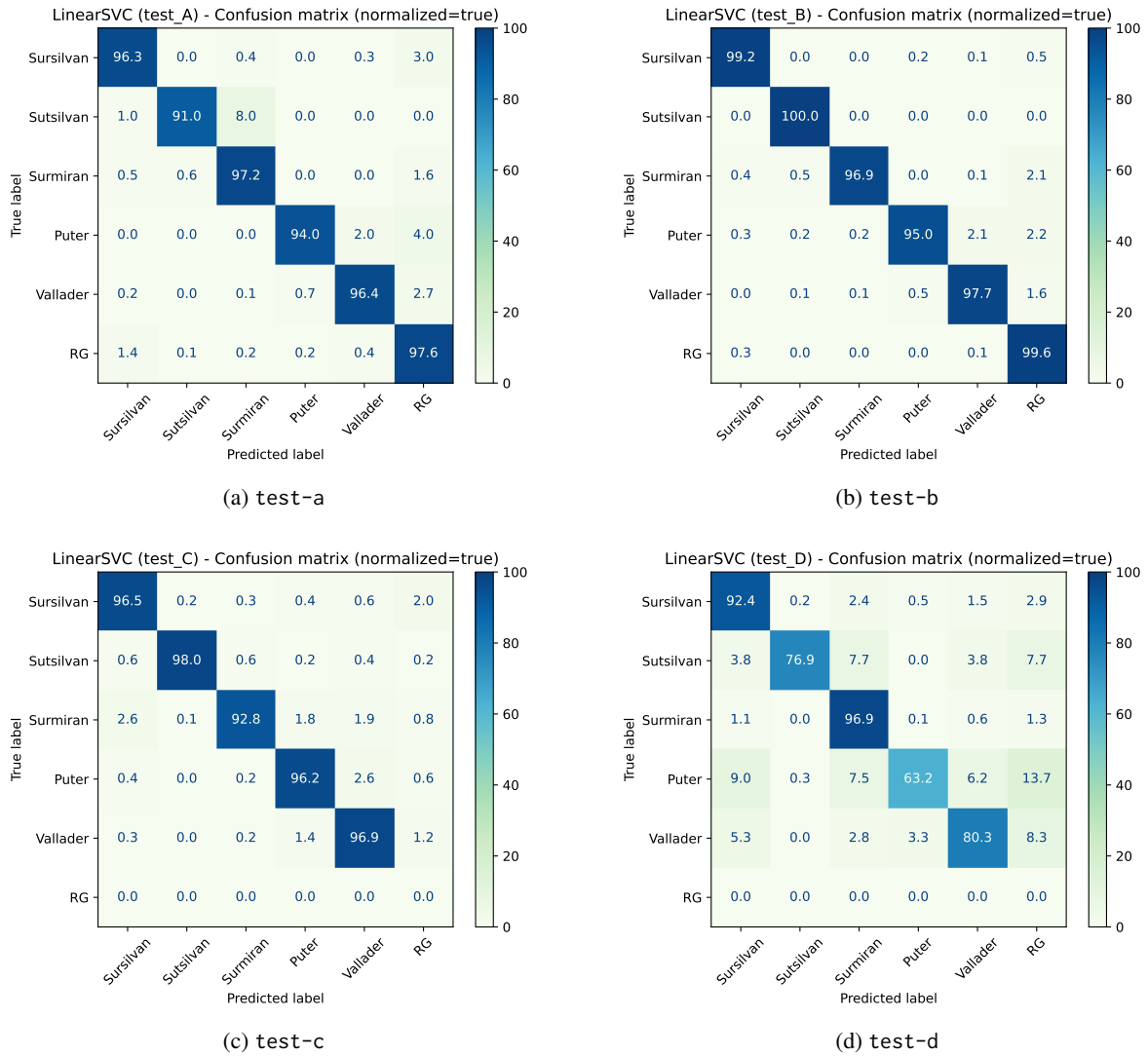


Figure 1: Row-normalized confusion matrices for all test sets. The model achieves near-perfect classification on the balanced in-domain set (test-b), while confusion increases on out-of-domain data (test-d).

Test Set	Accuracy		Macro F1	
	Masked	Unmasked	Masked	Unmasked
test-a	96.8	96.9	94.7	94.5
test-b	98.1	98.8	98.1	98.8
test-c	96.2	96.2	80.5	80.4
test-d	90.7	90.5	69.1	69.0

Table 3: Comparison of masked vs. unmasked training. No major effect can be observed.

notably low recall for Puter in test-d (63.2) warrants investigation, as samples were frequently misclassified as RG. This may reflect the informal, potentially noisy nature of the Telesguard notes data. Our analysis of misclassifications reveals several data quality issues. In test-a, many errors involved samples containing primarily named

entities or German text—artifacts from the journalism data source. In test-c, misclassifications often occurred on noisy samples containing mainly numbers, punctuation, or very short texts from the textbook data.

Idiom	1st	Type	2nd	Type	3rd	Type
Sursilvan	ei	char	scha	char	iu	char
Sutsilvan	àn	char	ù	char	eing	char
Surmiran	eir	char	dall	char	ous	char
Puter	ron	char	aunt	char	o	char
Vallader	à	char	on	char	ì	char
RG	␣	char	vegnis	word	ì	char

Table 4: Top 3 most discriminative features per idiom. “char” denotes character n -gram and “word” denotes word unigram. Character n -grams emerge as the most discriminative features across all varieties, with empty space $_$ being the most informative one for RG.

6.3 Discriminative Features

The most informative features are predominantly character n -grams rather than word unigrams, aligning with findings from similar LID tasks. We present these features in Table 4. Several patterns emerge: notably, Puter and Vallader features include underdotted characters (o, u, a, i, e) that are unique to these idioms in the Pledari Grond data. Although these features are highly discriminative for our classifier, they are not part of the standard orthography of Puter and Vallader: in Pledari Grond they function as phonetic stress markers, and would not appear in running text written in either idiom outside a dictionary setting. The whitespace character appearing as the top feature for RG is an artefact of severe class imbalance in the training data (≈ 171 K RG samples versus 44K–86K for other idioms; see Appendix B), since whitespace sequences are normalised to a single space during preprocessing and RG samples then dominate single-space n -grams.

We tested two ablations to address these artefacts: (i) restricting character n -grams to within-word boundaries only (rather than allowing them to span across whitespace), and (ii) removing character 1-grams from the feature set entirely. Both decreased overall macro F_1 , most plausibly because they remove the underdotted characters and other diacritics that supply most of the discriminative signal for Puter and Vallader.

Future work should explore balancing the training data, which we expect to be a more robust remedy for the whitespace artefact than feature-side filtering.

6.4 Impact of Named Entities

Comparing models trained on masked vs. unmasked data, as presented in Table 3, shows negligible performance differences (≤ 0.1 F1 points on most sets; on test-b the unmasked variant is 0.7 F1 higher, the largest gap). This confirms that named entities provide no idiom-specific information—unsurprising given the geographic proximity of Romansh communities. News outlets like La Quotidiana and RTR report region-wide events in multiple idioms, textbooks cover similar subjects across idioms, and RG serves as a pan-idiom standard.

7 Conclusion

We presented the first documented language identification system for Romansh idioms, capable of distinguishing between the five historical varieties (Sursilvan, Sutsilvan, Surmiran, Puter, and Vallader) and the standardised RG. Using an SVM classifier with character and word n -gram features, our model achieves up to 98% macro F1 on balanced in-domain data. Our experiments confirm that character n -grams are the most discriminative features for this task, with idiom-specific diacritics and orthographic patterns providing strong classification signals. Furthermore, we found that named entity masking has negligible impact on performance, suggesting that the classifier relies on genuine linguistic features rather than memorizing location-specific proper nouns.

Limitations

Our work has several limitations. First, the training data exhibits substantial class imbalance, with RG overrepresented due to the Pledari Grond dictionary data, which may explain artifacts such as whitespace emerging as a top discriminative feature. Second, performance degrades on out-of-domain data (test-d), where macro F1 drops to 69.1, indicating limited generalization to informal text genres like newsroom notes. Third, some discriminative features, particularly the underdotted characters unique to Pledari Grond, may not generalize to texts following standard orthographic conventions. Finally, our evaluation is restricted to written text; spoken language identification for Romansh remains unexplored. Despite these limitations, our system provides a practical tool for downstream applications such as idiom-aware spell checking and machine translation routing, and establishes

a baseline for future work on Romansh and other low-resource regional language varieties.

Acknowledgments

This work is based on a Bachelor’s thesis that was presented to the University of Zurich.⁶ We thank Lia Rumantscha and RTR for their support and for facilitating access to the data sources used in this study, including Pledari Grond, La Quotidiana, and RTR materials. We also thank Uniun dals Grischs for making dictionary data for Puter and Vallader available to us for research use, and Ignacio Pérez Prat for helpful feedback. Their commitment to preserving and promoting the Romansh language made this research possible.

References

- Noëmi Aepli, Antonios Anastasopoulos, Adrian-Gabriel Chifu, William Domingues, Fahim Faisal, Mihaela Gaman, Radu Tudor Ionescu, and Yves Scherrer. 2022. [Findings of the VarDial evaluation campaign 2022](#). In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–13, Gyeongju, Republic of Korea. Association for Computational Linguistics.
- Stephen R. Anderson. 2016. [Romansh \(rumantsch\)](#). In Adam Ledgeway and Martin Maiden, editors, *The Oxford Guide to the Romance Languages*, pages 169–184. Oxford University Press, Oxford.
- Gabriel Bernier-colborne, Cyril Goutte, and Serge Leger. 2023. [Dialect and variant identification as a multi-label classification task: A proposal based on near-duplicate analysis](#). In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 142–151, Dubrovnik, Croatia. Association for Computational Linguistics.
- Laurie Burchell, Alexandra Birch, Robert P. Thompson, and Kenneth Heafield. 2024. [Code-switched language identification is harder than you think](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian’s, Malta, March 17-22, 2024*, pages 646–658. Association for Computational Linguistics.
- Bharathi Raja Chakravarthi, Mihaela Gaman, Radu Tudor Ionescu, Heidi Jauhiainen, Tommi Jauhiainen, Krister Lindén, Nikola Ljubešić, Niko Partanen, Ruba Priyadharshini, Christoph Purschke, Eswari Rajagopal, Yves Scherrer, and Marcos Zampieri. 2021. [Findings of the VarDial evaluation campaign 2021](#). In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–11, Kyiv, Ukraine. Association for Computational Linguistics.
- Wei-Rui Chen, Ife Adebara, Khai Duy Doan, Qisheng Liao, and Muhammad Abdul-Mageed. 2024. [Fumbling in babel: An investigation into chatgpt’s language identification ability](#). In *Findings of the Association for Computational Linguistics: NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 4387–4413. Association for Computational Linguistics.
- Abdelghani Dahou, Abdelhalim Hafedh Dahou, Mohamed Amine Chérageui, Amin Abdedaïem, Mohammed A. A. Al-qaness, Mohamed Abd Elaziz, Ahmed A. Ewees, and Zhonglong Zheng. 2025. [A survey on dialect arabic processing and analysis: Recent advances and future trends](#). *Acm Transactions on Asian and Low-Resource Language Information Processing*.
- Negar Foroutan, Jakhongir Saydaliev, Grace Kim, and Antoine Bosselut. 2026. [ConLID: Supervised contrastive learning for low-resource language identification](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6693–6708, Rabat, Morocco. Association for Computational Linguistics.
- Matthias Grünert. 2024. [Rätoromanisch](#). In Elvira Glaser, Johannes Kabatek, and Barbara Sonnenhauser, editors, *Sprachenräume der Schweiz. Band 1: Sprachen*, pages 156–184. Narr Francke Attempto, Tübingen.
- Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, Barna Pasztor, Bettina Messmer, Dhia Garbaya, Eduard Frank Āurech, Ido Hakimi, Juan García Giraldo, Mete Ismayilzada, Negar Foroutan, Skander Moalla, Tiancheng Chen, Vinko Sabolčec, Yixuan Xu, Michael Aerni, Badr AlKhamissi, and 82 others. 2025. [Apertus: Democratizing open and compliant llms for global language environments](#). *Preprint*, arXiv:2509.14233.
- Zachary Hopton, Jannis Vamvas, Andrin Büchler, Anna Rutkiewicz, Rico Cathomas, and Rico Sennrich. 2026. [The mediomatix corpus: Parallel data for Romansh language varieties via comparable schoolbooks](#). In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 290–306, Rabat, Morocco. Association for Computational Linguistics.
- Lia Rumantscha. 2015. [Facts](https://www.liarumantscha.ch/sites/default/files/2023-07/PDF%20cumplet_d.pdf). https://www.liarumantscha.ch/sites/default/files/2023-07/PDF%20cumplet_d.pdf. Accessed: 2025-06-17.
- Rajesh Kumar Mundotiya, Manish Kumar Singh, Rahul Kapur, Swasti Mishra, and Anil Kumar Singh. 2021. [Linguistic resources for bhojpuri, magahi, and maithili: Statistics about them, their similarity estimates, and baselines for three applications](#). *Acm*

⁶<https://seafiler.ifi.uzh.ch/f/96df2a17539546e7a192/>

- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Elizabeth Salesky, Badr M. Abdullah, Sabrina Mielke, Elena Klyachko, Oleg Serikov, Edoardo Maria Ponti, Ritesh Kumar, Ryan Cotterell, and Ekaterina Vylomova. 2021. [SIGTYP 2021 shared task: Robust spoken language identification](#). In *Proceedings of the Third Workshop on Computational Typology and Multilingual NLP*, pages 122–129, Online. Association for Computational Linguistics.
- Jannis Vamvas, Johannes Graÿn, and Rico Sennrich. 2023. [SwissBERT: The multilingual language model for Switzerland](#). In *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 54–69, Neuchatel, Switzerland. Association for Computational Linguistics.
- Jannis Vamvas, Ignacio Pérez Prat, Not Soliva, Sandra Baltermia-Guetg, Andrina Beeli, Simona Beeli, Madlaina Capeder, Laura Decurtins, Gian Peder Gregori, Flavia Hobi, Gabriela Holderegger, Arina Lazzarini, Viviana Lazzarini, Walter Rosselli, Bettina Vital, Anna Rutkiewicz, and Rico Sennrich. 2025. [Expanding the WMT24++ benchmark with Rumantsch Grischun, Sursilvan, Sutsilvan, Surmiran, Puter, and Vallader](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 1028–1047, Suzhou, China. Association for Computational Linguistics.

A Data Sources

Source and URL to Dataset (if available)	Idiom	Total # of Samples	Total # of Tokens	Avg. # of Tokens/Sample
https://www.pledarigrond.ch	Sursilvan	34,724	92,182	2.65
	Sutsilvan	32,175	307,799	9.57
	Surmiran	43,954	301,390	6.86
	Puter	59,852	963,940	16.11
	Vallader	72,470	1,141,688	15.75
	RG	172,616	785,831	4.55
	total	415,791	3,592,830	8.64
La Quotidiana (LQ) https://huggingface.co/datasets/ZurichNLP/quotidiana	Sursilvan	6,088	3,030,828	497.84
	Sutsilvan	363	192,933	531.50
	Surmiran	1,750	1,008,736	576.42
	Puter	878	430,122	489.89
	Vallader	2,395	1,250,042	521.94
	RG	2,567	1,365,988	532.13
total	14,041	7,278,649	518.39	
Radiotelevisiun Svizra Rumantscha (RTR) https://developer.srgssr.ch/en/apis/rtr-linguistic	Sursilvan	6,979	353,205	50.61
	Sutsilvan	3,074	156,674	50.97
	Surmiran	7,196	245,931	34.18
	Puter	6,016	225,389	37.46
	Vallader	5,787	275,516	47.61
	RG	4,359	232,020	53.23
	total	33,411	1,488,735	44.56
Telesguard Notes (TG) n.a.	Sursilvan	4,931	641,844	130.17
	Sutsilvan	27	7,122	263.78
	Surmiran	3,033	261,001	86.05
	Puter	578	162,754	281.58
	Vallader	1,053	389,428	369.83
	RG	0	0	0.00
total	9,622	1,462,149	151.96	
Mediomatix Textbooks (TB) https://huggingface.co/datasets/ZurichNLP/mediomatix	Sursilvan	12,233	971,643	79.43
	Sutsilvan	12,116	1,005,356	82.98
	Surmiran	6,698	534,563	79.81
	Puter	12,238	1,023,779	83.66
	Vallader	12,283	1,019,339	82.99
	RG	0	0	0.00
total	55,568	4,554,680	81.97	

Table 5: Summary of the number of samples extracted from each data source for each idiom, along with the number of whitespace tokens across all collected samples per data source and the average number of tokens per sample.

B Data Split Statistics

Set	Idiom	Total # of Samples	Total # of Tokens	Avg. # of Tokens/Sample
train-set	Sursilvan	55,574	3,663,641	65.92
	Sutsilvan	44,188	1,342,065	30.37
	Surmiran	55,543	1,457,845	26.25
	Puter	74,355	2,353,323	31.65
	Vallader	86,313	2,372,016	27.48
	RG	171,199	1,190,808	6.96
	total	487,172	12,379,698	25.41
dev-set	Sursilvan	1,000	50,500	50.5
	Sutsilvan	1,000	50,523	50.52
	Surmiran	1,000	35,133	35.13
	Puter	1,000	36,891	36.89
	Vallader	1,000	48,482	48.48
	RG	1,000	53,577	53.58
	total	6,000	275,106	45.85
test-a	Sursilvan	1,000	502,709	502.71
	Sutsilvan	100	51,535	515.35
	Surmiran	800	454,843	568.55
	Puter	100	48,195	481.95
	Vallader	2,000	1,046,133	523.07
	RG	2,000	1,067,778	533.89
	total	6,000	3,171,193	528.53
test-b	Sursilvan	1,000	50,324	50.32
	Sutsilvan	1,000	51,542	51.54
	Surmiran	1,000	33,906	33.91
	Puter	1,000	37,401	37.4
	Vallader	1,000	48,440	48.44
	RG	1,000	52,523	52.52
	total	6,000	274,136	45.69
test-c	Sursilvan	1,250	99,690	79.75
	Sutsilvan	1,250	113,106	90.48
	Surmiran	1,000	79,629	79.63
	Puter	1,250	112,670	90.14
	Vallader	1,250	109,764	87.81
	RG	0	0	0
	total	6,000	514,859	85.81
test-d	Sursilvan	4,923	640,746	130.15
	Sutsilvan	26	7,120	273.85
	Surmiran	3,031	260,720	86.02
	Puter	576	162,488	282.1
	Vallader	1,051	388,699	369.84
	RG	0	0	0
	total	9,607	1,459,773	151.95

Table 6: Detailed statistics for all data splits by idiom, including number of samples, tokens, and average tokens per sample. The training set exhibits class imbalance, with RG containing the most samples but the shortest average length due to dictionary entries from Pledari Grond. test-c and test-d lack RG samples by construction.

Skill Extraction from Resumes and Job Offers across Six Languages

Laura Vázquez-Rodríguez^{1,2,*}, Bertrand Audrin³, Samuel Michel², Samuele Galli⁴,
Julneth Rogenhofer³, Jacopo Negro Cusa⁴, Lonneke van der Plas⁵

¹Doodle AG, Switzerland

²Idiap Research Institute, Switzerland

³EHL Hospitality Business School, HES-SO,
University of Applied Sciences and Arts Western Switzerland, Switzerland

⁴Arca24.com SA, Switzerland

⁵Università della Svizzera italiana, Switzerland

Correspondence: bertrand.audrin@ehl.ch, lonneke.vanderplas@usi.ch

Abstract

We comprehensively evaluate multiple skill extraction approaches, including rule-based, semantic, and supervised methods, using resumes and job offers in English, French, German, Italian, Spanish, and Portuguese. Due to inherent privacy concerns in Human Resources (HR) data and the high cost of manual annotations, research on identifying relevant skills for the job market remains limited, often restricted to specific domains, datasets, and entity types, and is available in only a few languages. In the context of an industrial project, we have annotated 1,200 job offers and resumes across diverse domains and six languages, through a multidisciplinary collaboration among HR researchers, NLP researchers, and HR tech professionals. Our evaluation assesses the effectiveness of these systems in a multilingual, multidomain setting, capturing both standardized job offers and highly variable resumes. The results show that supervised models achieve F1 scores of up to 0.6, while rule-based methods offer better interpretability. Furthermore, we find large differences between how skills are formulated in job offers and resumes, while the latter is understudied in academic research.¹

1 Introduction

In recent years, candidate selection in recruitment has increasingly relied on automated methods to identify relevant competencies in job postings and resumes (Guo et al., 2016; Boselli et al., 2018; Gan et al., 2024). Applicant Tracking Systems (ATS) primarily manage hiring processes with skill identification as a core function. However, the precision of existing methods remains insufficient to ensure a fair and unbiased recruitment process compared to human decision-making (Fabris et al.,

2025). Tackling this challenge is crucial for real-world NLP applications to build long-lasting and trustworthy collaboration with industry.²

The task of skill extraction has been extensively studied (Senger et al., 2024; Koppurapu, 2010; Kivimäki et al., 2013; Zhao et al., 2015), using a wide variety of methods such as vector-based methods (Javed et al., 2017; Gughani and Misra, 2020), BERT-based models (Tamburri et al., 2020; Zhang et al., 2022c, 2023) and conversational Large Language Models (LLMs) (Clavié and Soulié, 2023; Decorte et al., 2023). However, research remains narrow, focusing on specific languages, domains (e.g., IT, finance), data types (mostly job advertisements), skill categories (mostly hard skills), and extraction methods, which is very far from the actual contexts in which the HR tech industry is operating.

One of the main challenges is data scarcity, primarily due to the sensitivity of personal information in resumes. Researchers thus mainly focus on job advertisements (also referred to as job offers or job postings). While job offers are rich in career-related concepts, they strongly differ from resumes, which tend to showcase skills with more variety. Therefore, a more systematic, in-depth study is needed (Zhang et al., 2023) to address the challenges in resumes. In a practical scenario, ATS systems could not be effective when skills rely solely on job offers.

Another difficulty is the language. Linguistic differences influence how skills are presented, both in resumes and job offers. While English is sometimes considered the lingua franca for job applications (and for research on the topic), many countries primarily use their local languages, highlighting the importance of a multilingual approach. Skill extraction methods are typically designed for a monolingual setting and applied indiscriminately

*Work done while at Idiap Research Institute.

¹We will publish our non-proprietary data and models on GitHub: https://github.com/idiap/multilingual_skill_extraction.

²We discuss our perspective towards the challenges of translating research into NLP applications in Appendix A.1.

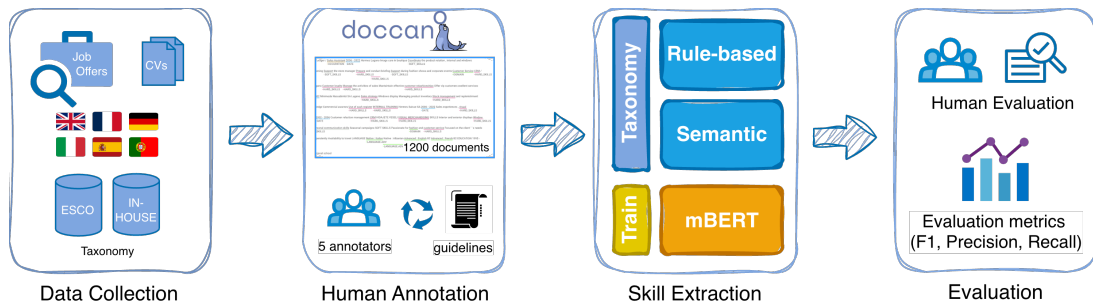


Figure 1: Overview of the end-to-end skill extraction system from *Data Collection* to the *Evaluation* stage.

across languages such as English (Zhang et al., 2022b; Bhola et al., 2020; Zhang et al., 2022a), French (Beauchemin et al., 2022), Swiss German (Gnehm et al., 2022), and Danish (Zhang et al., 2022b), without accounting for these critical differences.

A final challenge is that of explainability, which is essential to build trust. While LLMs are well known for their inference capabilities (Clavié and Soulié, 2023), they are often criticized for their lack of transparency, making it difficult to trace the source of each decision, unlike rule-based systems that rely on manually crafted taxonomies. Although rule-based approaches are not ideal due to the difficulty of keeping skill terms updated, they foster trust between systems, HR specialists, and candidates. Hybrid approaches can balance explainability and skill extraction capabilities, enhancing fairness in recruitment. These combinations can also be optimized by aligning with end-user requirements while maintaining acceptable inference speeds. Moreover, this also aligns and supports global regulations (e.g., EU Act), which urge having more transparent and fair models, especially from the deployers’ side.

This study takes place within the SEM24 Innosuisse project,³ which aims to develop skill extraction algorithms for practical applications in HR systems. This interdisciplinary initiative brings together HR specialists, HR researchers, and NLP researchers to identify skills in resumes and job offers across multiple languages and methods. Specifically, we propose a multilingual, multidomain evaluation of the skill extraction task for both job offers and resumes. For our assessment, we selected the most stable, predictable, and hardware-efficient models (Vásquez-Rodríguez et al., 2024) using hand-crafted annotations. We hypothesize that exploring

a broader range of methods, languages, and data types will contribute to the development of more accurate skill extraction techniques. Our key contributions are as follows:

1. A multilingual, multidomain implementation of skill extraction systems (i.e., rule-based, semantic, and supervised) for job offers and resumes;
2. A comprehensive analysis of the challenges in skill extraction across diverse scenarios, considering different languages, document types, and domains;
3. An assessment of the advantages and limitations of a deployed real-world NLP application in an industrial setting, with detailed error analysis.

2 Methodology

We present our task definition in Section 2.1, and further, we detail the necessary data for our experiments, including taxonomies⁴ and datasets, as described in Section 2.2 and 2.3. Section 2.4 explains the annotation process and guidelines for labeling the datasets. Finally, Section 2.5 presents the selected models, which serve as the foundation for our multilingual experiments.

2.1 Task Definition

Our main task aims to understand the performance of the skill extraction task for both resumes and job offers, in the aforementioned scenarios, including English, French, German, Italian, Portuguese, and Spanish. The selection of these first languages was determined by the availability of the annotations.

⁴We refer to a taxonomy as a collection of interconnected terms that define relationships, for example, between skills and occupations. The entities that describe the taxonomy can vary between sources.

³<https://www.idiap.ch/en/scientific-research/projects/SEM24>

Surface Properties →		Total Entities		Unique Entities		Unique/Total		Avg. Len	
Text ↓	Language	Hard	Occ	Hard	Occ	Hard	Occ	Hard	Occ
Jobs (our annotated data)	EN	878	322	777	109	0.88	0.34	37.53	22.19
	FR	1119	126	886	107	0.79	0.85	30.67	29.13
	IT	704	116	651	101	0.92	0.87	53.67	22.52
	DE	848	133	700	115	0.83	0.86	19.56	21.26
	ES	476	81	461	73	0.97	0.90	42.09	21.72
	PT	667	81	560	68	0.84	0.84	30.87	21.37
Jobs (Green)	EN	12573	2571	10079	1591	0.80	0.62	32.67	17.49
Resumes (our annotated data)	EN	4024	692	3020	565	0.75	0.82	20.01	24.85
	FR	2063	645	1700	466	0.82	0.72	25.76	21.66
	IT	1985	645	1613	435	0.81	0.67	26.77	21.91
	DE	2075	691	1592	445	0.77	0.64	24.35	17.43
	ES	1086	271	922	241	0.85	0.89	30.74	25.42
	PT	3312	729	2439	447	0.74	0.61	28.79	24.78

Table 1: We report the number of total and unique entities and its ratio, and average character length of hard skills, and occupations in resumes and job offers.

Further, we also wanted to understand the incremental benefit of using simple rule-based systems, transitioning to semantic, and finally, supervised models, while keeping the explainability of the skill extraction outcomes under control. In Figure 1, we present an overview of the end-to-end system.

2.2 Taxonomies

We selected two taxonomies for the rule-based and semantic system, which model industrial and publicly available data to search for skills in text:

ESCO_DB: a collection of 3,039 occupations and 13,939 skills, translated in 28 official languages to standardize the European labor market.⁵ This taxonomy is publicly available and widely used in previous work (Zhang et al., 2023; Decorte et al., 2023; Li et al., 2023). For our experiments, we extracted 131,623 entities relevant to the skill extraction task in English, with similar proportions across all languages.

IN-HOUSE_DB: a collection of 10,379 skill entities from our industrial partner. These entities were manually extracted from resumes and job offers from different European job markets. Each concept was also curated by specialized annotators and manually translated into more than 10 languages. For our study, we selected a subset of these languages due to the significant annotation effort required for producing reliable gold-standard data. Furthermore, we were also constrained by the available annotators for each language, the scope, and the timeline of the industrial project.

⁵<https://esco.ec.europa.eu/en/use-esco/download>

2.3 Datasets

For training our supervised models in Section 2.5, we selected job offers and resume datasets for both academic⁶ and industrial settings. In particular, for our industrial dataset, **Arca24_JOB** and **Arca24_CV**, we obtained job offers and resumes from our industrial partner, who provides HR services and software to companies within the European market. From the provided data, we randomly sampled 100 resumes and 100 job offers for each of the 6 selected languages, resulting in a total of 1200 documents from ~45 different domains (e.g., Engineering, Administration, Management). These documents were manually annotated by HR researchers and specialists with the following entities: *Degrees / Certifications, Domain, Hard Skills, Knowledge, Language, Occupations, and Soft Skills*. Further details of the annotation process are shared in Section 2.4.

All these datasets have different labeling schemas for each entity. Thus, we have remapped or discarded the original labels, considering only hard skills and occupations in this study. We show our datasets’ entity distribution between hard skills and occupations, including total and unique counts and ratio between these two metrics and the average length of entities (by characters) in Table 1.

2.4 Annotation Guidelines

To identify skills within our dataset, we first developed well-defined annotation guidelines, relying on both in-house expertise (i.e., two HR researchers) and HRM literature, focusing on principles of KSAOs (Campion et al., 2011), which refers to knowledge, skills, abilities, and other at-

⁶We include the details of our selected public dataset **Green** in the Appendix A.2.1.

Evaluation				Exact (F1-score)			Partial (F1-score)			Time
DB	Test	Model	Lang.	Skills	Occ.	Overall	Skills	Occ.	Overall	(min)
IN-HOUSE	Green	Rule-based	EN	0.145	0.275	0.179	0.258	0.419	0.300	5.200
ESCO				0.166	0.355	0.216	0.361	0.493	0.396	57.800
IN-HOUSE		Semantic		0.135	0.280	0.173	0.265	0.483	0.322	5.600
ESCO				0.172	0.210	0.185	0.369	0.333	0.357	25.600
-				mBERT	0.351	0.603	0.418	0.555	0.705	0.595

Table 2: Comparison of skill extraction methods. We present the F1-score (exact, partial) results on **Green** dataset (Job Offers) for Rule-Based/Semantic (taxonomy matching) vs. Supervised (labeled data fine-tuning) systems.

tributes. Each category was defined and illustrated with examples. The annotation team met in person to discuss and align on the definitions of skills in English and to test the guidelines⁷ on a set of sample resumes. These initial annotations were collectively reviewed, and the guidelines were amended to clarify any ambiguous cases. Annotators then carried out a first round of annotations in their respective target languages, followed by an adjudication phase to resolve conflicts and ensure full alignment among the team.⁸

The labeling of entities was done by 4 annotators with previous experience in skills identification and/or annotation. All annotations were done using Docanno (Nakayama et al., 2018). The work of the annotators was distributed according to their language proficiency as follows: English-Spanish, French-German, Italian-Portuguese. Annotations in Italian, French, and Spanish were conducted by native speakers, whereas the others were done by annotators with a proficient C2 language level. Annotators gathered to review some uncertainties once all annotations had been conducted. Some degree of variability in annotations is likely to occur, as it reflects 1) variability in the content of the corpus of resumes and job offers annotated and 2) variability in the cultural norms and expectations of resumes and job offers across the six target languages.

2.5 Models

We experimented with various models that differ not only in complexity and level of explainability but also in resource requirements. While GPUs are commonly used, their implementation may not

be feasible for all businesses due to their high resource consumption and associated cost. Moreover, our NLP application requires that every decision is fully traceable and explainable for system improvement. Relying on the nondeterministic output of conversational LLMs compromises this necessary level of auditability and explainability. Therefore, we selected a rule-based system, a semantic model, and a supervised system for our benchmark.⁹ In the following section, we elaborate in the details of our supervised model and explain further on how it contributes better into the explainability of our results.

2.5.1 Supervised model

We selected multilingual BERT-base¹⁰ as our main experimental model for our human-annotated data.¹¹ This model can both understand the languages under study and also use minimal GPU resources. Further, we chose this general-purpose model over a pre-trained skill extraction model given that the latter does not always perform better on the skill extraction task due to the learned biases from previous approaches (Vásquez-Rodríguez et al., 2024). Larger models could potentially yield better scores in a single domain, but the objective of this work is to assess the tangible value and customer benefit delivered by the incremental rule-based to supervised journey. For this, we use models that are resource-efficient enough (e.g., deployable locally without GPUs), rather than achieving state-of-the-art benchmarks. The deterministic nature of BERT-based models allows for controlled outputs, ensuring that decisions can be traced and

⁷Due to privacy and intellectual property concerns, we limit the disclosure of our guidelines. However, we describe our annotation strategies as thoroughly as possible to maximize reproducibility.

⁸Given the variety of the languages involved in the project and the proficiency level in various languages of the annotation team (consisting of 4 people), inter-agreement was not feasible for any other language than English. The focus was rather put on a very thorough training procedure and a strong alignment regarding annotations across the team.

⁹Further details of the rule-based and semantic systems are explained in the Appendix A.3.2.

¹⁰<https://huggingface.co/google-bert/bert-base-multilingual-uncased>

¹¹The inclusion of multiple supervised baselines is planned as future work by our industrial partner. We prioritize demonstrating resource-efficient, incremental value over exhaustive comparative analysis. Moreover, the detailed investigation of supervised model variations has been the subject of our previously published work (Vásquez-Rodríguez et al., 2024).

explained to end users. In order to support the explainability of our approach, we have mapped predicted skills to an existing taxonomy, which indirectly justifies the reasoning behind each selection. Any future additions of skills to the model will be managed through a controlled process involving in-house annotation of previously undetected skills.

3 Experimental Setup

In this section, we present the experimental steps of our proposed methods, including the technical implementation details. For the rule-based and semantic systems, we have standardized the code into Python libraries to systematically run on the provided inputs. As for the supervised system, we performed fine-tuning of the model with the training and the validation set. We trained the skill extraction task as a span-based approach similar to Named Entity Recognition (NER)¹² where the final model was selected based on the F1-score on the validation set. We chose the best model for all the epochs run.

Lang.	Dataset	Categories						
		0	1	2	3	4	5	6
EN	Job Offers (Green)	45.1%	4.5%	3.6%	6.3%	15.3%	9.0%	16.2%
EN	Job Offers (our annotated data)	15.8%	5.2%	9.5%	3.1%	16.9%	8.4%	41.1%
FR		5.7%	63.9%	0.8%	1.6%	6.6%	18.0%	3.3%
IT		9.7%	6.5%	3.2%	17.7%	21.0%	11.3%	30.7%
DE		34.6%	16.4%	10.6%	1.9%	3.9%	6.7%	26.0%
PT		23.3%	15.0%	6.7%	8.3%	8.3%	18.3%	20.0%
ES		23.3%	7.8%	28.9%	2.2%	6.7%	16.7%	14.4%
EN	Resumes (our annotated data)	26.6%	5.3%	10.6%	1.1%	3.2%	17.0%	36.2%
FR		39.3%	0.9%	29.5%	0.0%	4.5%	8.9%	17.0%
IT		37.9%	6.0%	4.3%	0.9%	3.45%	24.1%	23.3%
DE		31.8%	15.1%	13.5%	1.6%	6.4%	9.5%	22.2%
PT		44.0%	2.4%	0.0%	2.4%	0.0%	26.4%	24.8%
ES		18.4%	1.0%	12.2%	1.0%	7.1%	12.2%	48.0%

Table 3: Results for the human evaluation

For the evaluation, we identified the entities in the corpora and evaluated the output IOB files against the reference using the *nervaluate* Python library.^{13,14} As for the evaluation schema, we report both the exact and partial scores (i.e., the identification of at least one word of the target skill) to adjust to the challenge of finding fixed entities in

¹²We used a NER-based evaluation to leverage existing resources and tools. Despite challenges in consistently defining concept boundaries, our results align with prior work.

¹³<https://pypi.org/project/nervaluate/>

¹⁴We selected IOB standard due to the availability of corpora for this task and for its compatibility with the annotations output format.

such a variable task. Finally, we performed a human evaluation of the supervised system output in Section 3.1.¹⁵

3.1 Human Evaluation

We conducted a human evaluation on 30 sample sentences per language, per document type (i.e., job offers and resumes), for a total of 360 sentences (see Table 3). These samples were randomly drawn from the test set of supervised model outputs. The primary goal was to analyze error patterns and identify possible drawbacks of neural models. To focus on the error analysis, we filtered out all correctly predicted samples. The remaining predictions and their corresponding references were displayed side by side in IOB format, token by token.

The human evaluation was carried out by three evaluators with prior experience in skill annotation. When possible, we would assign different annotators for labeling and human evaluation. This task assessed possible errors between the predicted output and the reference. Each entity is classified into different error categories (Nguyen et al., 2024), which are further explained in Section 3.1.1.

3.1.1 Human Evaluation Categories

For the human evaluation task, we categorized the differences between the predicted output and the reference. Each entity could be classified into the following categories: 1) *Skill definition misalignment*, when the prediction includes a career-related term that is not a skill; 2) *Wrong extraction*, when the detected entity is entirely unrelated to any skill; 3) *Conjoined skills*, when two distinct skills are incorrectly merged into one (e.g., develop report software and statistical software should be treated as two separate skills); 4) *Extended span*, when the predicted entity is longer than the correct reference; 5) *Incorrect annotations*, when the human annotation itself is imprecise, containing incorrect values in the reference; 6) *Other* cases, such as when the prediction is shorter than the reference or when the reference skill was not predicted at all. Additionally, we also label the correct entities when the prediction is equal to the reference. We refer to this classification as category 0.

4 Results and Discussion

We present our results evaluating the performance of the skills detection task using F1-score with an

¹⁵We report our data processing steps and training parameters in the Appendix A.

Evaluation				Exact (F1-score)			Partial (F1-score)			Time
DB	Test	Model	Lang.	Skills	Occ.	Overall	Skills	Occ.	Overall	(min)
IN-HOUSE		Rule-based		0.043	0.720	0.108	0.170	0.840	0.235	0.535
ESCO		Rule-based		0.047	0.270	0.078	0.230	0.297	0.239	6.000
IN-HOUSE		Semantic	EN	0.016	0.667	0.075	0.165	0.792	0.221	2.300
ESCO		Semantic		0.014	0.142	0.038	0.189	0.157	0.183	17.900
-		mBERT		0.220	0.560	0.247	0.405	0.680	0.427	1.900
IN-HOUSE		Rule-based		0.000	0.000	0.000	0.065	0.172	0.081	0.610
ESCO		Rule-based		0.022	0.037	0.024	0.160	0.222	0.170	2.600
IN-HOUSE		Semantic	FR	0.018	0.118	0.031	0.156	0.353	0.182	3.000
ESCO		Semantic		0.045	0.071	0.050	0.202	0.235	0.207	7.600
-		mBERT		0.443	0.591	0.464	0.595	0.659	0.605	2.200
IN-HOUSE		Rule-based		0.022	0.296	0.083	0.172	0.481	0.242	0.370
ESCO		Rule-based		0.017	0.065	0.027	0.214	0.226	0.216	1.300
IN-HOUSE	Jobs (our annotated data)	Semantic	IT	0.038	0.345	0.086	0.274	0.552	0.317	2.600
ESCO		Semantic		0.034	0.067	0.045	0.267	0.157	0.230	7.100
-		mBERT		0.135	0.615	0.235	0.419	0.692	0.476	2.800
IN-HOUSE		Rule-based		0.085	0.296	0.124	0.178	0.444	0.228	0.357
ESCO		Rule-based		0.036	0.303	0.097	0.098	0.333	0.152	0.768
IN-HOUSE		Semantic	DE	0.106	0.529	0.166	0.213	0.706	0.282	2.700
ESCO		Semantic		0.069	0.186	0.104	0.202	0.244	0.215	6.900
-		mBERT		0.299	0.667	0.367	0.476	0.738	0.524	2.200
IN-HOUSE		Rule-based		0.022	0.000	0.021	0.110	0.200	0.115	0.352
ESCO		Rule-based		0.027	0.000	0.025	0.195	0.231	0.198	1.800
IN-HOUSE		Semantic	ES	0.039	0.000	0.037	0.221	0.333	0.225	2.200
ESCO		Semantic		0.042	0.000	0.036	0.211	0.108	0.197	9.400
-		mBERT		0.355	0.800	0.378	0.505	0.800	0.520	1.800
IN-HOUSE		Rule-based		0.028	0.095	0.037	0.099	0.143	0.104	0.387
ESCO		Rule-based		0.068	0.258	0.092	0.179	0.323	0.197	1.400
IN-HOUSE		Semantic	PT	0.116	0.483	0.155	0.252	0.586	0.288	2.500
ESCO		Semantic		0.065	0.167	0.076	0.173	0.278	0.185	6.100
-		mBERT		0.444	0.357	0.429	0.595	0.500	0.578	2.100

Table 4: Comparison of skill extraction methods. We present the F1-score (exact, partial) results on the **Job Offers** dataset for Rule-Based/Semantic (taxonomy matching) vs. Supervised (mBERT, labeled data fine-tuning) systems.

exact match assessment. In Table 4, we show supervised systems trained in job offers that achieve the highest score in French, followed by Portuguese and English-Green. For the state-of-the-art results, we present the Green dataset in Table 2, where, for English, results remained competitive throughout. For resumes, supervised systems also perform well, as observed in Portuguese, followed by Italian, and English in Table 5. These results aim to characterize the breadth of variability real-word skill descriptions across document types (job offers vs resumes), domains, and different languages, as opposed to optimizing for benchmark performance on the skill extraction task. And consequently, these show how approaches that are currently being used in industry fare on these data types. Furthermore, we can confirm that the state-of-the-art performance on the academic Green Dataset (Nguyen et al., 2024) is comparable to our supervised model, which is expected to be higher ($\sim 10\%$) compared to our F-measure due to the choice of a smaller multilingual model to support a minimal resource setting with multiple languages.

In relation to the detection of occupations, results are more straightforward, yielding scores of up to

0.8 that are significantly higher than those of hard skills, when considering the best performance in the partial evaluation. Most methods demonstrate efficient memory consumption, with CPU-based approaches requiring a maximum of ~ 5 GB and GPU-based methods utilizing up to ~ 12.85 GB of RAM. Similarly, runtime remain reasonable across all approaches, except for the rule-based method on the Green dataset, which can take up to an hour due to its size. In industrial settings, optimizing resources and performance is crucial. Poor real-time skill extraction performance can negatively impact user experience and the management of large-scale data. Additionally, scaling up resources (e.g., adding more RAM) can significantly increase costs. Therefore, the trade-offs between different approaches, such as CPU-based vs. GPU-based methods, must be carefully justified.

Our study highlights key differences between job offers and resumes (Section 4.1), annotation difficulties (Section 4.2), multilingual considerations (Section 4.3), the trade-off between explainability, generalization and fairness (Section 4.4) and challenges involved in the transition of research into NLP applications (Appendix A.1).

Evaluation				Exact (F1-score)			Partial (F1-score)			Time
DB	Test	Model	Lang.	Skills	Occ.	Overall	Skills	Occ.	Overall	(min)
IN-HOUSE		Rule-based		0.160	0.226	0.171	0.253	0.481	0.289	2.000
ESCO		Rule-based		0.159	0.100	0.144	0.267	0.217	0.254	17.500
IN-HOUSE		Semantic	EN	0.130	0.236	0.146	0.217	0.535	0.263	3.900
ESCO		Semantic		0.119	0.074	0.106	0.217	0.156	0.200	19.700
-		mBERT		0.382	0.400	0.385	0.487	0.531	0.495	2.100
IN-HOUSE		Rule-based		0.026	0.260	0.068	0.124	0.410	0.175	2.600
ESCO		Rule-based		0.050	0.107	0.062	0.191	0.321	0.218	10.300
IN-HOUSE		Semantic	FR	0.057	0.281	0.093	0.184	0.545	0.242	3.600
ESCO		Semantic		0.044	0.050	0.046	0.190	0.249	0.203	8.000
-		mBERT		0.252	0.364	0.269	0.407	0.580	0.434	2.200
IN-HOUSE		Rule-based		0.111	0.442	0.197	0.258	0.565	0.338	0.917
ESCO		Rule-based		0.023	0.425	0.125	0.200	0.523	0.282	3.900
IN-HOUSE	Resumes (our annotated data)	Semantic	IT	0.095	0.444	0.171	0.295	0.632	0.369	3.600
ESCO		Semantic		0.024	0.203	0.076	0.252	0.334	0.276	7.600
-		mBERT		0.405	0.444	0.414	0.597	0.606	0.599	3.300
IN-HOUSE		Rule-based		0.069	0.448	0.168	0.151	0.510	0.245	0.763
ESCO		Rule-based		0.090	0.122	0.100	0.194	0.223	0.204	1.500
IN-HOUSE		Semantic		0.124	0.613	0.226	0.228	0.717	0.329	3.700
ESCO		Semantic	DE	0.074	0.165	0.107	0.197	0.231	0.210	7.800
-		mBERT		0.257	0.459	0.315	0.426	0.574	0.468	2.400
IN-HOUSE		Rule-based		0.045	0.333	0.085	0.132	0.476	0.179	1.200
ESCO		Rule-based		0.017	0.299	0.061	0.115	0.463	0.170	3.200
IN-HOUSE		Semantic	ES	0.085	0.435	0.128	0.228	0.587	0.272	3.000
ESCO		Semantic		0.042	0.139	0.065	0.206	0.222	0.210	10.300
-		mBERT		0.265	0.509	0.301	0.382	0.582	0.411	4.200
IN-HOUSE		Rule-based		0.029	0.217	0.058	0.141	0.337	0.172	1.300
ESCO		Rule-based		0.012	0.155	0.038	0.127	0.250	0.150	3.800
IN-HOUSE		Semantic	PT	0.038	0.279	0.070	0.185	0.589	0.239	4.100
ESCO		Semantic		0.022	0.114	0.040	0.195	0.247	0.205	7.900
-		mBERT		0.483	0.532	0.491	0.598	0.662	0.608	2.400

Table 5: Comparison of skill extraction methods. We present the F1-score (exact, partial) results on the **Resumes** dataset for Rule-Based/Semantic (taxonomy matching) vs. Supervised (mBERT, labeled data fine-tuning) systems.

4.1 Job offers vs Resumes

Job offers are more standardized than resumes, as organizations often follow similar guidelines for all their job openings. However, they often contain multiple career-related terms that are not necessarily related to skills specifically required for the position at hand, as shown in Table 3 (category 1), which can pose challenges for skill annotation and extraction. Job offers are also much more available for collection and analysis than resumes. This has two important consequences. First, their prevalence and availability on the internet could contribute to higher performance on publicly available datasets for English job offers, such as Green (Table 2). In contrast, performance is lower for IN-HOUSE job offers as shown in Table 4.

Second, academic research has focused on job offers for availability reasons. However, job offers are very distinct from resumes in how they are structured, in the way they formulate skills and experience, and in the information they contain. This is bound to eventually limit the quality of the systems if trained on a non-representative dataset. A notable difference is seen in the partial evaluation of the supervised method. The success of skill

detection varies between resumes and job offers, with some cases achieving a balanced trade-off between precision and recall, while others do not.

Furthermore, job offers and resumes vary across industries and languages. Although we used random sampling to select a meaningful population of job offers and resumes, the presence of skills varies according to the domain, their relative occupation, and the target language. Some job advertisements are concise, spanning only a few lines, while others are more detailed. A similar pattern is observed in resumes, which are highly personal and vary in format and style. For example, certain domains such as technology-oriented datasets (i.e., Green) show that skills are more straightforward to identify (see Table 6). In contrast, domains such as management might need a higher level of inference. Moreover, language and nationality play a role as norms on what to put in a resume, and what is legally required to write in a job offer varies.

4.2 Annotations Challenges

Although we established clear annotation guidelines based on academic literature and validated these with HR experts, the variability of resumes

across disciplines (e.g., software developer vs manager) and languages still poses a challenge for skill annotation consistency. As an example, we present the experiments for Italian job offers (F1-score: 0.235), where annotation quality significantly contributed to the poor performance. A manual review revealed that entity annotations in Italian were long, containing a higher number of words compared to other languages, which severely impacted the effectiveness of the entity-oriented detection approach. Our error analysis highlights how inconsistencies in gold annotations (see Table 3, category 5) can also impact model performance. The detailed error analysis shows that resumes present greater challenges, with most errors falling into Category 5 (annotation errors) or Category 6 ("other"), where the model struggles not only to predict a skill but also to align with the reference annotations (e.g., shorter annotations). The definition of skill is complex, whereas occupations are more straightforward to identify in both resumes and job offers, as seen when comparing the F1-scores of "Skills" and "Occ." in Tables 2, 4 and 5. We additionally show examples of these challenges in Table 6 and 7 (Appendix), where precise skill boundaries are challenging to define.

Model	Taxonomy	Example
Reference	-	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C
Rule-based	IN- HOUSE	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C
Rule-based	ESCO	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C
Semantic	IN- HOUSE	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C
Semantic	ESCO	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C
mBERT	-	From my experience, I learned and used different languages PHP HTML CSS Javascript jQuery SQL Visual Basic Linux Bash FileMaker Script C

Table 6: System outputs from English resumes

4.3 Multilinguality

Rule-based and semantic systems demonstrated similar performance (measured by exact F1-score) across languages (up to ~ 0.2), while more significant performance differences were observed in the supervised results for job offers in Table 4, comparing Italian and French. Overall, the mBERT multilingual model outperformed rule-based and

semantic systems. From our error analysis, we can notice that 1) In certain industries, the labor market is standardized to English, leading to a mix of local language and English terms within job descriptions, which could have a positive or negative impact in performance depending on the selected multilingual system; 2) English resumes are likely to be written by non-native speakers, which potentially can influence the quality of the submitted resume.¹⁶

4.4 Explainability, Generalization, and Fairness

Rule-based and semantic methods show a lower performance compared to supervised approaches, but offer high explainability because extracted skills can be directly linked to a curated taxonomy and are relatively easy to adapt, albeit at the cost of labor-intensive and expensive taxonomy maintenance. Supervised systems demonstrate better generalization, but their performance is highly dependent on the quality, quantity, and diversity of the data, as evidenced by the discussed results.

Taxonomy-driven methods are inherently fair, as concepts remain consistent across target languages. In contrast, neural methods may introduce cultural biases, potentially favoring or disadvantaging specific candidates. In some cases, a detailed taxonomy, such as the ESCO taxonomy, can enhance performance by using a rule-based system and more standardized job offers. However, for standalone academic datasets, the results appear to be "artificially" high, as in Green for partial evaluation. The large difference between this dataset and our multilingual datasets from actual client data suggests that the skill extraction task may be harder than one would expect from academic data, which is often limited to a particular domain, language, and to the relevant phrases with explicit skill mentions. For practical purposes, it is not necessary to exhaustively detect every skill, as many are redundant. A representative sample is often sufficient to provide a clear sense of the candidate's profile and to support manual candidate analysis. Therefore, we conducted a human error analysis to analyze the quality of system outputs and the distribution of errors. As shown in Table 3, for English job offers (Green), nearly 50% of entities were correctly identified (Category 0), suggesting a lower level of complexity in these documents compared to resumes, which had a success rate of only 26.60%.

¹⁶We further discuss the linguistic and cultural aspects of the skill extraction task in the Limitations section.

We found no conclusive evidence linking specific error types to the multilingual aspect, particularly given the additional differences in domain distribution across languages. Except for French and Spanish, most failures in the supervised models fall into categories 5 and 6, where errors stem from incorrect human annotations, missing predictions, or shorter-than-expected extractions. The significance of these analyses extends beyond research, as fair and reliable candidate selection remains a highly sensitive task, requiring human oversight to ensure trustworthiness and mitigate biases in automated hiring systems. The controllability and explainability of these systems are essential for the engagement and trust of stakeholders.

4.5 Reproducibility

Due to the proprietary nature of the job offers and resumes provided by our industrial partner, the release of the primary dataset is restricted. We evaluated pseudonymization as an alternative; however, the high density of sensitive personal information within resumes makes full anonymization and subsequent public release complex from both legal and ethical perspectives.

To ensure reproducibility while respecting these constraints, we adopt the following strategies:

1) Experiments on public data: To mitigate privacy constraints, we additionally report results using public job offers and taxonomies (e.g., the Green dataset and ESCO taxonomy);

2) Methodological transparency: Our study encompasses hard-skill extraction across job offers and resumes, utilizing multiple languages and resource-wise incremental baselines. We also include comprehensive details of our methodology to ensure these benchmarks remain as reproducible as possible, providing a transparent framework for our real-world skill extraction approach.

While we acknowledge the trade-offs inherent in using proprietary data, this study offers a rare, large-scale analysis of skill extraction in authentic industrial scenarios. We believe the broad scope and practical insights of this research provide significant value to the community beyond what is achievable with restricted or synthetic datasets.

5 Conclusion

Traditionally, the skill extraction task has been approached with a limited scope, focusing on one or two languages, exclusively analyzing job of-

fers, or relying on resource-intensive methods. Our work presents the first experiment of its kind that proposes and integrates different methodologies at multiple levels of complexity, languages, and side-by-side job offer and resume data through an extensive annotation effort covering 1200 resumes and job offers. We achieved an improved performance over production across different system variations, evaluation schemes, and data settings, detected a higher volume of qualitative instances, and provided valuable insights to guide future advancements. Finally, we present our discussions on future work in the Appendix A.4.

Limitations

In this section, we outline the limitations of our research in multiple aspects to be considered as follows:

Data release: As discussed in Section 4.5, we present our results using publicly available job offers as well, as our analysis builds on proprietary job offers and resumes from our industrial partner. However, we provide a detailed description of our research methodology to enhance reproducibility.

Taxonomies distribution: Our taxonomies are fundamentally different and non-standardized between them. The ESCO DB was created to standardize skills for the European job market, but it presents a formalized approach that does not reflect how people express skills in real-world resumes. In contrast, the IN-HOUSE taxonomy is crafted directly from the actual occurrences within real resumes and job offers, making it more practical. The ESCO DB is freely available. However, we refrain from providing specific distribution details of the IN-HOUSE taxonomy due to proprietary and privacy concerns.

NER evaluation: Another limitation is the use of the NER approach for evaluation, which allows the use of existing resources to assess the quality of our annotations and the use of open-source tools for entity labeling. However, we acknowledge that it is difficult to capture the boundaries of each concept given the variable and subjective nature of the labeling task, resulting in overall lower scores, but still consistent with previous work.

Cultural nuances in the annotation task: We acknowledge that variations in hiring cultures across domains and languages can impact our work.

However, a comprehensive evaluation of these nuances would require a dedicated study that falls outside the scope of the current paper.

Annotation-related performance differences:

We follow up on the discussion of the differences in performance in language as Italian, which are fundamentally attributed to the variability in annotation. We established rigorous, consistent guidelines and performed our annotations with in-house domain specialists instead of crowdworkers. Still, it cannot eliminate the inherent subjectivity present in human annotation. We have included this variability in our experiments to reflect real-world complexity. The persistence of this challenge for many years underscores why skill extraction remains a difficult research task.

Ethical Considerations

We ensure strict adherence to the corresponding data and privacy through non-disclosure agreements. Furthermore, we have obtained all necessary licenses and data consents for managing job offers and resume information in compliance with relevant regulations. Additionally, no data was published in the cloud; all operations were handled locally on premises. Furthermore, the final output of the skill extraction systems was always mapped to a controlled, real-world taxonomy, which effectively prevented the exposure of any sensitive data derived from the raw source. However, due to the sensitive nature of these resources and the ethical considerations involved, we are unable to share them publicly. Furthermore, we recognize the importance of maintaining fairness, transparency, and accountability in automated hiring systems. Our approach is designed to eventually support recruiters by providing them with additional insights, without delegating candidate selection decisions to any algorithm. This ensures that final hiring decisions remain under human supervision, mitigating potential biases and ethical concerns.

References

David G. Allen and James M. Vardaman. 2017. Recruitment and retention across cultures. *Annual Review of Organizational Psychology and Organizational Behavior*, 4(1):153–181.

David Beauchemin, Julien Laumonier, Yvan Ster, and Marouane Yassine. 2022. "fijo": a french insurance soft skill detection dataset. *arXiv*.

Akshay Bhola, Kishaloy Halder, Animesh Prasad, and Min-Yen Kan. 2020. Retrieving skills from job descriptions: A language model based extreme multi-label classification framework. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5832–5842, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Roberto Boselli, Mirko Cesarini, Fabio Mercorio, and Mario Mezzanzanica. 2018. Classifying online job advertisements through machine learning. *Future Generation Computer Systems*, 86:319–328.

Michael A. Campion, Alexis A. Fink, Brian J. Ruggeberg, Linda Carr, Guy M. Phillips, and Ronald B. Odman. 2011. Doing competencies well: Best practices in competency modeling. *Personnel Psychology*, 64(1):225–262.

Benjamin Clavié and Guillaume Soulié. 2023. Large language models as batteries-included zero-shot esco skills matchers. In *RecSys in HR'23: The 3rd Workshop on Recommender Systems for Human Resources, in conjunction with the 17th ACM Conference on Recommender Systems, September 18–22, 2023, Singapore, Singapore*.

Jens-Joris Decorte, Severine Verlinden, Jeroen Van Haute, Johannes Deleu, Chris Develder, and Thomas Demeester. 2023. Extreme multi-label skill extraction training using large language models. In *International Workshop on AI for Human Resources and Public Employment Services (AI4HR&PES) at ECML-PKDD*.

Alessandro Fabris, Nina Baranowska, Matthew J. Dennis, David Graus, Philipp Hacker, Jorge Saldivar, Frederik Zuiderveen Borgesius, and Asia J. Biega. 2025. Fairness and bias in algorithmic hiring: A multidisciplinary survey. *ACM Trans. Intell. Syst. Technol.*, 16(1).

Chengguang Gan, Qinghao Zhang, and Tatsunori Mori. 2024. Application of llm agents in recruitment: A novel framework for resume screening. *Preprint*, arXiv:2401.08315.

Ann-sophie Gnehm, Eva Bühlmann, Helen Buchs, and Simon Clematide. 2022. Fine-grained extraction and classification of skill requirements in German-speaking job ads. In *Proceedings of the Fifth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS)*, pages 14–24, Abu Dhabi, UAE. Association for Computational Linguistics.

Thomas Green, Diana Maynard, and Chenghua Lin. 2022. Development of a benchmark corpus to support entity recognition in job descriptions. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1201–1208, Marseille, France. European Language Resources Association.

Akshay Gugnani and Hemant Misra. 2020. Implicit skills extraction using document embedding and its

- use in job recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(08):13286–13293.
- Shiqiang Guo, Folami Alamudun, and Tracy Hammond. 2016. **Résumatcher: A personalized résumé-job matching system**. *Expert Systems with Applications*, 60:169–182.
- Faizan Javed, Phuong Hoang, Thomas Mahoney, and Matt McNair. 2017. **Large-scale occupational skills normalization for online recruitment**. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(2):4627–4634.
- Ilkka Kivimäki, Alexander Panchenko, Adrien Dessy, Dries Verdegem, Pascal Francq, Hugues Bersini, and Marco Saerens. 2013. **A graph-based approach to skill extraction from text**. In *Proceedings of TextGraphs-8 Graph-based Methods for Natural Language Processing*, pages 79–87, Seattle, Washington, USA. Association for Computational Linguistics.
- Sunil Kumar Kopparapu. 2010. **Automatic extraction of usable information from unstructured resumes to aid search**. In *2010 IEEE International Conference on Progress in Informatics and Computing*, volume 1, pages 99–103.
- VI Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Proceedings of the Soviet physics doklady*.
- Nan Li, Bo Kang, and Tjil De Bie. 2023. **Skillgpt: a restful api service for skill extraction and standardization using a large language model**. In *International Workshop on AI for Human Resources and Public Employment Services (AI4HR&PES) at ECML-PKDD*.
- Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. 2018. **doccano: Text annotation tool for human**. Software available from <https://github.com/doccano/doccano>.
- Khanh Nguyen, Mike Zhang, Syrielle Montariol, and Antoine Bosselut. 2024. **Rethinking skill extraction in the job market domain using large language models**. In *Proceedings of the First Workshop on Natural Language Processing for Human Resources (NLP4HR 2024)*, pages 27–42, St. Julian’s, Malta. Association for Computational Linguistics.
- Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.
- Nils Reimers and Iryna Gurevych. 2020. **Making monolingual sentence embeddings multilingual using knowledge distillation**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.
- Elena Senger, Mike Zhang, Rob van der Goot, and Barbara Plank. 2024. **Deep learning-based computational job market analysis: A survey on skill extraction and classification from job postings**. In *Proceedings of the First Workshop on Natural Language Processing for Human Resources (NLP4HR 2024)*, pages 1–15, St. Julian’s, Malta. Association for Computational Linguistics.
- Damian A. Tamburri, Willem-Jan Van Den Heuvel, and Martin Garriga. 2020. **Dataops for societal intelligence: a data pipeline for labor market skills extraction and matching**. In *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*, pages 391–394.
- Laura Vásquez-Rodríguez, Bertrand Audrin, Samuel Michel, Samuele Galli, Julneth Rogenhofer, Jacopo Negro Cusa, and Lonneke Van Der Plas. 2024. **Hardware-effective approaches for skill extraction in job offers and resumes**. In *RecSys in HR’24: The 4th Workshop on Recommender Systems for Human Resources, in conjunction with the 18th ACM Conference on Recommender Systems, October 14–18, 2024, Bari, Italy*. CEUR Workshop Proceedings.
- Laura Vásquez-Rodríguez, Bertrand Audrin, Samuel Michel, Samuele Galli, Julneth Rogenhofer, Jacopo Negro Cusa, and Lonneke van der Plas. 2024. **Hardware-effective approaches for skill extraction in job offers and resumes**. In *Proceedings of the 4th Workshop on Recommender Systems for Human Resources (RecSys-in-HR 2024) co-located with the 18th ACM Conference on Recommender Systems (RecSys 2024)*, volume 3788 of *CEUR Workshop Proceedings*, pages 1–12, Bari, Italy. CEUR-WS.org.
- Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. **Multilingual universal sentence encoder for semantic retrieval**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.
- Mike Zhang, Kristian Jensen, Sif Sonniks, and Barbara Plank. 2022a. **SkillSpan: Hard and soft skill extraction from English job postings**. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4962–4984, Seattle, United States. Association for Computational Linguistics.
- Mike Zhang, Kristian Nørgaard Jensen, and Barbara Plank. 2022b. **Kompetencer: Fine-grained skill classification in Danish job postings via distant supervision and transfer learning**. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 436–447, Marseille, France. European Language Resources Association.
- Mike Zhang, Kristian Nørgaard Jensen, Rob van der Goot, and Barbara Plank. 2022c. **Skill extraction**

from job postings using weak supervision. In *RecSys in HR'22: The 2nd Workshop on Recommender Systems for Human Resources, in conjunction with the 16th ACM Conference on Recommender Systems, September 18–23, 2022, Seattle, USA*. CEUR Workshop Proceedings.

Mike Zhang, Rob van der Goot, and Barbara Plank. 2023. [ESCOXLM-R: Multilingual taxonomy-driven pre-training for the job market domain](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11871–11890, Toronto, Canada. Association for Computational Linguistics.

Meng Zhao, Faizan Javed, Ferosh Jacob, and Matt McNair. 2015. [Skill: A system for skill identification and normalization](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(2):4012–4017.

Acknowledgments

We would like to thank Arca24 HR specialists for their support with annotations in this project. Finally, we gratefully acknowledge the support from Innosuisse (grant 104.069 IP-ICT).

A Appendix

A.1 Research on NLP applications

We reflect on the challenges of translating NLP research into industry applications. In our study, we carried out experiments within the framework of an industrial project aimed at developing improved algorithms for candidate selection. In this context, we faced a clear trade-off between the complexity of the NLP methods and the state of the art of research and the pragmatic requirements of the industry regarding a deployable solution within existing systems over a limited timeframe. The project has now been completed, and our industrial partner has successfully integrated the algorithms. This outcome demonstrates not only the practical readiness of the research but also underlines a key lesson: the importance of a bottom-up development process, where all industry parameters need to be accounted for beyond raw performance taking into account the available infrastructure, costs and time constraints, and explainability for the users needs, costs, and performance are defined collaboratively to align with the company’s roadmap, while at the same time enabling innovation through research that is explainable and fair.

A.2 Methodology

In this section, we discuss the freely available data related the Green dataset (Section A.2.1). Finally,

we present our data preprocessing steps (Section A.3.2), the description of our rule-based and semantic models (Section A.2.1), including the training details for the supervised system (Section A.3.3).

A.2.1 Publicly available data

We present the selected academic dataset (Green) for our experiments:

Green_JOB (Green et al., 2022): This academic dataset is a set of job offers from UK job boards with 8670 sentences for training, 964 for validation, and 335 for testing.¹⁷ The annotations of these jobs were done by crowdsourcing efforts, labeling entities with the following types: *Skills, Knowledge, Occupation, Experience, and Domain*. We selected this dataset as a reference to previous work.

A.3 Experimental Setup

In this section, we extend the implementation details of the paper, including the data processing steps (Section A.3.1), the training parameters (Section A.3.3), and the selected categories for the human evaluation (Section 3.1.1).

A.3.1 Data Preprocessing

We preprocessed the taxonomy to support the rule-based system by precalculating all terms and storing them in a local JSON file. This allows for efficient matching of all possible term variants for a given word (e.g., Software developers → software developers). For the semantic system, we did not change the original terms as the semantic similarity will generalize enough to capture these minor syntactic differences.

Concerning the datasets, we divided the texts presented in Section 2.3, for both job offers and resumes, into 80% for training, 10% for validation, and 10% for testing. In particular, for the supervised system, we split documents into 200 tokens to avoid any issues with the input size of the multilingual models. For the Green dataset, we kept the original distribution of the splits. All systems were evaluated on the same test set and in 6 languages. As for the input and output format for all systems, we relied on the IOB schema (Ramshaw and Marcus, 1999) to clearly understand the boundaries of each entity.

A.3.2 Models

In this section, we explain the implementation details of the rule-based and supervised system.

¹⁷<https://huggingface.co/datasets/jjzha/green>

Rule-based: We followed the implementation of the rule-based system proposed by [Vásquez-Rodríguez et al. \(2024\)](#) with minor enhancements. The rule-based system will search each term from the taxonomies (see Section 2.2) in the text. To maximize the matching of concepts, the taxonomy is normalized using techniques such as lemmatization and stemming. The final list of matching concepts is ranked according to their similarity using semantic similarity and Levenshtein distance ([Levenshtein, 1966](#)), as overlapping of n-grams can happen.

Semantic SBERT: We present the semantic system as a better generalization of the rule-based algorithm. To achieve this, we have improved the system proposed by [Vásquez-Rodríguez et al. \(2024\)](#), replacing the similarity mechanism from Spacy models¹⁸ for SBERT embeddings ([Reimers and Gurevych, 2020](#)). Sentence embedding comparisons are more accurate and showed to be 36X faster than spacy-based methods based on the evaluation of the *Green_JOB* test set. In comparison to the rule-based systems, the semantic model can find concepts that are closely related rather than an exact match. The main benefit is that there is no need for an exhaustive taxonomy, which in that case, degrades the performance of the algorithm as it grows. Finally, for the semantic comparisons, we selected the *distiluse-base-multilingual-cased-v1* model,¹⁹ a multilingual model supporting 15 languages, distilled from the universal multilingual sentence encoder proposed by [Yang et al. \(2020\)](#).

A.3.3 Training Parameters

We conducted experiments using three random seeds and reported the average results across all runs. The selected hyperparameters include a batch size of 16, a learning rate of 5.0×10^{-5} , and a maximum of 10 epochs. For training, jobs were executed on heterogeneous hardware configurations, using either four Intel(R) Xeon(R) Platinum 8468 or AMD EPYC 7742 CPU cores, 16 GB of RAM, and one of two types of GPUs: an NVIDIA V100 with 32 GB of memory or an NVIDIA RTX 3090 with 24 GB of memory. Evaluation was carried out on a separate setup with two AMD EPYC 7742 CPU cores and 32 GB of RAM.

¹⁸<https://spacy.io/models/>

¹⁹<https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1>

Model	Taxonomy	Example
Reference	-	An analytical approach to problem solving excellent team working skills
Rule-based	IN-HOUSE	An analytical approach to problem solving excellent team working skills
Rule-based	ESCO	An analytical approach to problem solving excellent team working skills
Semantic	IN-HOUSE	An analytical approach to problem solving excellent team working skills
Semantic	ESCO	An analytical approach to problem solving excellent team working skills
mBERT	-	An analytical approach to problem-solving excellent team working skills

Table 7: System outputs from the English job offers.

A.4 Future Work

We discuss our future work in terms of the inclusion skills that arise in the job market, the cultural nuances that can affect our work, and possible approaches to avoid redundancy between taxonomies.

Inclusion of new skills: The inclusion of novel, in-domain skills is done by continuously analyzing skills extracted by the supervised systems that are currently missing from the core taxonomy. This approach offers an improved selection method over the traditional, costly, fully manual method performed by specialists and accelerates the process by already suggesting potential skills that later will be curated by humans. Only when a completely new or unfamiliar domain arises, specific re-annotation efforts are necessary to maintain accuracy and performance. The unsupervised detection of new skills and the integration of large LLMs for this purpose remain outside the current scope of our project.

Cultural nuances: Beyond this work, there remain interesting avenues for exploration, particularly concerning the cultural and linguistic aspects of skill expression ([Allen and Vardaman, 2017](#)). The differences in recruiting culture significantly influence resume structure and how strengths are expressed. Investigating these cultural nuances is a promising direction for future research.

Taxonomies alignment: Given the public access to both taxonomies, we could compute semantic similarity between entries to identify redundant occurrences of existing skills. In addition, it would be very useful to paraphrase and standardize the concepts into a more realistic, resume-like style using LLMs, assuming that sufficient resources are available, that document privacy can be preserved, and hallucinations are mitigated.

WORK EXPERIENCE Visual Merchandising / Floor Manager / Sales Assistant 2006 - 2022 Hermes Lugano Image care in boutique Coordinate the product rotation , interr
 •DOMAIN •OCCUPATION •DATE •SOFT_SKILLS

Partecipate in and support the success of store oppening Support the store manager Prepare and conduct briefing Support during fashion showa and corporate events |
 •SOFT_SKILLS •HARD_SKILLS •SOFT_SKILLS
 •HARD_SKILLS

Sales Assistant 2002 - 2006 Ermenegildo Zegna Lugano Customer loyalty Manage the activities of sales Mantaintain effective customer relationships Offer vip custom
 •OCCUPATION •DATE •HARD_SKILLS •HARD_SKILLS •HARD_SKILLS

Sales of made-misure suits SaleAssistant 1996 - 2002 Minimoda Nassabimbi SA Lugano Sales strategy Windows display Maneging product inventory Stock managem
 •DATE •HARD_SKILLS •HARD_SKILLS

Apprentice 1993 - 1996 Bally Lugano Product knowledge Commercial awarnes Use of cash register INTERNAL TRAINING Hermes Suisse SA 2006 - 2022 Sales experien
 •OCCUPATION •HARD_SKILLS •HARD_SKILLS •DATE

merchandising , floor manager Ermenegildo Zegna 2002 - 2006 Customer relation management CRM HIDAJETE VESELI VISUAL MERCHANDISING SKILLS Interior and
 •DATE •HARD_SKILLS •HARD_SKILLS

display arrangement Sales strategy Exellent interpersonal communication skills Seasonal campaigns SOFT SKILLS Passionate for fashion and customer service Focus
 •HARD_SKILLS •SOFT_SKILLS •DOMAIN •HARD_SKILLS

and experience Positive attitude Ability to work indipendently Availability to travel LANGUAGE Native : Italian Native : Albanian Advanced : English B2 Advanced : French
 •LANGUAGE.ADV •LANGUAGE.ADV •LANGUAGE.ADV
 •LANGUAGE.ADV

Figure 2: Example of the annotation effort using the Docanno tool. Data was anonymized to avoid correlations to a real candidate.

RUMLEM: A Dictionary-Based Lemmatizer for Romansh

Dominic P. Fischer Zachary Hopton Jannis Vamvas

Department of Computational Linguistics, University of Zurich

{dominicphilipp.fischer, zacharywilliam.hopton, jannisnikos.vamvas}@uzh.ch

Abstract

Lemmatization – the task of mapping an inflected word form to its dictionary form – is a crucial component of many NLP applications. In this paper, we present RUMLEM, a lemmatizer that covers the five main varieties of Romansh as well as the supra-regional standard variety Rumantsch Grischun. It is based on comprehensive, community-driven morphological databases for Romansh, enabling RUMLEM to cover 77–84% of the words in a typical Romansh text. Since there is a dedicated database for each Romansh variety, an additional application of RUMLEM is variety-aware language classification. Evaluation on 30'000 Romansh texts of varying lengths shows that RUMLEM correctly identifies the variety in 95% of cases. In addition, a proof of concept demonstrates the feasibility of Romansh vs. non-Romansh language classification based on the lemmatizer.

 <https://github.com/ZurichNLP/rumlem>

1 Introduction

Romansh is a minority Romance language spoken by approximately 40'000–60'000 speakers in several Alpine valleys of Switzerland. It comprises five regional varieties, or *idioms* (Sursilvan, Sutsilvan, Surmiran, Puter, and Vallader), as well as Rumantsch Grischun (RG), the supra-regional standard variety of Romansh. The varieties differ to such an extent that mutual intelligibility is often limited, highlighting the need for variety-specific NLP-tools.

The proposed lemmatizer RUMLEM, given Romansh text, uses morphological databases to (1) infer the possible lemmas of word forms, (2) identify morphological features of word forms, and (3) identify the likely Romansh variety of the input (cf. Figure 1). Together with Model et al. (2026), our system is among the very few systems to reliably perform such a classification; with the additional

lavuraiva → **Lemma:** ‘lavurar’ or ‘lavurer’
Morph: Impf. Tense, 1./3. Sg.
Idiom: Vallader or Puter

Figure 1: The main functionalities of RUMLEM.

benefit that it can be used to distinguish between Romansh and non-Romansh text. It thus provides a transparent complement to machine learning approaches for language identification and a core component to variety-aware Romansh NLP-tools.

Our approach builds on existing, maintained, and community-driven dictionary data (cf. Table 1), which we process into 725'005 unique word forms mappable to 178'467 lemmas (cf. Table 2).

These data allow our lemmatizer to cover around 80% of a typical Romansh text (cf. Section 4.1) and to identify the variety correctly in 95% of cases (cf. Table 4). Language identification experiments (cf. Figure 2) show that a threshold of ca. 0.6 (i.e., 60% of words recognised as a particular Romansh variety) serves to distinguish Romansh texts from the most closely related Romance languages.

2 Dictionary Resources for Romansh

2.1 Bilingual Dictionaries

Pledari Grond,¹ the dictionary underlying RUMLEM, covers all six Romansh varieties, with translations provided in German (DE), as well as, in part, additional annotations (cf. Table 1). Users may report potentially erroneous German–Romansh pairs and suggest alternative translations.

The dictionaries for Rumantsch Grischun, Surmiran, Sursilvan and Sutsilvan are openly licensed (© Lia Rumantscha 1980–2025). The Vallader and Puter dictionaries were kindly provided by Uniun dals Grischs for use solely as part of this lemmatizer (© Uniun dals Grischs. All rights reserved).

¹<https://pledarigrond.ch/>

	Unique Entries	Single Words	DE Translations	POS Tags	Gender	Infl. Verbs
Sursilvan	147,977	93,211	147,971	67,201	72,460	5,031
Sutsilvan	58,584	39,191	58,581	42,817	30,876	3,021
Surmiran	74,986	44,365	74,986	39,224	36,956	2,947
Puter	89,908	32,084	89,807	13,712	36,918	3,383
Vallader	106,690	35,322	106,438	10,854	48,435	3,779
RG	249,169	94,291	249,165	98,046	161,942	3,867
Total	727,314	338,464	726,948	271,854	387,587	22,028

Table 1: Description of the *Pledari Grond* dictionary for each Romansh idiom as well as Rumantsch Grischun. A single ‘l’ means ‘thereof’: Unique entries, *thereof* X Single Words, German Translations, etc. *Infl. Verbs* refers to the number of unique verbs for which inflected forms are provided.

2.2 Spellchecking

Pledari Grond also provides a Romansh spell-checking system based on HUNSPELL.² With the focus lying primarily on orthographic conventions rather than on inflectional or derivational morphology across the different varieties, we do not use the spell-checker for inflectional processing. We restrict its use to providing a fallback vocabulary, together with the *Mediomatix* corpus (Hopton et al., 2026) and the Rumantsch Grischun newspaper *La Quotidiana*.³ Our lemmatizer uses said vocabulary to check non-lemmatizable words against a variety’s lexicon.

3 Software Design

3.1 Preprocessing of Dictionary Data

Lemma mappings constitute the central building block of RUMLEM, making the transformation of the available dictionary data into this format a key step. We treat the morphologically rich and frequently annotated parts of speech (POS) nouns, verbs, and adjectives separately. Other POS (where present) or entries lacking POS tags were treated jointly. Where possible, we used conservative, rule-based heuristics to assign missing POS tags – for example, treating entries whose German translation begins with a capital letter as nouns.

The entries in the Pledari Grond dictionaries exhibit a wide range of structural patterns, with e.g. approximately 200 distinct patterns each for nouns and adjectives. Pattern recognition distinguishes between single (w) and multiple words (w+), punctuation symbols, as well as special morphological tags (marked below as MT; e.g., m., f., sg., pl.).

²<https://hunspell.github.io/>

³<https://huggingface.co/datasets/ZurichNLP/quotidiana>

Two of the most frequent noun patterns and a less frequent one serve as illustration (# occurrences):

w (208,000): armaziun; f; Bewaffnung

w, w (5067): admiratur, admiratura; m/f; Bewunderer(in)

w (w, MT); w (w, MT) (93): arrestà (arrestats, pl); arrestada (arrestadas, pl); m/f; Gefangene

Based on such recurring patterns, informed decisions could be made about how to process the data; for example, that a **w, w** entry such as the one above should yield two separate entries, *admiratur; m; Bewunderer* and *admiratura; f; Bewundererin*. To ensure clean and consistent processing, we automatically generated test skeletons for each distinct pattern occurring more than ten times in each of the four POS categories (N, V, ADJ and other), and manually annotated the corresponding gold-standard outputs. 200 such tests, covering 99.9% of the input data, contribute to high data quality (cf. Appendix A). Table 2 presents the resulting data available to the lemmatizer.

3.2 Variety Identification Process

RUMLEM takes a text and an optional variety. If none is given, it predicts the most likely variety based on the input text. More specifically, the text is tokenized using an adapted version⁴ of the Italian Moses tokenizer (Koehn et al., 2007); then, for each variety, the system counts lemmatizable tokens and tokens found in the variety-specific vocabulary and divides this count by the total number of tokens.

⁴The adaptations consist of regex-based preprocessing and protected token patterns designed to correctly tokenize apostrophe-based contractions in each Romansh variety.

	Vocab	Mapped Forms	Lemmas	Noun	Adj	Verb	Other
Sursilvan	223,826	222,860	36,505	23,206	4,977	5,858	2,464
Sutsilvan	129,519	87,902	19,326	12,467	2,671	3,033	1,155
Surmiran	149,078	84,481	22,838	15,145	3,107	3,204	1,382
Puter	180,361	107,758	26,201	15,534	3,122	3,102	4,443
Vallader	165,354	109,090	30,479	19,841	4,821	3,625	2,192
RG	180,690	112,914	43,118	31,200	6,099	4,049	1,770
Total	1,028,828	725,005	178,467	117,393	24,797	22,871	13,406

Table 2: RUMLEM’s data coverage, single words only. *Mapped Forms* describes the amount of entries linked to a lemma. A single ‘l’ means ‘thereof’: Vocab, *thereof* X Mapped Forms, *thereof* X Lemmas, etc.

3.3 Lemmatization Process

Consider the sentence *La vuolp d’eira darcheu üna jada fomantada* (“The fox was once again hungry”). RUMLEM identifies the variety correctly as Vallader, and returns per-token analyses, as shown for *fomantada* in Table 3: assuming Vallader, it may correspond to the feminine form of the adjective *fomantà*, a feminine noun *fomantada*, or the past participle of the verb *fomantar*.

3.3.1 Unknown Word Forms

To try and map unknown word forms to known lemmas, the lematizer’s edit tree component may be invoked. Adapting a system for unsupervised morphological paradigm completion (Jin et al., 2020; Kann et al., 2020), this component learns frequent inflectional patterns and their morphological tags for each variety and POS category. For example, masculine adjectives receive an -a when turned feminine, and an -s when pluralized, meaning the edit trees store two adjective-specific lemmatization paths: drop -a (feminine) or drop -s (plural). At inference, upon encountering an unknown word, paths across all POS are searched and transformations applied if suffixes match. The resulting potential lemmas are checked against existing lemmas within the same POS-tag. Matches are collected,

and the candidate with the shortest edit distance to the original word form is selected.

4 Evaluation

4.1 Lemmatization Coverage

The data used for this task as well as for variety identification consisted of 3000–7000 texts for each variety, covering a range of input lengths. Shorter texts consist of validated speech transcripts from Romansh broadcasts by Radiotelevisiun Svizra Rumantscha (RTR). Texts longer than 300 tokens are taken from a set of children’s stories called *Babulins*, which exist in each Romansh variety. Note that the distributions are not even and differ between varieties; we report them in Appendix C.

We define lemmatization coverage as the percentage of word forms in a Romansh text for which our lemmatizer returns an analysis (i.e., excluding forms in the fallback vocabulary). Removing the high-frequency punctuation symbols “., !?; :”, we find that RUMLEM lemmatizes around 80% of all word forms, with variety-specific coverages ranging between 77% and 84%. The edit trees component manages to cover another 5% of words, raising the coverage to around 85%; as it is designed conservatively (cf. Section 3.3.1) and learns

Variety	Form + features	Gloss
RM-SURMIRAN	fomanto [PoS=ADJ; Gender=FEM; Number=SG]	hungrig
RM-SURMIRAN	fomantar [PoS=V; VerbForm=PTCP; Tense=PST; Gender=FEM; Number=SG]	aushungern
RM-VALLADER	fomantà [PoS=ADJ; Gender=FEM; Number=SG]	ausgehungert
RM-VALLADER	fomantà [PoS=ADJ; Gender=FEM; Number=SG]	hungrig
RM-VALLADER	fomantada [PoS=N; Gender=FEM; Number=SG]	Ausgehungerte
RM-VALLADER	fomantada [PoS=N; Gender=FEM; Number=SG]	Hungrige
RM-VALLADER	fomantar [PoS=V; VerbForm=PTCP; Tense=PST; Gender=FEM; Number=SG]	jn aushungern

Table 3: RUMLEM’s output given the token ‘fomantada’, with potential lemmas and morphological annotations of the form itself returned, as well as the German translation (variants).

from high-quality dictionary data, the risk of erroneous lemma mappings remains limited. We report detailed coverage scores in Appendix B.

4.2 Dictionary-based Variety Identification

We also evaluated the performance of our lemmatizer in terms of variety classification accuracy. The results, summarized in Table 4, show that RUMLEM accurately recognizes the variety of the vast majority of Romansh texts, especially longer texts. We note that the text genre might, in addition to text length, play a role in classification accuracy.

Length	2–10	10–50	50–300	300–800	800+	All
Sursilvan	0.85	0.85	0.87	1.00	1.00	0.86
Sutsilvan	1.0	0.99	1.0	1.00	1.00	1.00
Surmiran	0.92	0.94	0.99	1.00	1.00	0.95
Puter	0.97	0.98	0.99	1.00	1.00	0.98
Vallader	0.94	0.91	0.93	1.00	1.00	0.92
RG	0.89	1.00	1.00	1.00	1.00	1.00
All	0.94	0.94	0.95	1.00	1.00	

Table 4: Classification accuracy by Rumantsch variety across length buckets (number of tokens).

These scores are comparable to what Model et al. (2026) reports on balanced in-domain data, as well as unbalanced in-domain data with longer samples (avg. ca. 530 tokens). On shorter unbalanced in-domain data (avg. ca. 85 tokens) and out-of-domain data, their SVM classifier struggles, with F1 scores dropping to ca. 0.8 and 0.7, respectively.

4.3 Dictionary-based Language Identification

We selected about 5000 texts from Fineweb⁵ in Romansh itself as well as the four Romance languages French, Italian, Catalan and Romanian, as these languages are typologically close to Romansh and therefore most likely to exhibit overlapping dictionary forms (cf. Table D). We record the “winning” scores, i.e., the highest score assigned to a text across the Romansh varieties. Figure 2 shows the Romansh score distributions in turquoise and the non-Romansh ones in rust color for three different setups: as-is, using the sets of words, and removing Romance-language (FR, IT, CA, RO) stopwords.

Figure 2 and App. D.2 show that, using the sets of words, a separating threshold can be found for all three tested buckets. Perfect separation was achieved apart from bucket 50–300; however, each

⁵The data is made up of webpages crawled by Common-Crawl between 2013 and 2024.

of the misclassified samples was highly noisy, containing a mix of languages (cf. App. D.3). Further manual inspection revealed the presence of many similar samples on the lower end of the Romansh distribution; the ideal threshold may thus lie higher.

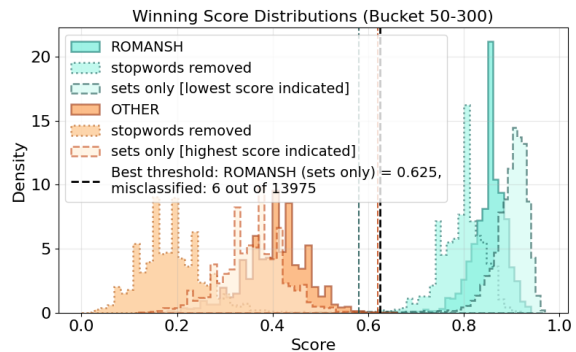


Figure 2: Distributions of Romansh (turquoise) and other Romance languages (rust) according to the highest Romansh variety score assigned to each text sample.

These results suggest that a straightforward Romansh language identification system could be built using RUMLEM. Since the distributions are well-separated, a small validation set would suffice to determine the optimal classification threshold.

5 Related Work

NLP for Romansh Romansh and its varieties are not yet covered by popular NLP tools and resources such as SpaCy, Universal Dependencies (de Marnaffe et al., 2021), and UniMorph (Batsuren et al., 2022), motivating the development of dedicated tools. Recent years have nonetheless seen progress in other areas of Romansh NLP, including contextualized token embeddings and named entity recognition (Vamvas et al., 2023), word alignment (Dolev, 2023), and machine-learning-based variety identification (Model et al., 2026). This paper extends this line of work by providing a dictionary-based system for (context-agnostic) lemmatization and morphosyntactic analysis that, as we show, can also serve as a basis for variety and language identification.

Dictionary-based Lemmatization Our work is situated in the tradition of rule- and lexicon-based computational morphology (Koskenniemi, 1984; Schmid et al., 2004). While more recent, neural approaches to lemmatization and morphosyntactic analysis can take into account the context of word forms and even generalize to unseen forms (Straka et al., 2016; McCarthy et al., 2019; Qi et al., 2020),

they require supervised data, typically in the form of treebanks. In the absence of such treebanks, dictionary-based lemmatization is a viable alternative when a comprehensive dictionary is available for a language. Dictionary-based lemmatizers have been proposed for, among others, German (based on Wiktionary; [Liebeck and Conrad, 2015](#)), Middle English ([Karimov et al., 2016](#)), Latin ([Passarotti et al., 2017](#)), and Somali ([Mohamed and Mohamed, 2023](#)). Our system builds on six large-scale, highly consistent dictionaries for the Romansh varieties that include inflection tables and German translations, enabling relatively high word coverage and a rich feature set.

6 Conclusion

We presented RUMLEM, a dictionary-based lemmatizer covering all six Romansh varieties. Beyond lemmatizing around 80% of a given Romansh text, RUMLEM reliably identifies Romansh varieties – averaging 95% accuracy across varieties and text lengths – and can be used to distinguish Romansh even from its most closely related Romance languages. RUMLEM’s transparent design makes it a useful complement to machine learning approaches for Romansh NLP.

Limitations

RUMLEM’s performance is inherently bounded by its dictionary coverage and quality: words absent from Pledari Grond cannot be lemmatized, and annotation errors will be propagated as-is. Since the lemmatizer cannot account for context, ambiguous forms may receive multiple analyses without disambiguation. Future work could explore the use of statistical or neural approaches to make it more context-aware.

Finally, we note that while the software of RUMLEM itself is released open-source including the postprocessed dictionary data, the Vallader and Puter dictionaries are released without an open-source license, and use of these dictionaries for research beyond RUMLEM will require written permission from the copyright holders.

Acknowledgments

We thank the Swiss Federal Office of Culture, the Lia Rumantscha, and the Uniun dals Grischs for their support, the participants of the digidi 2025 workshop for fruitful discussions, and Sina Ahmadi for helpful feedback.

References

- Khuyagbaatar Batsuren, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, and 76 others. 2022. [UniMorph 4.0: Universal Morphology](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France. European Language Resources Association.
- Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. [Universal Dependencies](#). *Computational Linguistics*, 47(2):255–308.
- Eyal Liron Dolev. 2023. [Does mBERT understand Romansh? evaluating word embeddings using word alignment](#). In *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 41–53, Neuchâtel, Switzerland. Association for Computational Linguistics.
- Zachary Hopton, Jannis Vamvas, Andrin Büchler, Anna Rutkiewicz, Rico Cathomas, and Rico Sennrich. 2026. [The mediomatix corpus: Parallel data for Romansh language varieties via comparable school-books](#). In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 290–306, Rabat, Morocco. Association for Computational Linguistics.
- Huiming Jin, Liwei Cai, Yihui Peng, Chen Xia, Arya McCarthy, and Katharina Kann. 2020. [Unsupervised morphological paradigm completion](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6696–6707, Online. Association for Computational Linguistics.
- Katharina Kann, Arya D. McCarthy, Garrett Nicolai, and Mans Hulden. 2020. [The SIGMORPHON 2020 shared task on unsupervised morphological paradigm completion](#). In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 51–62, Online. Association for Computational Linguistics.
- Raoul Karimov, Maria Samkova, Svetlana Nikitina, and Andrei Akinin. 2016. [Using a hybrid algorithm for lemmatization of a diachronic corpus](#). In *Proceedings of the Workshop on Computational Linguistics and Language Science*, volume 1886 of *CEUR Workshop Proceedings*, pages 1–8, Moscow, Russia. CEUR-WS.org.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. [Moses: Open source toolkit for statistical machine translation](#). In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion*

- Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.
- Kimmo Koskenniemi. 1984. [A general computational model for word-form recognition and production](#). In *10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pages 178–181, Stanford, California, USA. Association for Computational Linguistics.
- Matthias Liebeck and Stefan Conrad. 2015. [IWNLP: Inverse Wiktionary for natural language processing](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 414–418, Beijing, China. Association for Computational Linguistics.
- Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sabrina J. Mielke, Jeffrey Heinz, Ryan Cotterell, and Mans Hulden. 2019. [The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection](#). In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–244, Florence, Italy. Association for Computational Linguistics.
- Charlotte Model, Sina Ahmadi, and Jannis Vamvas. 2026. Robust language identification for Romansh varieties. In *Proceedings of the 11th edition of the Swiss Text Analytics Conference*, Zurich, Switzerland. Association for Computational Linguistics.
- Shafie Abdi Mohamed and Muhidin A. Mohamed. 2023. [Lexicon and rule-based word lemmatization approach for the Somali language](#). In *4th Workshop on African Natural Language Processing*.
- Marco Passarotti, Marco Budassi, Eleonora Litta, and Paolo Ruffolo. 2017. [The lemlat 3.0 package for morphological analysis of Latin](#). In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, pages 24–31, Gothenburg. Linköping University Electronic Press.
- Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. [Stanza: A python natural language processing toolkit for many human languages](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.
- Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. [SMOR: A German computational morphology covering derivation, composition and inflection](#). In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).
- Milan Straka, Jan Hajič, and Jana Straková. 2016. [UD-Pipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA).
- Jannis Vamvas, Johannes Graën, and Rico Sennrich. 2023. [SwissBERT: The multilingual language model for Switzerland](#). In *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 54–69, Neuchatel, Switzerland. Association for Computational Linguistics.

A Preprocessing Tests

The example below illustrates a preprocessing test case. The first line shows the raw dictionary data, while the lines following '>>>' show the validated format – i.e., how the data is processed before being fed into the lemmatizer.”

```
'antalg(iant)evel, antalg(iant)evla'; adj
>>> antalgevel:
        antalgevel; ADJ;MASC;SG
        antalgevla; ADJ;FEM;SG
        antalgiantevel:
        antalgiantevel: ADJ;MASC;SG
        antalgiantevla: ADJ;FEM;SG
```

B Lemmatization Coverage

Table 5 shows coverage values across texts of varying lengths and varieties. The text data used are the same as in the variety identification experiments – shorter texts come from Fineweb, longer ones from Babulins (cf. Section 4.1). Overall, coverage does not vary significantly with text length or genre, except for fragmentary texts, where individual missing word forms have a greater impact.

Of the approximately 40,000 missing (i.e., non-lemmatizable) tokens across all varieties, around 11,000 are proper nouns or German nouns. Another 4,000 consist of tokens containing numbers, special tokens, or characters such as dashes and hyphens. Notable cases include contractions that are absent from the Pledari Grond dictionaries. Out of the remaining 25’000 – mostly Romansh – words, around 5000 were not lemmatized due to the respective texts being misclassified, the remaining 20’000 are indeed absent from the lemmatizer. The edit trees component cuts this number down to about 13’500 genuinely missing word forms.

Length	2–10	10–50	50–300	300–800	800+	All
Surs.	0.76	0.81	0.84	0.83	0.78	0.84
Suts.	0.71	0.77	0.77	0.73	0.73	0.77
Surm.	0.73	0.80	0.84	0.82	0.79	0.82
Puter	0.79	0.83	0.84	0.84	0.81	0.84
Vall.	0.66	0.77	0.80	0.83	0.81	0.80
RG	0.93	0.79	0.79	0.80	0.78	0.79
All	0.75	0.80	0.82	0.81	0.79	

Without edit trees

Length	2–10	10–50	50–300	300–800	800+	All
Surs.	0.79	0.84	0.86	0.83	0.79	0.86
Suts.	0.81	0.84	0.84	0.82	0.80	0.84
Surm.	0.80	0.86	0.90	0.86	0.84	0.88
Puter	0.83	0.88	0.89	0.86	0.83	0.89
Vall.	0.72	0.82	0.85	0.86	0.84	0.84
RG	0.96	0.86	0.86	0.82	0.80	0.86
All	0.80	0.86	0.87	0.84	0.82	

With edit trees

Table 5: Coverage ratios by text length and variety. Values for each variety and bucket are averaged across samples. ‘All’ shows the total number of lemmatizable tokens divided by the total tokens in the bucket/variety.

C Variety ID

C.1 Samples per Variety and Bucket

Table 6 shows the number of samples per Romansh variety and bucket. The same samples were used for both coverage evaluation as well as variety identification.

Length	2–10	10–50	50–300	300–800	800+	Tot
Surs.	68	2647	4173	7	5	6900
Suts.	6	1190	1795	5	7	3003
Surm.	1113	3751	2204	6	6	7080
Puter	660	2783	2468	5	7	5923
Vall.	277	2209	3202	5	7	5700
RG	9	1218	3052	6	6	4291
Tot	2133	13798	16894	34	38	29897

Table 6: #samples by Romansh variety across buckets.

D Language ID

D.1 Samples per Variety and Bucket

Table 7 shows the number of samples per Romance variety and bucket, used for Romansh vs. non-Romansh identification.

D.2 Separating Thresholds per Bucket

Note that the best threshold is defined as, primarily, the one that best separates the data, and, secondarily, the one with the widest margin of separation.

Length	50–300	300–800	800–2000	Tot
French	2517	1551	693	4761
Italian	2493	1620	671	4784
Romanian	2128	1513	922	4563
Catalan	2595	1575	593	4763
Romansh	4242	661	75	4978
Tot	13975	6920	2954	23849

Table 7: #samples by Romance language across buckets.

In both buckets 300–800 and 800–2000, all methods resulted in perfect separation. However, using the sets of words provided the widest margin, indicating that using sets results in the best separation across the different buckets.

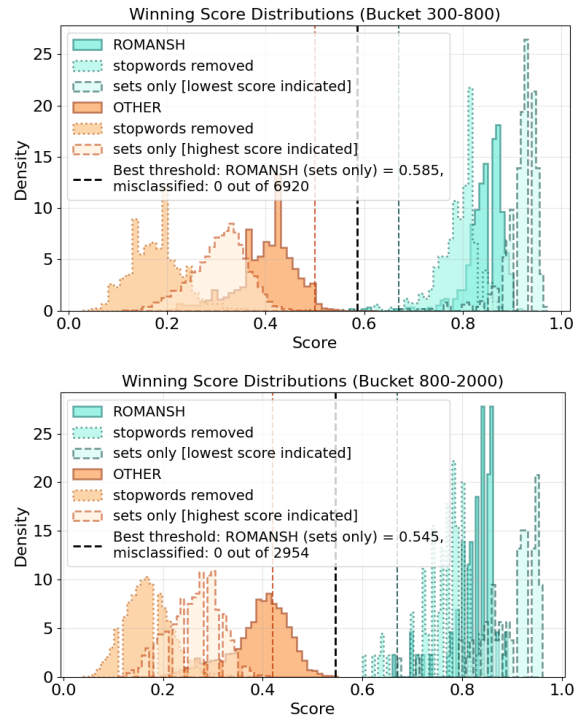


Figure 3: Winning variety score distributions for Romansh (turquoise) and other Romance languages (rust), for token length buckets 300–800 (top) and 800–2000 (bottom).

This was confirmed when we repeated the experiment with different Romansh data, namely the data from the variety classification task. Here too, reducing the texts to their sets of words before being processed by the lemmatizer produced the best threshold in all three token budget settings.

For all experiment settings, we also calculated the average score distribution across varieties instead of the winning score, which resulted in slightly more misclassifications and less inter-

pretable thresholds. This is due to the non-trivial differences between the Romansh varieties, which should be treated separately instead of being conflated.

Finally, note that the ideal threshold decreasing with text length is due to longer texts widening the gap between the Romansh and non-Romansh text, whereby the threshold is placed in the middle.

D.3 Analysis of Misclassified Samples

The six misclassified data samples from Figure 2 were all highly noisy. Four out of the six contained parallel translations (Romansh in bold, Italian in italic):

- “[...] **II cussagl da scoula as cumpuona da duos commembers e dal suprastant dal decasteri.** [...] Der Schulrat setzt sich aus zwei Mitgliedern zusammen sowie dem für diesen Bereich zuständigen Gemeinderatsmitglied. [...]”
- “Herzlich Willkommen Wir begrüßen Sie herzlich auf der Seite unserer Kirchgemeinde und danken Ihnen für Ihr Interesse an unserem Pfarreileben. [...] **Cordial bainvegni silla pagina dalla pleiv catolica Sevgein/Castrisch/Riein. Nus engraziein a Vus che Vus s’interesseis per nossa pleiv.** [...]”
- “**Gorbatschow und Freund Sbalzs classics sün las cordas L’interpret da Balalaika straordinari da nos temp es il virtuos Prof. Andreij Gorbatschow chi viva a Moskau.** [...] Klassische Saitensprünge Der herausragende Balalaika-Interpret unserer Zeit ist der in Moskau lebende Star-Virtuose Prof. Andreij Gorbatschow. [...]”
- “*Un cordiale benvenuto* – Herzlich willkommen - **Cordial bainvegni** *Associazione Spitex dei Grigioni Siamo l’associazione mantello delle 19 organizzazioni Spitex che operano nel Canton dei Grigioni.* [...] Spitex Verband Graubünden Wir sind der Dachverband der 19 im Kanton Graubünden tätigen Spitex-Organisationen. [...] **Federaziun grischuna da spitex Nus essan l’uniun tetgala da las 19 organisaziuns da spitex activas en il chantun Grischun.** [...]”

The remaining two contained what looked like web-scraping artifacts in German:

- “Foto aus dem Akt-Channel Teilnahme am Forum Fotos verkaufen Mehr Foto-Ordner anlegen? Mehr Fotos speichern? [...] **Igl october vargau havein nus era visitau quei marcacau ed jeu muossel in maletg ord il casti da Schönbrunn. Amicabels salids giu da Glion Glieci**”
- “**Cla Rauch ha orientà davart l’Archiv Cultural d’Engiadina Bassa (fotografia: Benedict Stecher).** [...] Haben Sie noch kein Konto? Registrieren Sie sich hier [...]”

Graph-Augmented LLMs for Swiss MP Ideology Prediction

Yifei Yuan¹, Luis Salamanca¹, Sophia Schlosser², Laurence Brandenberger^{2*}

¹Swiss Data Science Center, ETH Zürich, Zürich, Switzerland

²Department of Political Science, University of Zürich, Zürich, Switzerland
yifei.yuan@sdsc.ethz.ch, laurence.brandenberger@ipz.uzh.ch

Abstract

Approximating the ideological position of Members of Parliament (MPs) is a fundamental task in political science, helping researchers understand legislative behavior, party alignment, and policy preferences. While Large Language Models (LLMs) have shown promising results in estimating MPs ideological stances, there are more actors and elements in the parliamentary system, and relations between them, that could provide a wider and more informative picture. However, due to the complexity of integrating them in the prediction task, these additional elements are generally ignored. In this work, we propose an LLM framework, *PG-RAG*, that implements a retrieval-augmented generation pipeline: it first queries a political knowledge graph (KG) and then integrates the resulting graph-structured information into the context. This allows for capturing both textual semantics and inter-MP relationships, another relevant information source in any parliamentary system. We evaluate the approach on the task of ideology prediction, using data from a Swiss parliamentary dataset. When comparing graph-augmented models against several state-of-the-art baselines, the results demonstrate that incorporating this enriched information, which encodes information about different entities and relations, improves prediction performance. These results help to highlight the value of domain-specific relational information in modeling political behavior.

1 Introduction

A central question in political science is how to infer the ideological positions of Members of Parliament (MPs) from their observable political behavior (Poole and Rosenthal, 1985; Clinton et al., 2004). Scholars have increasingly used text-based methods to estimate ideology scores

from speeches, parliamentary debates, and manifestos, enabling more fine-grained and scalable assessments of MPs’ political positions (Laver et al., 2003; Slapin and Proksch, 2008; Proksch and Slapin, 2010; Lauderdale and Herzog, 2016).

As Large Language Models (LLMs) have demonstrated strong performance across a range of NLP tasks, recent studies have explored their potential for predicting MPs ideological stances from textual sources, leveraging their ability to capture semantic nuances and latent political signals embedded in parliamentary speeches (Liu et al., 2022; Bernardelle et al., 2024). Although these approaches have shown reasonable performance, several limitations persist. First of all, they typically treat MPs as independent text generators and ignore the relational structure in parliamentary systems, such as co-sponsorship networks, committee memberships, or party blocs, despite recent work suggesting that such relational information can substantively enrich MPs’ representations (Russo et al., 2023). Moreover, LLM-based methods often struggle in low-data settings or when long-term dependencies across multiple parliamentary sessions must be considered, highlighting the need for methods that can jointly leverage textual content and structured relational knowledge (Huang et al., 2024).

To address these limitations, we propose the integration of graph-structured information to capture ideological alignment and relational influence. By using graph-augmented LLMs, which combine the language understanding capabilities of LLMs with structured relational knowledge captured in graphs, we investigate whether this additional information can improve prediction accuracy. Specifically, we propose a **Political Graph Retrieval-Augmented Generation (PG-RAG)** framework, where a graph encoding parliamentary relationships – such as co-sponsorship links, committee memberships, party affiliations,

*Corresponding author.

and ideological clusters – is queried to retrieve relevant subgraphs for each MP. To encode the complexity of the retrieved graphs, comprised by nodes and relations connected to certain MPs and related to specific parliamentary aspects, we explored two approaches. First, we leverage the great summarization capabilities of LLMs and prompt the model to first summarize the elements retrieved. Second, we explore the ability of the LLM to understand the retrieved graph by providing it as a raw set of nodes and relations. For each case, the additional context, either the obtained summary or the raw graph, is provided as additional information to a pre-defined prompt, also including some general metadata of the MP.

The goal of our work is to investigate whether graph-structured information indeed improves ideology prediction and which types of relational signals are most informative. To assess this, we compare the described approaches against strong baselines, covering several state-of-the-art LLMs. To provide a more systematic evaluation, we benchmark zero-shot and few-shot LLM setups, as well as models of different size. Our experimental results show that incorporating graph information improves prediction performance, with the effect being particularly pronounced for smaller-scale LLMs. We also observe that LLM-based models struggle when positioning Social Democrats, highlighting directions for future analysis.

2 Related Work

2.1 Ideology Prediction

Ideology prediction aims to infer the political or ideological orientation of individuals, groups, or textual content and has been widely studied in the Political Science and NLP domains. Early work focused on predicting the ideological leanings of political actors, such as legislators or parties, using legislative speeches, manifestos, or voting records (e.g., Cox and Poole, 2002; Bakker et al., 2015; Kraft et al., 2016; Vafa et al., 2020; Patil et al., 2019). Approaches from the Political Science domain relied traditionally on scaling procedures (Poole and Rosenthal, 1985; Slapin and Proksch, 2008; Burnham, 2024). In contrast, some NLP-based approaches relied on traditional machine learning models and linguistic features, including SVMs (Sapiro-Gheiler, 2019) and RNNs (Sinno et al., 2022), to distinguish ideological positions. More recent studies have lever-

aged pre-trained language models to capture richer contextual representations for ideology prediction across different domains and languages; for instance, Liu et al. (2021) pre-train a Transformer-based language generator to minimize ideological bias in generated text. With the emergence of LLMs, researchers have begun examining whether ideological orientations can be inferred directly from generated or summarized content, as well as how biases present in training data may affect model predictions (Liu et al., 2022; Bernardelle et al., 2024; Kim et al., 2025).

2.2 LLMs for Political Tasks

Recent studies have explored the capabilities of LLMs in political analysis. Prior work shows that LLMs can perform tasks such as political stance detection (Li et al., 2021; Wagner et al., 2024; Pangtey et al., 2025), ideology classification (Haroon et al., 2025), policy analysis (Chen et al., 2025), often achieving performance comparable to or surpassing traditional NLP models. Researchers have also examined the extent to which LLMs encode political biases or ideological patterns in their training data, investigating whether model outputs reflect systematic political preferences or framing effects (Zhang, 2025; Kim et al., 2025; Rettenberger et al., 2024). In addition, several studies evaluate LLMs in political reasoning and multimodal settings, including tasks such as policy debate generation (Dzeparoska et al., 2023; Chuang et al., 2025), argument analysis (Li et al., 2025), and political question answering (Santurkar et al., 2023).

3 Our Framework

3.1 Preliminary

Given a dataset D consisting of k MP records, $D = \{(i, p_i, g_i, l_i)\}_{i=1}^k$, where p_i denotes the party of the i -th MP, g_i denotes the corresponding party group, and l_i represents the ideology score, the ideology prediction task aims to learn a function F that maps an specific MP i , and its party and group information (i, p_i, g_i) to the corresponding ideology score l_i .

$$l_i = F(i, p_i, g_i) \quad (1)$$

The information of the MP encoded in i can simply contain personal and demographic data, such as age and education, which can already support the task of ideology detection. However, more

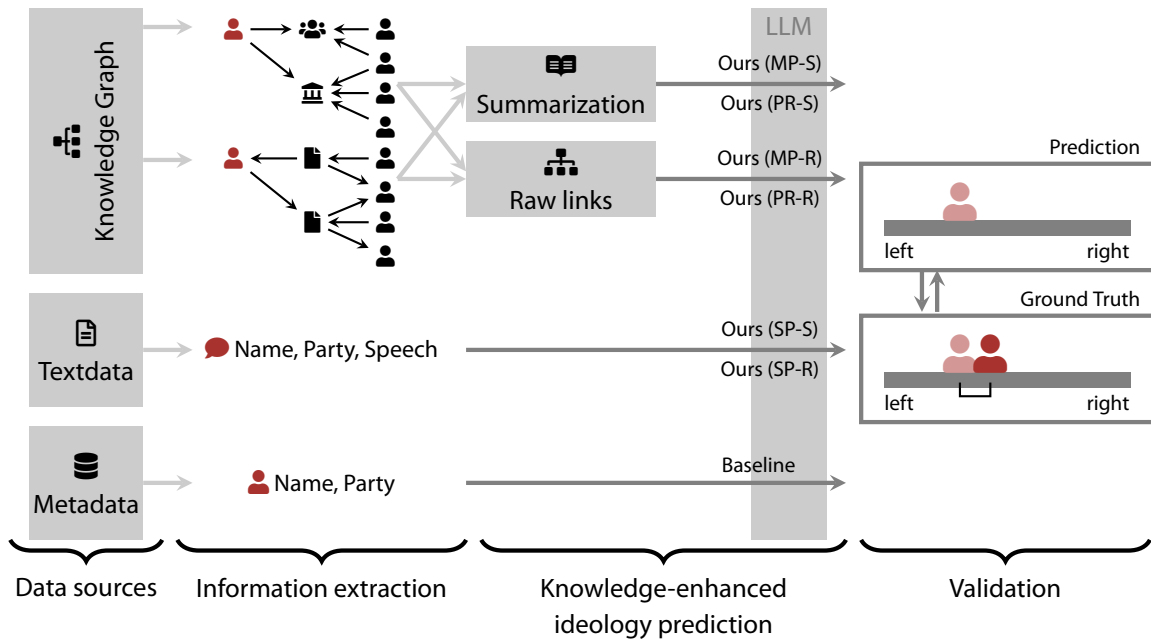


Figure 1: The overall framework of our proposed method *PG-RAG*.

complex information can be additionally provided. In the following sections, we discuss the proposed methodology *PG-RAG* to further enrich *i* with information queried from a Political KG.

3.2 PG-RAG Method

As shown in Figure 1, we propose a RAG-inspired methodology that leverages information extracted from a political knowledge graph (KG) to address the task of ideology prediction. Specifically, the approach uses the information contained in a (1) **political KG**, on which it performs (2) **subgraph extraction**, to finally carry out (3) **knowledge-enhanced ideology prediction**. The following sections detail these steps.

3.2.1 Political KG

The Political KG utilized is built using the information extracted from the Bulletin of the Swiss Parliament, as detailed in (Salamanca et al., 2024). The schema implemented by this KG aims at encoding the policy-making process, from the moment a pursuit text is proposed by a committee, to all the discussions occurring in the parliament chambers related to it. This is captured through entities such as Pursuit and Speech, with relations encoding temporal dependencies. Furthermore, rich metadata, related to the MPs, Parties, etc., is additionally integrated into the graph, providing further context. A subset of this Political KG, which corresponds to the legislative periods

48th to 51st, ranging from 2007 to 2023, is available at (Brandenberger et al., 2024), with further details on the KG structure and its usage.

3.2.2 Subgraph extraction

Due to the large size and complexity of the KG, we decided to define meaningful subgraphs that can be queried independently when generating additional context for the prediction tasks. Each of these subgraphs comprised a subset of entities and the relations connecting them, linked to a specific parliamentary process. Now, given an MP record, we first match it to the corresponding Person node in the KG. Starting from this MP node, the three subgraphs explored are defined according to the following paradigms:

- **Speech-centric (SP)**: We assume that an MP’s speeches and legislative activities provide important signals of their ideological position. Therefore, we collect all speeches linked to the MP Person node, through the relation gives. This serves as an important **baseline** for the other two scenarios, as the retrieved subgraph contains only a single relation and is purely textual.
- **MP-centric (MP)**: In this setting, we extract a subgraph that captures the structural and institutional relationships surrounding the MP (see Figure 6). The extracted subgraph includes entities representing the MPs political

affiliations and institutional roles, such as the Party, Parliamentary Group, Committee, and Chamber to which the MP belongs or is elected. In addition, the subgraph incorporates contextual entities describing administrative and geographic connections, such as the represented Canton.

- **Pursuit-centric (PR):** The pursuit-centric paradigm focuses on the legislative activities initiated or supported by the MP. Specifically, starting from the target MP node, we retrieve all Pursuit entities that are sponsored or co-sponsored by the MP (see Figure 7). These pursuits represent legislative proposals or initiatives that reflect the MP’s policy interests and political priorities.

Above, the subgraphs are ordered by their complexity. First, in the SP subgraph, the only relevant information is the textual data contained in the Speech node. Hence, this approach is similar to recent methods relying on the semantics of textual data. On the contrary, the MP subgraph captures a true graph structure by querying different entity types within the 1-hop vicinity of the Person node. Finally, the PR subgraph increases the complexity by enabling 2-hop extraction, as well as entities connected by different relation types. The specific queries used to parse the graph are provided in the Appendix D.

3.2.3 Knowledge-enhanced ideology prediction

Given the extracted subgraph, we need to generate a suitable representation that can be provided as additional context to the LLMs, aiming at improving the ideology prediction task. We propose the following two approaches to encode the subgraph:

- **Summarization (S):** Motivated by (Zhao et al., 2023), we first serialize the subgraph into natural language sentences. Specifically, each triplet in the knowledge graph is converted into a textual statement that describes the relationship between entities. These serialized statements collectively form a structured textual representation of the MP’s political context, including institutional affiliations, legislative activities, and other relevant relations captured in the subgraph. We use GPT-5 to summarize the subgraph, using the prompts presented in Appendix A.3.

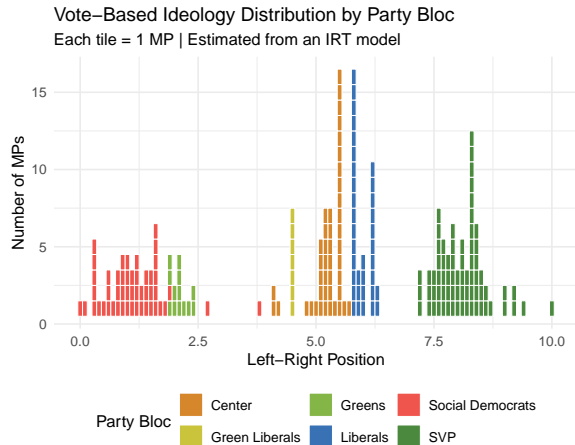


Figure 2: Vote-based ideology scores of Swiss members of the National Council

- **Raw-Graph (R):** The graph is provided as retrieved from Neo4J (Neo4j Core Team, 2024). The JSON formatting encodes the connected nodes as sub-elements, indicating explicitly the relationship type and the different attributes’ values. An example is provided in Appendix C.

Specifically, SP-S provides a summarization of all the queried speeches. In contrast, SP-R presents a subset of them independently. Similarly, (MP/PR)-(S/R) encode truly relational information between different entity types, either serialized into text, or provided as a raw-graph. We report the performance of these 6 different methods accordingly in the following section.

4 Experiments

4.1 Experimental Setup

Dataset. We collect a dataset from the Swiss National Council, the lower chamber of the Swiss Federal Assembly, comprising 225 unique members of parliament (MPs) during the 50th legislative period (2015-2019). The number exceeds the 200 seats ($N = 200$) because some MPs left and were replaced over the four-year period. We compare ideology predictions for these MPs to vote-based scaled estimates. These vote-based estimates stem from a random sample of 1000 votes recorded in the Swiss National Council. All votes in the National Council are recorded electronically and include points of order votes, standard votes on proposals as well as final votes. The

Method	MAE ↓	MSE ↓	RMSE ↓	RC ↑
GM	2.29	7.44	2.73	-
PM	0.30	0.20	0.44	0.97
PBM	0.33	0.23	0.48	0.97
zero-shot				
GPT-5	0.75	1.06	1.03	0.94
Qwen3-8B	1.20	2.32	1.52	0.86
Qwen3-32B	1.11	2.03	1.43	0.86
Apertus-8B	2.01	4.58	2.14	0.77
PG-RAG (MP-S)	0.72	0.77	0.88	0.94
PG-RAG (MP-R)	0.73	0.74	0.86	0.94
few-shot				
GPT-5	0.61	0.61	0.78	0.94
Qwen3-8B	0.88	1.57	1.25	0.90
Qwen3-32B	0.87	1.45	1.21	0.90
Apertus-8B	2.47	7.91	2.81	0.43
PG-RAG (MP-S)	0.58	0.58	0.76	0.94
PG-RAG (MP-R)	0.61	0.60	0.78	0.94

Table 1: Main results comparing all baseline methods to our best performing models, those using the MP-centric subgraph. RC represents Ranking Correlation.

voting data is provided by the Swiss Parliament¹ and incorporated into the DemocraSci KG (Brandenberger et al., 2026). We use a widely-used dimensional-reduction technique, a two-parameter Item Response Theory model, to estimate a one-dimensional model, as per the standard approach applied in Political Science (for a methodological discussion of scaling procedures, see Cox and Poole 2002; Cai et al. 2016; Bailey and Voeten 2018, based on early scaling techniques developed by Poole and Rosenthal 1985). We use the `mirt` package (Chalmers, 2012) in the Statistical Environment R to estimate vote-based ideology scores. Figure 2 shows the stacked distribution of ideology scores (one-dimensional, commonly interpreted as left-right ideological positions).

Evaluation Metrics. We evaluate the ideology prediction performance using the following evaluation metrics: (1) **Regression Metrics:** Since the task involves predicting continuous ideology scores, we measure prediction accuracy using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Regular Mean Squared Error (RMSE). (2) **Ranking Metrics:** We also assess whether the model preserves the relative ordering of ideological positions. For this purpose, we employ Spearman’s rank correlation (ρ). These sets of metrics provide complementary perspectives, helping to reach more insightful and interpretable results.

Compared Methods. We compare our method

¹<https://www.parlament.ch/de/ratsbetrieb/abstimmungen/abstimmungs-datenbank-nr>

Subgraph	Enc.	MAE	MSE	RMSE	RC
zero-shot					
MP	S	0.72	0.77	0.88	0.94
SP	S	0.73	0.78	0.88	0.93
PR	S	0.83	1.03	1.02	0.93
MP	R	0.73	0.74	0.86	0.94
SP	R	0.76	0.78	0.89	0.94
PR	R	0.79	0.89	0.94	0.93
few-shot					
MP	S	0.58	0.58	0.76	0.94
SP	S	0.60	0.70	0.84	0.93
PR	S	0.68	0.83	0.91	0.94
MP	R	0.61	0.60	0.78	0.94
SP	R	0.62	0.59	0.76	0.94
PR	R	0.62	0.60	0.77	0.94

Table 2: Exhaustive results for all variants of PG-RAG, using different subgraphs and context encoding methods. It is important to highlight that the SP method does not provide a graph per-se, but rather a summarization of all speeches (SP-S), or a subset of complete speeches (SP-R).

against several baselines, including: (1) **Naive baselines:** Global Mean (GM), which assigns each MP the overall mean ideology score; Party Mean (PM), which assigns each MP the average ideology score of their party; and Party Bloc Mean (PBM), which assigns each MP the mean ideology score of their party bloc and can be considered an **upper bound** for this category of methods. (2) LLM-based methods: We also compare our method with several state-of-the-art LLMs, including GPT-5 (Singh et al., 2025), the Swiss LLM Apertus (Apertus et al., 2025) (8B version), Qwen3-8B and Qwen3-32B-AWQ (Yang et al., 2025), under both zero-shot and few-shot settings. For the few-shot setup, we randomly select three examples from the vote-based dataset to serve as in-context demonstrations (prompts see Appendix A.1 and A.2). For all of them, we use the default parameter settings.

4.2 Experimental Results

Table 1 reports the performance of different methods on the ideology prediction task. Several observations can be made: among LLM-based baselines, GPT-5 achieves the best zero-shot performance (MAE = 0.75, Rank Corr = 0.94), while Qwen3-32B performs moderately well. In contrast, Apertus-8B shows substantially weaker performance, suggesting that general-purpose LLMs struggle to infer ideology reliably without structured signals. When incorporating graph-derived summary knowledge, our approach improves pre-

diction accuracy. In the zero-shot setting, the MP-centric subgraph reduces MAE to 0.72, while maintaining the same ranking correlation. Furthermore, the MSE is significantly reduced from 1.06 to 0.74 for the PG-RAG (MP-R) case, which demonstrates how our approach is capable of reducing the prediction error even in cases where GPT-5 deviates substantially from the ground truth value. For the few-shot scenario, i.e., when we provide in-context MP examples in the prompt, with their associated metadata and ideology score, the results are more on par. Still, the PG-RAG (MP-S) approach provides some slight improvement. Nevertheless, it is important to highlight that, during our experiments, we noticed a really brittle behavior of the few-shot approach, and adding more examples did not always lead to better results.

Overall, these results demonstrate that injecting structured knowledge distilled from political graphs into LLM prompts substantially enhances ideology prediction, enabling LLMs to better capture ideological ordering among MPs. In particular, the improvements in MSE for the zero-shot scenario allow to demonstrate how the proposed methods can help recover from predictions that deviate substantially from the ground truth value.

In Table 2, we provide results for all the subgraphs queried and used as context. As discussed before, both SP cases resemble previous approaches in which only some textual input is provided as context. In all scenarios, the MP-centric subgraph provides the best results. We believe this is because the MP-centric subgraph provides rich complementary information, such as committees and chambers, while maintaining a moderate context size. On the contrary, the results for the Pursuit-centric subgraph present a degraded performance, likely related to its larger size and complexity, which the LLM still falls short in correctly leveraging. We added more detailed study in Appendix E.

5 Extensive Analysis

5.1 Party-wise Analysis

Figure 3 plots the best LLM-based ideology prediction (MP-S) against the vote-based ideology scores. MPs are colored by party blocs (with representatives of the Christian and Conservative Democratic Parties merged into the Center bloc). Overall, the two ideology scores correlate at a score of $r = 0.963$ (Pearson’s, $t = 53.904$, 223 de-

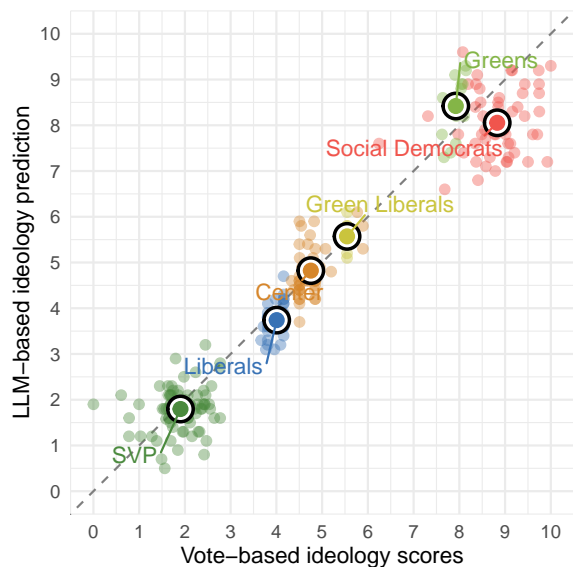


Figure 3: Scatterplot of the best LLM prediction scores (MP-S) vs. the vote-based ideology scores. MPs are colored by party blocs and within-bloc averages are highlighted (black-lined circles).

grees of freedom, $p - value < 0.0001$). The three smaller centric party blocs all show strong coherence in their ideology predictions. The two pole party blocs (dominated by SVP and SP members) show more dispersion around the diagonal, indicating that the LLM sometimes predicts the MPs to be more left or right-leaning than would be expected from their voting behavior. MPs from the Green party bloc (represented by members of the Green party and affiliate communist parties) are generally predicted to be more left by the LLM. This more left-leaning prediction stems from the fact that the Green party is often ideologically positioned to the left of the Social Democrats and is renowned for collaborating with the Swiss communist parties (who inhabit the left-extreme position) (Ladner, 2019, 2012; Hug and Schulz, 2007; Jolly et al., 2022). However, in the 50th legislative period, the Greens have often voted along left ideological positions and deviated towards the center in order to strengthen their alliances to left-leaning centric members. This has brought the MPs from the green party away from the extreme-left position in terms of their voting behavior. The strongest deviation in the LLM prediction stems from the Social Democrats. Here, two factors are at work. First, the LLM judges Social Democrats to be more right-leaning in their ideology than they present in their voting behavior. Second, the LLM

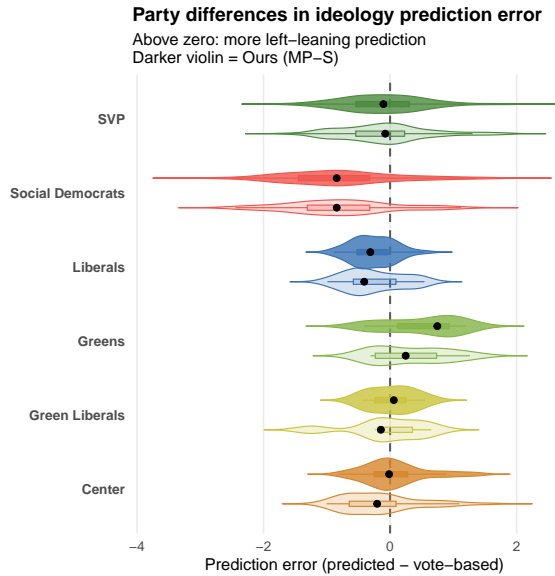


Figure 4: Prediction error by party blocs. The dark violins represent PG-RAG (MP-S) model, the lighter violins show the GPT-5 model.

does not deal with the within-party diversity as well as it does with other party blocs. It is worth investigating whether this dispersion stems from the fact that the LLM is unsure where to place these MPs, or whether the signals from their affiliations are different to their signals from their voting behavior. The latter could possibly stem from increased party discipline in voting, however, then the ideological placements of Social Democrats from these votes would be more unified.

Next, we compare the model predictions from GPT-5 (few-shot) and PG-RAG (MP-S) by party bloc in order to see where the additional graph information has helped improve model predictions. Figure 4 presents prediction errors. For some party blocs, the additional graph information helps contract the predictions around the true (vote-based) values. This is the case for the Liberals, the Green Liberals as well as for the Center party bloc. However, for others, the raw GPT-5 predictions are closer to the vote-based ideologies (Greens, SVP). This should not necessarily be interpreted as a failure of our model, but rather that the inclusion of additional MP information has shifted the ideology predictions. It is well-known that voting behavior is not the only ideological indicator in legislative studies (Snyder Jr and Groseclose, 2000; Rheault and Cochrane, 2020; Barber, 2022) and, as we have indicated, it is based on legislative behavior that is biased in and of itself (e.g., Car-

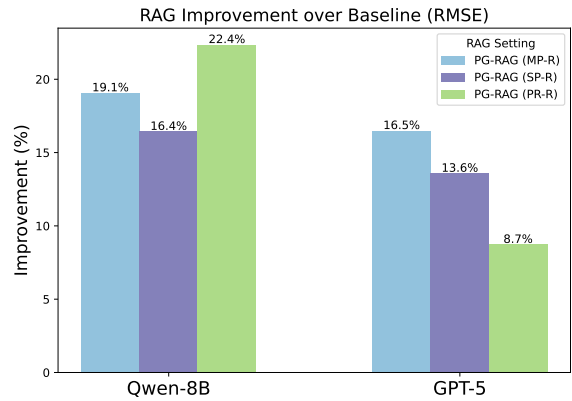


Figure 5: Performance improvements from RAG over the non-RAG baseline for Qwen-8B and GPT-5.

rubba et al., 2006, 2008; Hug, 2010). As such, it would be interesting to study in larger and more varied datasets whether the LLM-based and graph-enhanced ideology predictions reflect a more nuanced ideology placement that encompasses MPs' legislative behavior outside voting.

Figure 4 also shows the difficulties the LLM-based models face when placing Social Democrats. Here, the prediction errors span the largest range in both models, indicating that both the raw GPT-5 as well as the graph-enhanced predictions are generally more right-leaning for Social Democrats than their voting behavior would suggest. The reason for this distortion needs to be explored further: is it based on heterogeneous actions by these MPs that make them difficult to pinpoint, or is it an inherent bias in LLMs to bias Social Democrats more towards the right?

5.2 Backbone Analysis

Figure 5 shows the RMSE improvement from RAG over the non-RAG baseline for Qwen-8B and GPT-5 under the zero-shot scenario, with the context encoding method as raw subgraph (R). From the figure, we observe that the PG-RAG (MP-R) setting yields the most significant improvement for GPT-5, reducing the error rate by 16.5%. This suggests that for large-scale models, providing high-density MP-centric graph context is highly effective. Additionally, Qwen-8B consistently shows higher percentage improvements across all RAG settings (ranging from 16.4% to 22.4%) compared to GPT-5 (8.7% to 16.5%). This indicates that graph-based context provides a more substantial "knowledge boost" to the model with a smaller scale than to the more advanced GPT-

Person	Party (Party Bloc)	Summary of the MP-centric subgraph	GT	GPT-5	PG-RAG
MP-A	LDP (Liberals)	MP-A (b. [YEAR], [CITY]) is a National Council (NR) member for [CANTON]. He belongs to the Liberal Democratic Party (LDP) and sits with the FDP-Liberale parliamentary group (RL), the liberal bloc in Swiss politics. He serves on the [COMMITTEE SEAT]. The LDP/FDP family in Switzerland generally emphasizes a market-oriented economy, individual liberties, and strong support for education, research, and cultural policy.	4.2	4.6	4.2
MP-B	FDP-Liberale (Liberals)	MP-B (b. [YEAR], [CITY]) is a National Council (NR) member representing the canton of [CANTON]. He is a member of FDP.Die Liberalen and sits in the FDP-Liberal parliamentary group (formerly Freisinnig-demokratische Fraktion). The FDP in Switzerland is associated with market-oriented economic policy, competitiveness and individual freedoms. He serves on the [COMMITTEE SEAT A] and the [COMMITTEE SEAT B] committees, focusing on these policy areas.	3.8	2.9	3.6
MP-C	SVP (SVP)	MP-C (born [YEAR] in [CITY]; citizen of [CITIZENSHIP]) represents [CANTON] in the Council of States (SR). He is a member of the Swiss People’s Party (SVP) and its parliamentary group (V). His committee work includes the [COMMITTEE SEAT] and an ad hoc committee [COMMITTEE NUMBER]. The SVP is a right-wing, conservative party emphasizing national sovereignty, restrictive immigration policy, lower taxes, and skepticism toward EU integration.	1.7	2.1	1.8
MP-D	SP (Social Democrats)	MP-C (born [YEAR] in [CITY]) is a National Council member for the canton of [CANTON] from the Sozialdemokratische Partei der Schweiz (SP) and sits in the Sozialdemokratische Fraktion. The SP is a center-left social-democratic party advocating social justice, strong public services, labor rights, and progressive social policy. Naef serves on the [COMMITTEE SEAT A] and the [COMMITTEE SEAT B], indicating a focus on international and legal matters. He is a citizen of the city of [CITIZENSHIP].	9.0	6.5	7.1

Table 3: Examples (de-identified) of the MP ideology prediction from GPT-5 and PG-RAG (MP-S). The summaries are generated through GPT-5 using the information queried from the KG, for the MP-centric subgraph. The prompt used to carry out ideology prediction contained the fully identified information.

5. Interestingly, while the PG-RAG (PR) (Pursuit-based) setting was the strongest for Qwen-8B, it shows the weakest improvements for GPT-5, providing an 8.7% improvement. This suggests that smaller models depend more on external structured guidance, whereas larger models can internally absorb and reason over the same information. Overall, integrating graph knowledge provides a moderate, stable improvement for both models, acting as a reliable middle-ground strategy.

5.3 Case Study

We further demonstrate several examples to understand our model, MP-S case, compared to GPT-5 raw model in Table 3. We find that across the cases, our model’s predictions consistently align more closely with the ground truth than GPT-5. For MP-A and MP-B, both members of liberal parties, our prediction is almost identical to the GT. Similarly, for MP-C, our prediction is nearer to the GT than GPT-5’s. In these cases, our method is better able to leverage the graph and textual context to make accurate predictions. In addition, we observe that for politicians from left-leaning parties, such as MP-D, our model predicts a score of 7.1, which is closer to the ground-truth value of 9.0 compared to GPT5’s prediction of 6.5. This example highlights the challenge of accurately predicting left-leaning MPs, consistent with our find-

ings in Section 5.1. This difficulty arises because left-leaning MPs often have high ideology scores, while LLMs tend to generate more moderate predictions. Our model partially mitigates this difficulty, producing predictions closer to the ground truth for some of these cases. By leveraging textual summaries from the knowledge graph, which capture rich contextual information, such as committee memberships and policy focus, our model enables better predictions.

6 Conclusion

We introduce *PG-RAG*, a RAG-inspired graph-augmented LLM framework for political ideology prediction. Leveraging the data from a Swiss parliamentary knowledge graph, we explore three subgraph scenarios, speech-centric, MP-centric, and pursuit-centric, with two context encoding methods for each, correspondingly. We then compare our method with several LLMs. Our experiments show that with graph-structured relational data, our approach captures the complex web of inter-MP relationships and parliamentary elements that define legislative behavior. In addition, with graph knowledge, the model shows improved understanding of political tendencies across different parties. These results lay the groundwork for future work on extending to additional parties and enhancing performance on left-leaning parties.

Limitations

While our model demonstrates strong overall predictive performance, it still exhibits reduced accuracy when predicting positions of left-leaning parties. This suggests that the model may not fully capture the nuances in the rhetoric or policy preferences characteristic of these groups. Additionally, our current dataset and analysis are limited to a subset of MPs, and extending the model to a broader set of representatives could improve generalizability. However, this expansion is not feasible for Switzerland due to data constraints and the limited availability of annotated parliamentary records.

Acknowledgments

This work is supported by the Swiss National Science Foundation (grant 10.003.190, measuring political success).

References

- Project Apertus, Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, Barna Pasztor, Bettina Messmer, Dhia Garbaya, Eduard Frank urech, Ido Hakimi, Juan García Giraldo, Mete Ismayilzada, Negar Foroutan, Skander Moalla, Tiancheng Chen, Vinko Sabolec, Yixuan Xu, Michael Aerni, and 84 others. 2025. [Apertus: Democratizing open and compliant llms for global language environments](#). *Preprint*, arXiv:2509.14233.
- Michael A Bailey and Erik Voeten. 2018. A two-dimensional analysis of seventy years of united nations voting. *Public Choice*, 176(1):33–55.
- Ryan Bakker, Catherine De Vries, Erica Edwards, Liesbet Hooghe, Seth Jolly, Gary Marks, Jonathan Polk, Jan Rovny, Marco Steenbergen, and Milada Anna Vachudova. 2015. Measuring party positions in europe: The chapel hill expert survey trend file, 1999–2010. *Party Politics*, 21(1):143–152.
- Michael Barber. 2022. Comparing campaign finance and vote-based measures of ideology. *The Journal of Politics*, 84(1):613–619.
- Pietro Bernardelle, Leon Fröhling, Stefano Civelli, Riccardo Lunardi, Kevin Roitero, and Gianluca Demartini. 2024. [Mapping and influencing the political ideology of large language models using synthetic personas](#). *Companion Proceedings of the ACM on Web Conference 2025*.
- Laurence Brandenberger, Julian Minder, Luis Salamanca, Sophia Schlosser, Lilian Gasser, Vincent Jung, Kouros Shariat, Marta Balode, Anna Schmidt-Rohr, Leon Babi, Fernando Perez-Cruz, and Frank Schweitzer. 2024. [DemocraSci - a parliamentary Knowledge Graph \(4 legislative periods\)](#).
- Laurence Brandenberger, Sophia Schlosser, Luis Salamanca, Lilian Gasser, Marta Balode, Julian Minder, Vincent Jung, Yaren Durgun, Leon Babi, Fernando Perez-Cruz, and Frank Schweitzer. 2026. [DemocraSci: A Knowledge Graph on the Swiss parliament](#). *Submitted Manuscript*.
- Michael Burnham. 2024. Semantic scaling: Bayesian ideal point estimates with large language models. *arXiv preprint arXiv:2405.02472*.
- Li Cai, Kilchan Choi, Mark Hansen, and Lauren Harrell. 2016. Item response theory. *Annual Review of Statistics and Its Application*, 3(1):297–321.
- Clifford Carrubba, Matthew Gabel, and Simon Hug. 2008. Legislative voting behavior, seen and unseen: A theory of roll-call vote selection. *Legislative Studies Quarterly*, 33(4):543–572.
- Clifford J Carrubba, Matthew Gabel, Lacey Murrah, Ryan Clough, Elizabeth Montgomery, and Rebecca Schambach. 2006. Off the record: Unrecorded legislative votes, selection bias and roll-call vote analysis. *British Journal of Political Science*, 36(4):691–704.
- R Philip Chalmers. 2012. mirt: A multidimensional item response theory package for the r environment. *Journal of statistical Software*, 48:1–29.
- Yuxin Chen, Peng Tang, Weidong Qiu, and Shujun Li. 2025. [Using llms for automated privacy policy analysis: Prompt engineering, fine-tuning and explainability](#). *ArXiv*, abs/2503.16516.
- Yun-Shiuan Chuang, Ruixuan Tu, Chengtao Dai, Smit Vasani, Binwei Yao, Michael Henry Tessler, Sijia Yang, Dhavan Shah, Robert Hawkins, Junjie Hu, and 1 others. 2025. Debate: A large-scale benchmark for role-playing llm agents in multi-agent, long-form debates. *arXiv preprint arXiv:2510.25110*.
- Joshua Clinton, Simon Jackman, and Douglas Rivers. 2004. The statistical analysis of roll call data. *American Political Science Review*, pages 355–370.
- Gary W Cox and Keith T Poole. 2002. On measuring partisanship in roll-call voting: The us house of representatives, 1877-1999. *American Journal of Political Science*, pages 477–489.
- Kristina Dzevaroska, Jieyu Lin, Ali Tizghadam, and Alberto Leon-Garcia. 2023. Llm-based policy generation for intent-based management of applications. In *2023 19th International Conference on Network and Service Management (CNSM)*, pages 1–7. IEEE.
- Muhammad Haroon, Magdalena Wojcieszak, and Anshuman Chhabra. 2025. ["whose side are you on?" estimating ideology of political and news content using large language models and few-shot demonstration selection](#). In *IJCNLP-AACL*.

- Wenyu Huang, Guanchen Zhou, Mirella Lapata, Pavlos Vougiouklis, Sébastien Montella, and Jeff Z. Pan. 2024. [Prompting large language models with knowledge graphs for question answering involving long-tail facts](#). *Knowl. Based Syst.*, 324:113648.
- Simon Hug. 2010. Selection effects in roll call votes. *British Journal of Political Science*, 40(1):225–235.
- Simon Hug and Tobias Schulz. 2007. LeftRight positions of political parties in Switzerland. *Party Politics*, 13(3):305–330.
- Seth Jolly, Ryan Bakker, Liesbet Hooghe, Gary Marks, Jonathan Polk, Jan Rovny, Marco Steenbergen, and Milada Anna Vachudova. 2022. Chapel hill expert survey trend file, 1999–2019. *Electoral studies*, 75:102420.
- Junsol Kim, James Evans, and Aaron Schein. 2025. [Linear representations of political perspective emerge in large language models](#). *ArXiv*, abs/2503.02080.
- Peter E. Kraft, Hirsh Jain, and Alexander M. Rush. 2016. [An embedding model for predicting roll-call votes](#). In *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*.
- Andreas Ladner. 2012. Switzerland’s green liberal party: a new party model for the environment? *Environmental politics*, 21(3):510–515.
- Andreas Ladner. 2019. Switzerland: The "green" and "alternative parties". In *New Politics In Western Europe*, pages 155–165. Routledge.
- Benjamin E Lauderdale and Alexander Herzog. 2016. Measuring political positions from legislative speech. *Political Analysis*, 24(3):374–394.
- Michael Laver, Kenneth Benoit, and John Garry. 2003. Extracting policy positions from political texts using words as data. *American Political Science Review*, 97(02):311–331.
- Hao Li, Viktor Schlegel, Yizheng Sun, Riza Theresa Batista-Navarro, and Goran Nenadic. 2025. [Large language models in argument mining: A survey](#). *ArXiv*, abs/2506.16383.
- Yingjie Li, Tiberiu Sosea, Aditya Sawant, Ajith Jayaraman Nair, Diana Inkpen, and Cornelia Caragea. 2021. [P-stance: A large dataset for stance detection in political domain](#). In *Findings*.
- Ruibo Liu, Chenyan Jia, Jason Wei, Guangxuan Xu, Lili Wang, and Sorous Vosoughi. 2021. [Mitigating political bias in language models through reinforced calibration](#). *ArXiv*, abs/2104.14795.
- Yujian Liu, Xinliang Frederick Zhang, David Wegsman, Nick Beauchamp, and Lu Wang. 2022. [Politics: Pretraining with same-story article comparison for ideology prediction and stance detection](#). *ArXiv*, abs/2205.00619.
- Neo4j Core Team. 2024. [Neo4j - the worlds leading graph database](#).
- Lata Pangtey, Anukriti Bhatnagar, Shubhi Bansal, Shahid Shafi Dar, and Nagendra Kumar. 2025. [Large language models meet stance detection: A survey of tasks, methods, applications, challenges and future directions](#). *ArXiv*, abs/2505.08464.
- Pallavi Patil, Kriti Myer, Ronak Zala, Arpit Singh, Sheshera Mysore, Andrew McCallum, Adrian Benton, and Amanda Stent. 2019. Roll call vote prediction with knowledge augmented models. In *CoNLL 2019 - 23rd Conference on Computational Natural Language Learning, Proceedings of the Conference*.
- Keith T Poole and Howard Rosenthal. 1985. A spatial model for legislative roll call analysis. *American Journal of Political Science*, 29(2):357–384.
- Sven-Oliver Proksch and Jonathan B Slapin. 2010. Position taking in European Parliament speeches. *British Journal of Political Science*, 40(03):587–611.
- Luca Rettenberger, Markus Reischl, and Mark Schutera. 2024. [Assessing political bias in large language models](#). *Journal of Computational Social Science*, 8.
- Ludovic Rheault and Christopher Cochrane. 2020. Word embeddings for the analysis of ideological placement in parliamentary corpora. *Political Analysis*, 28(1):112–133.
- Giuseppe Russo, Christoph Gote, Laurence Brandenberger, Sophia Schlosser, and Frank Schweitzer. 2023. Helping a friend or supporting a cause? disentangling active and passive cosponsorship decisions in the u.s. congress. *ACL2023, The 61st Annual Meeting of the Association for Computational Linguistics. Toronto, Canada, July 2023*.
- Luis Salamanca, Laurence Brandenberger, Lilian Gasser, Sophia Schlosser, Marta Balode, Vincent Jung, Fernando Perez-Cruz, and Frank Schweitzer. 2024. Processing large-scale archival records: The case of the Swiss parliamentary records. *Swiss Political Science Review*, 30(2):140–153.
- Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cino Lee, Percy Liang, and Tatsunori Hashimoto. 2023. [Whose opinions do language models reflect?](#) *ArXiv*, abs/2303.17548.
- Eitan Sapiro-Gheiler. 2019. [Examining political trustworthiness through text-based measures of ideology](#). In *AAAI Conference on Artificial Intelligence*.
- Aaditya K. Singh, Adam Fry, Adam Perelman, Adam Tart, Adithya Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr A Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 464 others. 2025. [Openai gpt-5 system card](#).

Barea M. Sinno, Bernardo Oviedo, Katherine Atwell, Malihe Alikhani, and Junyi Jessy Li. 2022. [Political ideology and polarization: A multi-dimensional approach](#). In *North American Chapter of the Association for Computational Linguistics*.

Jonathan B Slapin and Sven-Oliver Proksch. 2008. A scaling model for estimating time-series party positions from texts. *American Journal of Political Science*, 52(3):705–722.

James M Snyder Jr and Tim Groseclose. 2000. Estimating party influence in congressional roll-call voting. *American Journal of Political Science*, 44(2):193–211.

Keyon Vafa, Suresh Naidu, and David M Blei. 2020. Text-based ideal points. *arXiv preprint arXiv:2005.04232*.

Stefan Sylvius Wagner, Maike Behrendt, Marc Ziegele, and Stefan Harmeling. 2024. [The power of llm-generated synthetic data for stance detection in online political discussions](#). *ArXiv*, abs/2406.12480.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#).

Tianyi Zhang. 2025. [Probing political ideology in large language models: How latent political representations generalize across tasks](#). In *Conference on Empirical Methods in Natural Language Processing*.

Jianan Zhao, Le Zhuo, Yikang Shen, Meng Qu, Kai Liu, Michael M. Bronstein, Zhaocheng Zhu, and Jian Tang. 2023. [Graphtext: Graph reasoning in text space](#). *ArXiv*, abs/2310.01089.

A LLM prompts

A.1 Zero-shot setting

Predict the ideology score of this Swiss MP (0=Right, 10=Left). Important guidelines:

- Base your judgment primarily on the MP’s background information.
- Do NOT rely only on the party or bloc label.
- MPs within the same party or bloc can have different ideological positions.
- Choose a precise value (e.g., 3.7 or 6.2).

Name: {MP Name}, Party: {MP Party}, Bloc: {MP Party Bloc}
Return ONLY the number.
Score:

A.2 Few-shot setting

You are a political scientist. Below are examples of MPs and their ideology scores (0 = Far-Right, 10 = Far-Left):

Name: {example MP name} | Party: {example MP party} | Party Bloc: {example MP party bloc} | Score: {example MP ideology score}
...

Predict the ideology score of this Swiss MP (0=Right, 10=Left). Important guidelines:

- Base your judgment primarily on the MP’s background information.
- Do NOT rely only on the party or bloc label.
- MPs within the same party or bloc can have different ideological positions.
- Choose a precise value (e.g., 3.7 or 6.2).

Name: {MP Name}, Party: {MP Party}, Bloc: {MP Party Bloc}
Return ONLY the number.
Score:

A.3 Prompt for subgraph summarization

You are a political science expert.

Given structured information about a Member of Parliament (MP), write a concise neutral description of this politician that could help infer their political ideology.

Avoid speculation and keep the description factual.

MP information:
{mp_context}

Write a short summary (around 500 characters) describing the MP’s political background and potential ideological positioning.

B Subgraph Demonstration

B.1 MP-centric

Figure 6 shows the MP-centric subgraph. For the focal MP, we extract the chamber they are elected to, the committees they sit on, the parties and parliamentary groups they belong to, the canton they represent and the city they live in. Whenever relations are time-stamped we extract only those relations that are within the 50th legislative period.

B.2 Pursuit-centric

Figure 7 shows the pursuit-centric subgraph. This two-hop subgraph starts with the focal MP and their links to pursuits. These relations either represent sponsorship (i.e., submitting agent) or co-sponsorship (i.e., supporting agent). The second hop represents links from the pursuit to other cosponsors as well as to committees or parliamentary groups who can act as sponsors themselves.

Model	MP-centric				Speech-centric				Pursuit-centric			
	MAE	MSE	RMSE	RC	MAE	MSE	RMSE	RC	MAE	MSE	RMSE	RC
zero-shot												
PG-RAG (r-10)	0.75	0.81	0.90	0.94	0.80	0.88	0.94	0.94	0.85	0.99	1.00	0.92
PG-RAG (r-50)	0.77	0.83	0.91	0.94	0.77	0.81	0.90	0.94	0.80	0.92	0.96	0.93
PG-RAG (r-100)	0.79	0.85	0.92	0.93	0.78	0.84	0.92	0.94	0.80	0.91	0.95	0.93
few-shot												
PG-RAG (r-10)	0.55	0.55	0.74	0.94	0.61	0.60	0.78	0.94	0.66	0.72	0.85	0.93
PG-RAG (r-50)	0.60	0.61	0.78	0.94	0.60	0.59	0.77	0.94	0.68	0.72	0.85	0.93
PG-RAG (r-100)	0.59	0.59	0.77	0.94	0.61	0.61	0.78	0.93	0.68	0.74	0.86	0.94

Table 4: Performance comparison of ideology prediction under different settings.

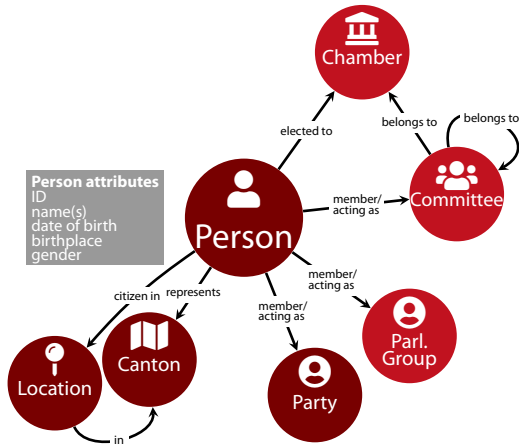


Figure 6: Graph demonstration of the MP-centric subgraph.

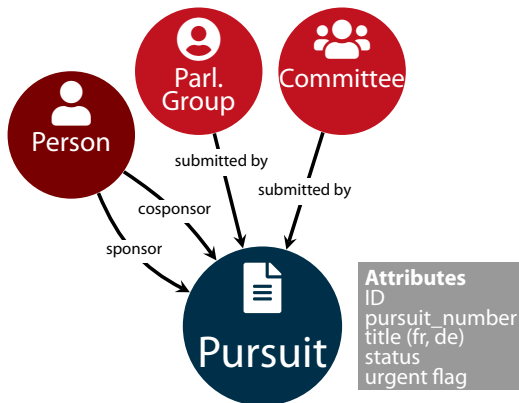


Figure 7: Graph demonstration of the pursuit-centric subgraph.

C Example Subgraph Demonstration

Here's an example of how our extracted Neo4J subgraph looks in the MP-centric scenario:

```
[
  {
    "csv_uid": "74",
    "original_label": null,
    "person_info": {
      "date_birth": "xxx",
      "uid": "xx",
```

```
"gender": "x",
"last_name": "xxx",
"first_name": "xxx"
},
"graph_context": [
  {
    "type": "City",
    "rel": "BORN_IN",
    "properties": {
      "post_code": "4000",
      "name": "Basel"
    }
  },
  {
    "type": "City",
    "rel": "CITIZEN_IN",
    "properties": {
      "post_code": "4000",
      "name": "Basel"
    }
  },
  ...
]
]
```

D Graph-parsing Queries

We provide the queries we used to parse the subgraph under the three scenarios.

D.1 Speech-centric

```
MATCH (p:Person {uid: $uid})
OPTIONAL MATCH (p)-[r]-(neighbor:Speech)
WHERE datetime(neighbor.time_end) >=
datetime("2015-11-30T00:00:00")
AND datetime(neighbor.time_start) <=
datetime("2019-12-01T23:59:59")
RETURN
properties(p) as p_props,
labels(neighbor)[0] as n_label,
properties(neighbor) as n_props,
type(r) as rel_type
```

D.2 MP-centric

```
MATCH (p:Person {uid: $uid})-[r]-(neighbor)
WHERE any(label IN labels(neighbor) WHERE label
IN [
  'Chamber', 'Committee', 'Party', 'Canton',
```

```

    'Location', 'Parliamentary Group'
  ])
  WITH p, neighbor, r,
    labels(neighbor)[0] AS n_label,
    properties(neighbor) AS n_props,
    type(r) AS rel_type
  // Apply constraints for Chamber and Committee
  WHERE (n_label = 'Chamber' AND rel_type =
    'ELECTED_TO'
    AND r.date_election >= date("2015-11-30")
    AND r.date_election <= date("2019-12-01"))
  OR (n_label = 'Committee'
    AND r.date_joining >= date("2015-11-30")
    AND r.date_leaving <= date("2019-12-01"))
  OR (n_label <> 'Chamber' AND n_label <>
    'Committee')
  RETURN DISTINCT properties(p) AS p_props,
    n_label, n_props, rel_type

```

D.3 Pursuit-centric

```

MATCH (p:Person {uid: $uid})

OPTIONAL MATCH (p)-[r1]-(n1:Pursuit)

OPTIONAL MATCH (n1)-[rs:SUBMITTED_TO]->()
WHERE rs.date >= date("2015-11-30")
  AND rs.date <= date("2019-12-01")

OPTIONAL MATCH
(n1)-[r2:SPONSORS|COSPONSORS]-(n2)
WHERE n2 IS NULL OR n2 <> p

RETURN DISTINCT
  properties(p) AS p_props,
  labels(n1)[0] AS n1_label,
  properties(n1) AS n1_props,
  type(r1) AS r1_type,
  labels(n2)[0] AS n2_label,
  properties(n2) AS n2_props,
  type(r2) AS r2_type

```

E Model Variants Comparison

To avoid generating a massive context that could instead confound the model, we restrict the raw graph to N nodes. To ensure a correct coverage of the full subgraph, these are randomly selected among all the sub-elements. Table 4 shows the performance comparison under different setups when we randomly include N nodes with the prompt.

Extracting Article-Level Legal Dependencies from Swiss Federal Law using LLMs

Steven Cho¹ Anna Kiepura² Jessica Lam² Richard Hahnloser²

¹ETH Zurich

²Institute of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland

stecho@student.ethz.ch

{akiepura, jlam, richardh}@ini.ethz.ch

Abstract

Understanding dependencies between legal provisions is essential for analyzing statutory corpora; yet, such relationships are rarely available in machine-readable form. We present a hybrid pipeline for extracting article-level dependencies from Swiss federal legislation on Fedlex, combining deterministic XML preprocessing with large language model (LLM)-based semantic resolution. Additionally, we release three complementary data splits—document-level JSON, structured citation candidates, and LLM-based article assignments—to support downstream legal NLP research. We evaluate our approach on 2,103 SR documents, yielding over 63,000 citation instances. While LLMs are effective at resolving semantically complex references, we observe substantial limitations in structured output reliability: approximately 21% of generated items violate the expected schema, with most errors being unrecoverable. Our findings highlight a key challenge in applying LLMs to structured legal information extraction and provide a new resource for tasks such as legal knowledge graph construction, citation analysis, and benchmarking structured prediction in the legal domain.

1 Introduction

Understanding relationships between legal provisions is essential for maintaining a coherent body of law. Legal texts frequently reference other statutes and articles, forming a dense dependency network across the legal corpus (Kjær, 2000). Extracting these dependencies enables applications such as legal retrieval, impact analysis, and knowledge graph construction. However, this task is challenging because citations appear in diverse linguistic forms and are rarely explicitly encoded.

In Switzerland, the official platform for federal legislation, Fedlex¹, provides access to the *Systematische Rechtsammlung* (SR). Although doc-

¹<https://www.fedlex.admin.ch/>

uments are available in XML, article-level cross-references are not machine-readable and must be reconstructed from fragmented signals in text, footnotes, and hyperlinks.

To address this, we release three complementary data splits: (i) document-level JSON representations of SR documents, (ii) structured citation candidates for LLM processing, and (iii) LLM-based article assignments linking citing and cited provisions.

We evaluate our pipeline on 2,103 SR documents, yielding over 63,000 citation items. While LLMs can resolve semantically complex references, structured output reliability remains a key bottleneck: about 21% of outputs violate the expected schema.

Overall, we make the following contributions:

- We introduce and release three JSON data splits for Swiss federal legislation, covering article-level document content, LLM-ready citation candidates, and LLM-based article assignments.
- We propose a hybrid deterministic-LLM pipeline for extracting article-level legal dependencies from Fedlex.
- We present a large-scale empirical analysis showing that, while LLMs can resolve many semantically ambiguous legal references, structured output reliability remains a key bottleneck.

2 Related Work

Swiss legal corpora and resources. Recent work has introduced datasets for Swiss legal NLP, including multilingual corpora of legislation (Felici, 2025) and judicial decisions (Rolshoven et al., 2025). In particular, CHEU-lex provides a parallel corpus of Swiss and EU legislation, focusing on multilingual alignment and linguistic analysis.

However, these resources do not model article-level dependencies or provide structured representations of cross-references within Fedlex. Earlier work on Swiss legal corpora (Höfler and Piotrowski, 2011) predates Fedlex and relies on outdated collections, limiting its applicability to modern legal data.

Structured information extraction with LLMs. Recent work highlights both the potential and limitations of LLMs for structured output generation. SLOT (Wang et al., 2025) shows that high schema fidelity can be achieved through large-scale synthetic training and fine-tuning, while Structured RAG (Lin et al., 2025) demonstrates that zero-shot prompting over complex inputs leads to unstable and often invalid structured outputs. These findings are particularly relevant for legal documents, which exhibit deep nesting and heterogeneous structures.

Positioning of this work. We extract article-level dependencies from Swiss federal legislation and release a structured dataset derived from Fedlex XML. Unlike prior work, our approach combines deterministic preprocessing with LLM-based resolution without relying on fine-tuning, enabling machine-readable legal dependency structures not available in existing resources.

3 Fedlex Dataset

3.1 Source Data: Swiss Federal Law on Fedlex

Fedlex is the official platform of the Swiss Confederation for publishing federal legislation. It provides access to legally binding documents, including laws, ordinances, and treaties, in multiple formats (XML, HTML, PDF) and languages. In this work, we focus on *Systematische Rechtsammlung* (SR), which organizes federal law into a structured, topic-based collection.

Within SR, legal documents are organized hierarchically by subject domain and document identifier (e.g., SR 141.0). Each document represents a complete legal act and is internally structured into articles, which form the smallest unit of legal normativity (Kelsen, 1967). Because articles contain the operative legal content, recovering relationships at the article level is essential for a meaningful analysis of legal dependencies.

We restrict our study to the *Landesrecht* section of SR in German. We collected 2,103 SR XML documents by issuing a SPARQL query through the Fedlex interface and subsequently using a custom

XML scraper to bulk download the corresponding files.

3.2 Dataset Construction and Data Splits

To enable both citation extraction and downstream NLP applications, we construct and release three complementary JSON data splits derived from the Fedlex corpus:

- **Document-level JSON (SR content).** Each SR XML document is converted into a structured JSON file containing article-level content. This representation preserves document hierarchy, article boundaries, and metadata, making the data directly usable for NLP tasks.
- **LLM input (citation candidates).** We extract citation-relevant context from the XML and represent it as structured JSON items. Each item contains a citing article, a target SR reference, and a local text snippet capturing the citation context. These inputs are designed to provide sufficient semantic and structural information for LLM-based citation disambiguation.
- **LLM output (article assignments).** For each input item, an LLM assigns the corresponding cited article(s) within the target SR document or identifies document-level references when no article is specified. The outputs form a machine-readable representation of article-level legal dependencies.

3.3 Challenges to Article-Level Dependency Extraction

Although Fedlex provides structured XML, article-level dependencies are not explicitly encoded. Relevant information is instead distributed across article text, hyperlinks, and footnotes, requiring a combination of structural parsing and semantic interpretation.

Legal citation patterns are diverse and irregular. References may appear in implicit or non-standard forms, including acronym-based citations, preamble-defined references, intra-document mentions, missing article numbers, or cases where article numbers and SR identifiers are separated. Multiple reference types may also co-occur, creating ambiguity that cannot be resolved with simple heuristics.

¹We release all code and data to facilitate future research: <https://github.com/stevencho24/Extracting-Article-Level-Legal-Dependencies-from-Swiss-Federal-Law-using-LLMs>

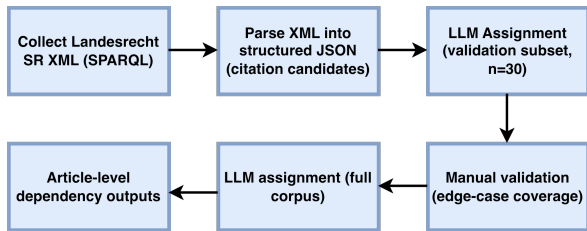


Figure 1: Pipeline for extracting article-level dependencies from Fedlex SR XML. A subset of documents is first processed and manually validated to ensure coverage of citation edge cases, after which the pipeline is applied to the full corpus.

We identify recurring citation edge cases that must be addressed to reconstruct article-level dependencies; a detailed taxonomy is provided in Appendix A.

Given this dataset, the task is to map each *home article* to the correct *cited article(s)* or document. This requires resolving ambiguous citation structures and distinguishing intra- from inter-document references, making the task non-trivial even for modern LLMs.

4 Methodology

We propose a hybrid pipeline that combines deterministic pre-processing of Fedlex XML documents with LLM-based semantic resolution. The pipeline consists of three stages: (i) data collection from Fedlex, (ii) structured extraction of citation candidates, and (iii) LLM-based article assignment.

Given an SR XML document, the goal is to transform raw legal text into structured citation instances and assign each instance to the correct cited article(s) or document. Figure 1 provides an overview of the pipeline.

4.1 Data Collection

We collect SR documents from the *Landesrecht* of Fedlex using its SPARQL interface. The resulting dataset consists of 2,103 XML documents, each representing a complete legal act. These documents serve as the input to the preprocessing pipeline.

4.2 Structured Citation Extraction

The pre-processing stage converts each XML document into structured JSON items representing potential legal references. We first parse the XML and build a parent map to identify whether a reference occurs in an article or in the preamble and to recover its local context.

After thorough and exhaustive manual tracking of how Fedlex heuristically cites SR relations, we concluded citation candidates can be extracted from these four sources. First, we process **authorial notes**, which often contain hyperlink-based SR references. For each note with an **eli/cc** link, we extract the target SR, target URL, enclosing article identifier, normalized note content, and a paragraph-level snippet containing the current note while excluding unrelated neighboring notes.

Second, we detect **preamble acronyms** by linking parenthesized acronyms outside authorial notes to the nearest SR-bearing authorial note in the preamble and then propagating these mappings to later article occurrences.

Third, we capture **local acronyms** inside articles: if an all-caps acronym appears immediately before an authorial note, we collect matching non-footnote lines from the same article as additional citation context.

Finally, we extract **explicit article mentions** outside authorial notes using patterns such as “Art.” and “Artikel” to capture intra-document references not covered by hyperlink-based extraction. The result is a de-duplicated set of JSON items encoding SR-link detections, preamble-derived acronym references, local-acronym-enhanced references, and explicit article-reference snippets.

4.3 LLM-based Article Assignment

In the second stage, we use an LLM to assign cited articles to the structured citation items. Processing is performed at the document level: for each SR document, the model receives a task-specific prompt (details included in Appendix B), a project README as auxiliary context, and the corresponding citation-candidate JSON.

We query the gpt-5.1 model with deterministic decoding (temperature = 0.0) and require strictly valid JSON output. For each item, the model determines whether the reference corresponds to (i) specific cited articles in a target SR document, (ii) a document-level reference, or (iii) an intra-document reference. The output preserves the original item identifiers and augments them with assigned article(s), a confidence label, and a short justification.

¹“eli/cc” refers to URL patterns used in the Swiss Fedlex system, where ELI (European Legislation Identifier) provides a standardized, machine-readable identifier for legal documents, and “cc” denotes consolidated versions of federal acts or ordinances. Together, such links point to specific, up-to-date legal provisions within the Swiss legal corpus.

This setup enables the resolution of implicit, acronym-based, and long-distance references that are difficult to capture with deterministic rules.

5 Results and Discussion

5.1 Validation on Edge Cases

Before applying the pipeline to the full corpus, we perform targeted manual validation to ensure the correct handling of diverse citation patterns. We select a representative subset of 30 Landesrecht SR documents so that all identified citation edge-case categories are covered at least once.

For each document, all extracted article-level dependencies are manually verified against the source text. The pipeline achieves full correctness on this subset after iterative refinement, indicating that the preprocessing and LLM-based assignment components are able to handle the known structural and semantic variations of legal references. In Appendix C, we provide a more detailed analysis of five SR documents.

This validation step provides confidence that errors observed at scale are not primarily due to unhandled edge cases, but rather arise from other factors such as output reliability.

5.2 Structural Consistency and Coverage

We analyze the outputs of the pipeline over the full dataset, comprising 2,103 SR documents and a total of 63,821 citation items. Each item is expected to contain at least the fields `ITEM`, `ARTICLE_EID`, `TARGET_SR`, and `assigned_articles`, which together define a valid article-level dependency. Here, `ITEM` denotes a unique identifier for the extracted citation instance, `ARTICLE_EID` identifies the citing (home) article within the source SR document, `TARGET_SR` specifies the referenced SR document, and `assigned_articles` contains the article(s) within the target document that are predicted to be cited.

An item is considered *structurally inconsistent* if any of these required fields are missing. Overall, 50,240 items (78.72%) are structurally consistent, while 13,581 items (21.28%) violate the expected schema. Thus, approximately one in five generated items is not directly usable for downstream processing.

From a pipeline perspective, this also corresponds to an end-to-end *coverage* of 78.72%, defined as the proportion of extracted citation candidates that result in structurally valid article-level

assignments. In other words, roughly four out of five detected citation instances can be successfully transformed into usable structured dependencies.

One likely reason why, despite explicit prompting to the LLM, about 20 percent of citation items were not structurally consistent is due to task load + ambiguous precedence: long inputs, duplicated `SR_link_detection` guidance, and soft rules compete for attention with JSON format rules, which weakens strict JSON compliance in practice (Liu et al., 2024). Together, (i) missing hard - structural decoding constrained -schema constraints, (ii) pressure toward natural-language slots when unsure, and (iii) heavy, partly overlapping instructions over large lists may explain the structural misses.

5.3 Recoverability of Inconsistent Items

We further distinguish between *recoverable* and *unrecoverable* inconsistencies. An item is considered recoverable if missing fields can plausibly be reconstructed from auxiliary information (e.g., the reasoning field), and unrecoverable otherwise. In particular, items missing the citing article identifier (`ARTICLE_EID`) or the pair (`TARGET_SR`, `assigned_articles`) are treated as unrecoverable.

Among the 13,581 inconsistent items, only 1,624 (11.96%) are recoverable, corresponding to 2.54% of all items. The remaining 11,957 items (88.04% of inconsistent items, 18.74% overall) are unrecoverable. This indicates that most structural failures are not minor omissions but reflect fundamental breakdowns in output formation, limiting the effectiveness of post hoc correction.

5.4 Failure Modes

To better understand these failures, we analyze which fields are missing in unrecoverable items. The dominant failure mode is the absence of `ARTICLE_EID`, which occurs in 11,919 items (99.68% of unrecoverable cases). In contrast, only a negligible number of items fail due to missing target information alone.

This suggests that the primary bottleneck lies in reliably maintaining the association between extracted citation contexts and their originating articles, rather than in identifying target statutes or articles.

Finally, a key room for improvement is vetting the accuracy of the citation links that were given by this paper’s methodology. While ideally one would manually review at least 100 randomly selected

structurally consistent SR documents, due to time and financial constraints, we could not conduct such tests outside of the 30 mentioned in section 5.1.

6 Conclusion

We present a hybrid pipeline for extracting article-level dependencies from Swiss federal legislation, combining deterministic XML preprocessing with LLM-based semantic resolution. We also release three complementary data splits—document-level JSON, citation candidates, and LLM-based article assignments—providing a reusable resource for Swiss legal NLP. Our results show that, while LLMs can resolve complex legal references, structural reliability remains a key bottleneck, with over 20% of outputs failing to meet schema requirements. The resulting dataset enables downstream applications such as legal knowledge graph construction, citation analysis, and benchmarking of LLMs for structured information extraction.

7 Limitations

Our approach has several limitations. First, the pipeline relies on an LLM for article assignment, which introduces variability and structural unreliability: over 20% of outputs do not conform to the expected schema. Second, the method is evaluated only on German-language *Landesrecht* documents, and its generalizability to other languages or legal domains remains to be tested. Third, while manual validation confirms correctness on representative edge cases, large-scale semantic accuracy beyond structural consistency is not exhaustively evaluated.

Finally, the approach depends on heuristic preprocessing tailored to Fedlex XML, which may limit portability to other legal corpora with different structures. Addressing these limitations would require stronger guarantees on structured output, broader multilingual evaluation, and more comprehensive annotation for large-scale benchmarking.

References

- Annarita Felici. 2025. *Cheu-lex: a parallel multilingual corpus of swiss and eu legislation*. *Applied Corpus Linguistics*, 5(3):100151.
- Stefan Höfler and Michael Piotrowski. 2011. *Building corpora for the philological study of swiss legal texts*. *Journal for Language Technology and Computational Linguistics*, 26:77–89.

Hans Kelsen. 1967. *Pure Theory of Law*. University of California Press, Berkeley. Translated by Max Knight.

Anne Lise Kjær. 2000. *On the structure of legal knowledge : The importance of knowing legal rules for understanding legal texts*, pages 127–162. De Gruyter Mouton, Berlin, New York.

Teng Lin, Yizhang Zhu, Yuyu Luo, and Nan Tang. 2025. *Srag: Structured retrieval-augmented generation for multi-entity question answering over wikipedia graph*. *Preprint*, arXiv:2503.01346.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. *Lost in the middle: How language models use long contexts*. *Preprint*, arXiv:2307.03172. Preprint; frequently cited as Liu et al. 2023 on arXiv.

Luca Rolshoven, Vishvaksenan Rasiah, Srinanda Brügger Bose, Sarah Hostettler, Lara Burkhalter, Matthias Stürmer, and Joel Niklaus. 2025. *Unlocking legal knowledge: A multilingual dataset for judicial summarization in switzerland*. *Preprint*, arXiv:2410.13456.

Darren Yow-Bang Wang, Zhengyuan Shen, Soumya Smruti Mishra, Zhichao Xu, Yifei Teng, and Haibo Ding. 2025. *Slot: Structuring the output of large language models*. *Preprint*, arXiv:2505.04016.

A Citation Edge Case Taxonomy

In order to develop a Fedlex processing pipeline that preserves relevant contextual information for the LLM to assign article assignments, we exhaustively identified the semantic edge cases that LLMs will struggle with without additional context.

A.1 Edge Case 1: Very Short SR Documents Without Articles or Footnotes

Some *Landesrecht* SR documents contain extremely short legislative texts that do not include any formal articles or footnotes. In these cases, the *home article* does not exist as a discrete structural article element; instead, the entire SR document functions as a minimal unarticulated law. When such a document is referenced from another SR document, the reference relationship cannot rely on article-to-article mapping, since the *cited article* is structurally absent. This requires semantic handling that identifies the entire cited SR document as the target rather than an article-level node. Automated extraction must therefore detect missing article tags and treat the law as a single undivided article. See SR 901.2² for an example.

²<https://www.fedlex.admin.ch/eli/cc/2020/68/de>

A.2 Edge Case 2: References Appearing in the Preamble and Then Reused by Later Articles

In some cases, the *home article* indirectly references a cited SR document via a citation that originally appears only in the preamble of the *home SR document* with an acronym. Later, another article within the same SR document makes a direct reference to the same SR link using the acronym declared in the preamble, creating a chain of implicit and explicit relationships. Semantically, this requires connecting preambular citations with downstream article-level acronym references, even though the preamble itself is not an article and therefore cannot be treated as a normal *home article*. Extraction systems must store and propagate preamble-level citations as valid contextual citations for subsequent articles by remembering the cited SR document, its associated acronym, and article information. See article 9 of SR 901.022.2³ for an example.

A.3 Edge Case 3: Article Citing Another Article Within the Same SR Document

A further complication arises when a *home article* cites another *cited article* located within the same *home SR document*. In these cases, the reference does not point to a different *cited SR document* but instead refers internally to a separate article of the same law. These internal references often appear inline, embedded within the same sentence as other legislative content, and may cite ranges of articles (e.g., “Articles 4–7”). Because the citation is internal to the same SR document, the extraction system must avoid incorrectly interpreting it as an external reference. Additionally, this case can conflict with Edge Case 8, where sub-headers or preamble-derived acronyms may give the illusion of an intra-document reference when the citation actually points outward. Therefore, the algorithm must restrict its evaluation strictly to the sentence-level context when determining whether an article-to-article reference is internal or external. Semantic disambiguation is required to ensure that same-law citations are correctly classified as intradocument article relations rather than external SR dependencies. See paragraph 1, article 9 of SR 901.0⁴ for an example.

³https://www.fedlex.admin.ch/eli/cc/2016/350/de#art_9

⁴https://www.fedlex.admin.ch/eli/cc/2007/136/de#art_9

A.4 Edge Case 4: Missing Article Numbers for Cited SR Documents

Another challenge arises when the *home article* cites a *cited SR document* but does not specify in words or with article-granular hyperlink the *cited article* within that document. Automated systems must parse the sentence and surrounding sentences to infer which SR identifiers correspond to which article reference, while also allowing “article unspecified” relationships when appropriate. See article 18 of SR 725.11⁵ for an example.

A.5 Edge Case 5: Hyperlink Footers Containing Multiple Sources and Article Information

Many *home articles* contain footnotes or hyperlink footers listing multiple legislative sources in compact footnotes. These footers may include several cited AS documents and, in some cases, specific *cited articles* of cited SR law. Because the formatting compresses multiple references into a shared footnote, an extraction system must identify and isolate references corresponding only to the cited SR documents. Then the semantic task is to analyze the surrounding sentences of the sentence that had the footnote and disaggregate mixed citation lists into discrete article-to-article relationships. See article 14 of SR 725.11⁶ for an example.

A.6 Edge Case 6: Article Number Provided but Appearing in a Different Location Than the Cited SR Link

In this case, a *home article* cites an SR document via a hyperlink but provides the *cited article number* at a different location in the same paragraph. This breaks the typical assumption that the SR hyperlink and article number appear adjacently. Extraction must use semantic association—linking the textual location of the SR number to the nearest or logically corresponding SR hyperlink—to distinguish this case from mixed-source footers (Edge Case 5). The mapping requires reading the sentence semantics of surrounding sentences within the paragraph rather than purely structural XML patterns. See article 27d of SR 725.11⁷ for an example.

⁵https://www.fedlex.admin.ch/eli/cc/1960/525_569_555/de#art_18

⁶https://www.fedlex.admin.ch/eli/cc/1960/525_569_555/de#art_14

⁷https://www.fedlex.admin.ch/eli/cc/1960/525_569_555/de#art_27_d

A.7 Edge Case 7: Combined Intra-Document Article Citations and External SR References Within the Same Sentence

This edge case represents a hybrid scenario involving the complexities of both Edge Case 3 and Edge Case 4. The *home article* contains intra-document references to other *cited articles* within the same *home SR document*—often expressed as inline ranges such as “Articles 4–7”—while simultaneously referencing an external *cited SR document* within the exact same sentence. The external SR reference may even appear without a corresponding *cited article* number, further increasing ambiguity. Because both reference types appear in close proximity and share similar surface patterns, a naive heuristic extraction system may incorrectly merge them, misinterpret internal article ranges as belonging to the external SR document, or otherwise conflate distinct citation relationships.

Resolving this edge case requires sentence-level semantic decomposition. The algorithm must determine whether each article number or article range is an intra-document citation (i.e., referencing another article in the same SR document) or whether the reference is directed toward a different *cited SR document*. External SR references lacking article numbers must be preserved as valid cross-document links, while article ranges such as “4–7” must be interpreted exclusively within the context of the home SR document. Only through semantic differentiation can the system avoid misclassifying internal article citations as external legal dependencies, or vice versa. See article 114 of SR 711⁸ for an example. The first SR hyperlink 113 is edge case 3 of not mentioning an article and referencing the whole document. Succeeding that line is a self-reference article line. This cannot easily be heuristically filtered as two separate article references.

A.8 Edge Case 8: Cited SR Link Acronym Mentioned in the Subheader of the Home Article

Some SR documents include acronyms or abbreviated SR references in the subheader of a *home article*, often derived from the cited SR document’s preamble or title. These references do not follow standard hyperlink formatting and therefore require semantic recognition to determine whether

⁸https://www.fedlex.admin.ch/eli/cc/47/689_701_723/de#art_114

the acronym genuinely refers to a *cited SR document*. The example below shows that the cited article reference and the related cited article are written above the home article content but below the home article title. Extraction must account for these references that appear outside the article body, treating them as valid citations even when no explicit hyperlink is present. See article 1 of SR 611.01⁹ for an example.

A.9 Edge Case 9: A Cited SR Document Named Explicitly in Text After a Hyperlink

In this case, it is similar to edge case 6, but instead of the article number, the cited SR document is referenced again by name elsewhere in the same home article. The *home article* references an SR document via hyperlink (e.g., “SR 311.0”) but subsequently names the cited SR document explicitly in plain text (e.g., “Strafgesetzbuch”), extending the reference across multiple textual segments. This creates a compound citation that must be semantically unified: the hyperlink and the textual name must be merged into a single relationship with the correct *cited SR document*. Automated systems must therefore remember and treat adjacent textual mentions/acronyms as belonging to the same citation event rather than two separate references. See article 16 of SR 170.32¹⁰ for an example.

B LLM Prompt

Below, we provide the prompt used for LLM-based article assignment.

Read the README.md file for context.

Task: For each input JSON item, assign the
→ correct cited article(s).
Return a JSON array where each output item
→ corresponds to the input "ITEM".

General rules:

- "Artikel" and "Art." are equivalent.
- Preamble SR references apply globally to all
→ articles.

Item types:

1. SR_link_detection:
Use the "snippet" to determine whether article
→ numbers belong to TARGET_SR.
Note: Presence of an article number does not
→ guarantee assignment.

⁹https://www.fedlex.admin.ch/eli/cc/2006/228/de#art_1

¹⁰https://www.fedlex.admin.ch/eli/cc/1958/1413_1483_1489/de#art_16

2. preamble_acronym:
Use "snippet" and "snippet_acronym" to assign
↪ articles via preamble mappings.

3. artikel_reference:
Detect intra-document references (no
↪ TARGET_SR).

Edge cases:

- Conjunctions: "Art. 30, 33, 35 und 36"
- Ranges: "Artikel 4-6"

```
-----]
↪ -----
CRITICAL RULE: CONTEXTUAL RELATIONSHIP ANALYSIS
-----]
↪ -----
```

PRIMARY INDICATORS:

- Direct marking: "Art. 32 FHG"
- Prepositional link: "nach Art. 20 der
↪ Verordnung (SR ...)"
- Parenthetical association

SECONDARY INDICATORS:

- Clause dependency
- Sentence-level proximity with connectors

SELF-REFERENCE (DO NOT assign):

- No SR nearby
- Separate syntactic structure
- Internal references

Procedure:

1. Identify syntactic boundaries
2. Locate SR references
3. Check grammatical link
4. If unclear → self-reference

Example:

"(Art. 32 und 58 FHG) ... nach Artikel 12 Absatz
↪ 4 FHG ...
nach den Artikeln 18 und 21"

- Assign: 32, 58, 12
- Do NOT assign: 18, 21

```
-----]
↪ -----
```

Output:

- Valid JSON only
- Preserve ITEM identifiers
- Include assigned_articles, TARGET_SR,
↪ confidence, reasoning

DO NOT use automated scripts. Perform full
↪ semantic analysis.

C Manual verification result

Below, we provide additional details on the manual validation procedure described in Section 5.1. We report representative results for a subset of SR documents selected to ensure coverage of all identified citation edge cases. For each document, all extracted article-level dependencies were manually

checked against the source text.

Table 1 summarizes the presence of selected edge-case categories (C5–C9) and the corresponding verification accuracy. Across all evaluated documents, the pipeline achieves full correctness, and while it does not guarantee the same for the whole Fedlex, it indicates that the proposed approach reliably handles the diverse citation patterns observed in Swiss federal legislation.

	C5	C6	C7	C8	C9	Acc.
SR 711	Yes	Yes	No	Yes	No	100%
SR 725.11	Yes	Yes	Yes	Yes	No	100%
SR 420.1	Yes	Yes	No	Yes	No	100%
SR 611.01	Yes	Yes	Yes	No	Yes	100%

Table 1: Edge-case coverage and manual verification accuracy for four example Landesrecht SR documents. Columns C5–C9 indicate the presence of specific citation edge-case types (Cases 5–9), as defined in Appendix A. Each document is selected to ensure coverage of different edge-case categories, and all extracted article-level dependencies are manually verified.

Data Augmentation for Historical NER: A Systematic Comparison of Lexical and LLM-based Approaches

Léa Blinière¹, Maud Ehrmann¹,
Emanuela Boros¹, Simon Clematide², Frédéric Kaplan¹

¹École Polytechnique Fédérale de Lausanne, Switzerland

²University of Zurich, Switzerland

lea@bliniere.com {first.last}@epfl.ch simon.clematide@cl.uzh.ch

Abstract

Named Entity Recognition (NER) on historical materials suffers significant performance degradation compared with modern text, owing to optical character recognition (OCR) errors, language evolution, and scarce annotated training data. Although various remedies have been explored to increase robustness and generalization, data augmentation techniques, despite their proven effectiveness on modern NER benchmarks, remain largely unexplored in the historical setting. This article investigates data augmentation strategies for historical NER through a systematic comparison of two complementary approaches: intrinsic augmentation via mention replacement and extrinsic augmentation through large language model (LLM)-based corpus annotation. We experiment with different augmentation variants and corpus sizes on French and German Swiss historical newspapers. Our results show contrasting patterns: mention replacement yields stable improvements across settings, whereas LLM-based silver data is most useful at moderate scale and when quality-filtered, but its effectiveness degrades as additional pseudo-labeled data is introduced. Overall, simple lexical augmentation emerges as the more robust strategy for historical NER, while LLM-based approaches remain sensitive to annotation noise and data shift.

1 Introduction

Over the past two decades, large-scale digitization efforts by cultural heritage institutions across Europe and beyond have made vast collections of newspapers, books, and archival documents available in machine-readable form (Balk and ConTEH, 2011; Neudecker and Antonacopoulos, 2016), opening new possibilities for large-scale information extraction and exploration of historical sources (Neudecker, 2022; Ehrmann et al., 2023a; Bunout et al., 2023). Among the semantic enrichments applied to such collections, named entities

– people, places, and organizations – stand out as particularly valuable: their automatic recognition provides key entry points for document retrieval, semantic indexing, and the tracing of historical actors, events, and geographies across large corpora (Gooding, 2016; Fokkens et al., 2018; Düring et al., 2023; Koolen et al., 2023).

Despite this potential, Named Entity Recognition (NER) on historical texts remains substantially more challenging than on modern data (van Strien et al., 2020). OCR noise, diachronic language variation, and domain-specific genre and layout conventions all complicate the identification of entity boundaries and types (Hamdi et al., 2020; Ehrmann et al., 2023b). Moreover, annotated training data for historical NER is scarce and expensive to produce, especially in multilingual settings and under fine-grained annotation schemes. As a result, even strong NER models often show marked performance degradation when applied to historical newspaper corpora (Schweter and Baiter, 2019; Todorov and Colavizza, 2020).

Data augmentation is a natural way to address this limitation, with techniques successfully applied across NLP tasks and, more specifically, to NER across a range of domains and settings (Feng et al., 2021; Huang et al., 2025). In the historical setting, however, such techniques remain underexplored and it is unclear which forms of synthetic training data are most useful. In particular, there is still little evidence on the relative utility of two plausible strategies: lexical augmentation based on existing gold annotations, and LLM-based pseudo-labeling of additional in-domain text to create silver-standard training material. This raises a practical question: can data augmentation improve historical NER, and if so, which strategy proves most effective?

This article addresses this question through a controlled comparison of these two augmentation strategies for historical NER on French and Ger-

man Swiss historical newspapers. We compare intrinsic augmentation via mention replacement, which preserves gold labels while varying entity surface forms, with extrinsic augmentation based on LLM-generated silver annotations for additional in-domain sentences. Our results show that the two approaches behave differently: mention replacement yields stable improvements across settings, whereas LLM-based silver data is most useful at moderate scale and becomes less effective as more pseudo-labeled data is added. These findings suggest that simple mention replacement augmentation is currently the more robust strategy for historical NER, while LLM-based augmentation remains more sensitive to annotation noise and data shift.

Our contributions are as follows:

(1) We present a systematic comparison of lexical and LLM-based data augmentation for historical NER, evaluated on French and German Swiss historical newspapers across multiple augmentation variants and corpus sizes.

(2) We show that mention replacement yields stable gains, whereas LLM-based silver data is most useful at moderate scale and degrades with additional pseudo-labeled data.

(3) We discuss the implications of this contrast for the practical design of augmentation pipelines in low-resource historical NER.

The following sections describe the related work, data, methods, experiments, and results in turn.

2 Related Work

2.1 Historical NER

The HIPE shared task series¹ has played an important role in documenting the challenges of historical NE processing and establishing standardized evaluation protocols across multiple languages and document types, with editions in 2020 and 2022 (Ehrmann et al., 2020, 2022) and a forthcoming 2026 edition focusing on person-place relation extraction (Opitz et al., 2026). Participating teams developed both data-centered approaches, notably transfer learning through historical domain pretraining (Schweter et al., 2022), and model-centered ones, such as architectural modifications to improve OCR robustness (Boros et al., 2020). While these strategies yield consistent gains, performance on historical documents remains below that achieved on modern benchmarks.

¹Identifying Historical People, Places and other Entities: <https://hipe-eval.github.io>

A core limiting factor is the scarcity of annotated training data. Historical document collections span vast topical, linguistic, and temporal diversity — from 17th-century administrative records to 20th-century multilingual newspapers — making it difficult to build annotated resources with sufficient coverage. Producing gold-standard annotations is further complicated by OCR noise, evolving naming conventions, and period-specific entities, rendering the process both slow and costly (Ehrmann et al., 2023b). Despite continuous efforts to expand available resources — including recent work by Schneider et al. (2025) — dataset scale and diversity remain limited, highlighting the need for more cost-effective strategies to expand training data.

2.2 Data Augmentation for NER

Data augmentation (DA) refers to strategies that increase the size and diversity of training examples without explicitly collecting new data. While its application to NLP is complicated by language’s compositional nature — naive transformations risk producing incoherent or semantically inconsistent samples — augmentation techniques have nonetheless been successfully deployed across a range of NLP tasks addressing low-resource settings, corpus bias, and class imbalance (Feng et al., 2021). For NER specifically, DA has emerged as a valuable strategy for tackling data scarcity, with approaches spanning simple rule-based transformations to generative prompt-based methods (Huang et al., 2025).

Among rule-based methods, mention replacement (MR) has emerged as a widely used baseline. MR substitutes entity mentions in annotated sentences with alternative mentions of the same type drawn from the training corpus, preserving both labels and the distributional properties of the original data. Dai and Adel (2020) demonstrate consistent improvements in low-resource biomedical and materials science settings. Subsequent work has proposed semantically and contextually informed extensions: Phan and Nguyen (2022) introduce semantic neighbor replacement, which constrains substitutions to similar entity mentions using embedding-based filtering, yielding gains over random MR on biomedical corpora, while Bartolini et al. (2023) further refine this with dynamic context-aware selection, improving performance in few-shot scenarios. A shared limitation of these approaches is their closed-world assumption: augmentation draws exclusively from existing training entities, amplifying rather than correcting

any dataset biases.

LLMs offer complementary augmentation strategies, though with varying degrees of reliability. A first line of work leverages LLMs to assist annotation: [Naraki et al. \(2024\)](#) show that LLMs can identify and correct errors in existing annotations, including missing entities and label switches, with hybrid human-LLM labels outperforming purely manual ones on CoNLL-2003, though the authors caution that LLMs tend toward label imbalance. A second line generates entirely new annotated training instances: [Dao et al. \(2025\)](#) demonstrate gains on biomedical NER across multiple languages using LLaMA-3.2, while [Kamath and Vajjala \(2025\)](#) evaluate GPT-4-generated data for low-resource medical NER across eleven languages. Both studies, however, reach consistent conclusions regarding synthetic data limitations: synthetic annotations tend to introduce hallucinated entities, small amounts of gold data consistently outperform larger synthetic datasets, and LLMs struggle with domain-specific terminology — limitations likely compounded in the historical setting, where OCR artifacts and time-specific language fall outside the distribution of models trained primarily on contemporary text.

Finally, direct LLM inference for NER has also been explored, but consistently falls short of supervised models: even sophisticated few-shot strategies combining kNN-based demonstration selection and self-verification cannot match supervised performance on CoNLL-2003 ([Wang et al., 2025](#)), and this gap persists on historical documents, where [Zhang and Colavizza \(2025\)](#) find that DeepSeek-V3 with retrieval-based few-shot prompting remains below state-of-the-art fine-tuned models on HIPE-2022. This further motivates the use of LLMs as annotation tools rather than inference engines.

To our knowledge, this is the first systematic study of DA for NER on historical newspaper data.

3 Data

Our experiments draw on two historical newspaper datasets: the hiPE-2020 NE-annotated dataset for model training and evaluation, and historical-corpus as the external corpus for LLM-based corpus annotation.

Coarse tag set	Fine tag set
PERS	PERS.IND PERS.COLL PERS.IND.ARTICLEAUTHOR
ORG	ORG.ADM ORG.ENT ORG.ENT.PRESSAGENCY
PROD	PROD.MEDIA PROD.DOCTR
TIME	TIME.DATE.ABS
LOC	LOC.ADM.TOWN LOC.ADM.REG LOC.ADM.NAT LOC.ADM.SUP LOC.PHYS.GEO LOC.PHYS.HYDRO LOC.PHYS.ASTRO LOC.ORO LOC.FAC LOC.ADD.PHYS LOC.ADD.ELEC LOC.UNK

Table 1: HIPE dataset entity types for NERC.

3.1 HIPE NER Dataset

Developed for the HIPE-2020 and 2022 evaluation campaigns on NE recognition and linking in historical documents, the HIPE dataset comprises several NE-annotated collections, primarily historical newspapers and classical commentaries. We use the French and German portions of the hiPE-2020 subset, consisting of articles from Swiss and Luxembourgish newspapers (19th–20th century)². Annotations cover two tasks: entity recognition and classification, and entity linking, the latter not considered here. For NER, the dataset provides two levels of granularity: NER-Coarse with five top-level entity types, and NER-Fine with 21 specific subtypes (Table 1). Metonymic readings are additionally annotated for PERS, ORG, and LOC under NER-Coarse-Meto. Annotation followed detailed guidelines covering entity mention form (mainly proper names), type definitions and coverage, and annotation rules ([Ehrmann et al., 2020](#)). Corpus statistics and entity type distributions across splits are provided in Appendix A (Tables 6–8).

3.2 Corpus of Historical Newspapers

The historical-corpus dataset serves as the external source for the LLM-based annotation experiments. Sampled from the same digitized Swiss and Luxembourgish historical newspaper collection as

²Specifically, release v2.1 of the 2022 campaign: <https://github.com/hiPE-eval/HIPE-2022-data/releases/tag/v2.1-test-all-unmasked>

hipe-2020, it comprises yearly editorial content samples from four German and seven French newspapers spanning 1876–1945, already segmented into sentences. It therefore shares the same domain, time period, and textual characteristics, including OCR noise.

4 Methods

Our approach uses fine-tuned BERT models as baselines and compares two data augmentation approaches: intrinsic augmentation via mention replacement, which recombines existing gold annotations from hipe-2020, and extrinsic LLM corpus annotation, which generates pseudo-labeled data from the external historical-corpus.

4.1 Baseline Models

We adopt an extended BERT architecture that, following Boros et al. (2020), adds two transformer encoder layers between the pretrained backbone and the classification head. NER is performed at the sentence level, with each sentence processed independently as input to the model. As pretrained backbone, we use HMBERT (Schweter et al., 2022; Schweter, 2022), a multilingual model trained on digitized historical documents in five languages – including French and German – drawn from Europeana newspaper collections and British Library books (17th–20th century). Unlike standard multilingual BERT pretrained on contemporary Wikipedia, its historical pretraining corpus exposes the model to OCR noise and historical language variation, making it a well-suited choice for historical NER.

This baseline configuration – extended BERT with the HMBERT backbone – is referred to as HIST-base. It was selected based on an earlier French-only version of this study, in which 12 model configurations (3 pretrained models \times 2 architectures \times 2 sequence lengths) were systematically evaluated across 5 random seeds. HIST-base emerged as both a top-performing baseline and the configuration most responsive to the MR and LCA augmentation strategies. Further details are provided in Blinière (2026) (Sections 4.2, 5.1, and 5.4). The same pretrained weights and architecture are used for both languages, ensuring that performance differences across languages are attributable to augmentation strategies rather than model configuration.

4.2 Augmentation Strategies

4.2.1 Mention Replacement (MR)

MR substitutes entity mentions in annotated sentences with alternative mentions of the same type drawn from the training corpus. Given a sentence containing an entity (e.g., *Charles de Gaulle*), the method replaces it with another entity of the same type (e.g., *Winston Churchill*), generating a new training example while preserving sentence structure and annotation labels. This approach has two key advantages: label consistency is guaranteed since replacement mentions originate from gold-annotated data, and the distributional properties of the original corpus – including OCR noise and domain-specific vocabulary – are preserved in the augmented sentences.

MR candidate pool selection. For each sentence s in the hipe-2020 training set containing an entity mention m_i , a candidate pool $\mathcal{C}(m_i)$ is constructed by filtering all other entities in the training set through three successive stages: (1) type filtering, retaining only entities matching the coarse type, metonymic status, and fine-grained type of m_i ;³ (2) quality filtering, excluding mentions with annotation issues such as mentions spanning sentence boundaries; (3) deduplication filtering, discarding near-identical mentions via OCR-robust sentence embeddings (Michail et al., 2025), removing mention candidates with cosine similarity ≥ 0.99 . For each candidate c_j in $\mathcal{C}(m_i)$, an augmented sentence s' is generated by substituting m_i with c_j while leaving all other mentions unchanged. Up to two candidates are sampled per mention, yielding up to $2 \times N$ augmented sentences for a sentence containing N mentions.

MR augmentation variants. We evaluate four candidate-selection variants:

1. **Random:** no additional constraints.
2. **Semantic:** retains augmented sentences with sentence similarity between s and s' of at least 0.85 (same OCR-robust model), ensuring contextual coherence between the original and augmented sentences.
3. **Temporal:** restricts replacements to mentions from documents published within ± 10 years,

³Candidate selection does not enforce grammatical correctness; the impact of this limitation was not quantified or evaluated.

preserving period-specific naming conventions.

4. Semantic+Temporal: combines both constraints.

4.2.2 LLM Corpus Annotation (LCA)

LLM corpus annotation generates new pseudo-labeled training data by annotating sentences from `historical-corpus`.

Annotation. Three LLMs serve as annotators: GPT-5-Mini (OpenAI), Mistral-Small-3.2 (Mistral AI), and Qwen3-Next-80B (Alibaba Cloud). Both the model set and the prompting strategy were selected through preliminary experiments on the `hipe-2020` test set (FR), in which multiple LLMs and prompt configurations were compared based on NER performance; Qwen3-Next-80B achieved the highest overall performance, followed by Mistral-Small-3.2 and GPT-5-Mini (see [Blinière \(2026\)](#), Section 4.4.2).

For each language, 25,000 sentences are sampled from the `historical-corpus` (seed = 42), restricted to editorial content (no ads) and to sentences with lengths in $[20, 500]$ tokens. Each sentence is independently annotated by all three models using an identical system prompt – in French and German – describing the HIPE annotation scheme and guidelines (see Appendix C).

Ensembling. LLM predictions are validated using three strategies with increasing levels of quality control, trading off recall for precision through stricter agreement and validation constraints⁴:

1. LLM-Strict: all three LLMs must detect an overlapping span with matching Coarse and Fine types; Qwen boundaries are retained. This strategy maximizes precision at the expense of recall.
2. LLM-Majority: at least two of three LLMs must predict spans that overlap on at least one token and have matching types; boundaries follow Qwen > Mistral > GPT.
3. LLM-Majority+BERT: LLM majority vote confirmed by HIST-base (confidence ≥ 0.5) on an overlapping span with matching types; TIME entities reintroduced from BERT for validated sentences. This strategy combines

⁴Mentions of type TIME are excluded from ensembling due to poor zero-shot performance observed in preliminary experiments on the `hipe-2020` test set.

LLM consensus with BERT verification for higher annotation reliability.

Only sentences with at least one validated mention are retained, ensuring that augmented data contributes an NER signal, though this increases the proportion of entity-bearing sentences relative to the original corpus.

4.2.3 Augmented HIPE NER Dataset Construction

Both strategies produce pools of NE-annotated sentences that are appended to the original `hipe-2020` training data. To study the effect of augmented training data size, we construct eight augmented datasets ranging from +25% to +200% of `hipe-2020` sentence count, in steps of 25%. All augmented datasets include the full HIPE training set. Augmented sentences are added using nested sampling, such that lower augmentation levels are strict subsets of higher ones (e.g., $25\% \subset 50\% \subset \dots \subset 200\%$), enabling controlled comparison of scaling effects.

The two augmentation methods differ in their sentence selection strategies. Mention replacement uses a round-robin diversification scheme that maximizes both sentence and entity replacement diversity across augmentation levels. LCA applies filtered random selection, retaining only sentences containing at least one validated entity. Each resulting dataset is used to independently fine-tune HIST-base under the configuration described in Section 5. Table 2 reports entity volume growth and mean entity density per sentence for French at +25% and +200%. German exhibits comparable trends. Full per-level statistics for both languages – including entity density, volume growth, type distributions, and rates of novel entity introduction – are provided in Figures 2–5 (Appendix E).

In total, two augmentation paradigms, seven variants, and eight dataset scales are evaluated across two languages.

5 Experimental Setup

Training configuration. HIST-base models are fine-tuned for 3 epochs using AdamW optimization with a constant learning rate and a maximum sequence length of 512 tokens. Full hyperparameter details are provided in Appendix B.

Evaluation metrics. We evaluate at entity level, treating each entity mention as a single unit regardless of token length. Evaluation follows the

Method	+25%		+200%	
	Vol.	Dens.	Vol.	Dens.
Baseline	—	2.17	—	2.17
MR-Random	+45%	2.19	+519%	2.99
MR-Temporal	+45%	2.19	+519%	2.99
MR-Semantic	+51%	2.28	+928%	4.97
MR-Sem+Temp	+51%	2.28	+928%	4.97
LLM-Strict	+48%	2.23	+361%	2.33
LLM-Majority	+53%	2.31	+424%	2.53
LLM-Maj+BERT	+47%	2.22	+373%	2.29

Table 2: Entity volume growth ($\Delta\%$ vs. baseline) and mean entity density (ent./sent.) at +25% and +200% augmentation levels, for French (baseline: 7,138 entities). Full per-level evolution in Figures 2 and 3 (Appendix E).

HIPE scorer protocol⁵, with one adaptation: entity mentions spanning automatic sentence boundaries are split into two distinct gold entities, one per sentence. This is necessary because automatic sentence segmentation is not always reliable and models process text sentence by sentence. The HIPE scorer supports two evaluation settings: *strict*, requiring exact boundary and type match, and *fuzzy*, allowing partial boundary overlap. We adopt strict evaluation throughout. Performance is reported as micro-averaged F1 across all entity types. To account for training stochasticity, all models are trained with five random seeds and results are reported as mean F1 \pm standard deviation.

This work is designed to be fully replicable. All code and augmented datasets are publicly available⁶. The *hipe-2020* dataset is distributed publicly and our baseline replicates the architecture of Boros et al. (2020) on the same benchmark.

6 Results

6.1 Overall NER Performance

Figure 1 shows F1 as a function of augmentation level for coarse- and fine-grained NER, from which three patterns emerge.

First, all four MR variants match or improve on the baseline at every augmentation level, and their trajectories are nearly indistinguishable, suggesting that augmentation volume matters more than candidate selection strategy. By contrast, LLM-Strict and LLM-Majority fall below the baseline from the first augmentation step and de-

⁵<https://github.com/hipe-eval/HIPE-scorer>

⁶<https://github.com/impresso/impresso-named-entity-data-augmentation>

Task	System	French		German	
		F1	Δ	F1	Δ
Coarse	HIST-base	76.2 \pm 0.5	—	71.7 \pm 0.8	—
	Best MR	77.2 \pm 0.6	+0.9	74.9 \pm 1.0	+3.2
	Best LCA	76.7 \pm 0.6	+0.5	71.8 \pm 1.1	+0.1
Fine	HIST-base	69.3 \pm 0.7	—	63.2 \pm 0.7	—
	Best MR	72.4 \pm 0.7	+3.2	69.1 \pm 0.6	+5.9
	Best LCA	70.7 \pm 0.6	+1.4	65.2 \pm 0.8	+2.0

Table 3: F1 scores (mean \pm std over 5 seeds) of best MR and LCA variants for Coarse and Fine NER in French and German. Δ = absolute improvement over baseline (pp). Best configurations for each cell are listed in Table 10.

Type	HIST-base		Δ MR		Δ LCA	
	FR	DE	FR	DE	FR	DE
LOC	82.6	82.4	-0.1	+2.2	+0.1	+0.1
PERS	75.9	67.9	-0.2	+4.0	-0.1	-0.3
ORG	50.3	44.0	+8.0	+5.8	+3.4	+3.0
PROD	64.3	50.7	+10.1	+0.1	+2.5	-6.5
TIME	60.5	62.7	+1.8	+11.1	+0.4	+3.0

Table 4: Baseline F1 and Δ F1 (pp) per coarse entity type. MR = best per language (FR: Semantic +150%; DE: Sem+Temporal +125%); LCA = LLM-Majority+BERT +25% (both languages).

cline monotonically as more pseudo-labeled data is added. LLM-Majority+BERT occupies an intermediate position: it remains near-neutral or slightly positive on NER-Coarse, while consistently improving over the baseline on NER-Fine, peaking at early augmentation levels (+25% in French, approximately +75% in German) before stabilizing. Second, gains are generally larger in German than in French across all methods and tasks, consistent with the lower German baseline leaving more room for improvement. Third, MR gains are larger on NER-Fine than on NER-Coarse, a pattern we examine further at the entity type level in Section 6.2.

Table 3 summarizes peak performance across settings for both granularity levels. The best MR configurations (detailed in Table 10) yield gains ranging from +0.9 pp (FR, coarse) to +5.9 pp (DE, fine), whereas the best LCA configurations reach +1.4 pp (FR, fine) and +2.0 pp (DE, fine) but remain close to the baseline on coarse NER. Full per-configuration results are reported in Tables 11–16 (Appendix D).

6.2 Performance by Entity Type

Table 4 reports Δ F1 by coarse entity type for the best MR and LCA configurations. In French, gains

NER Learning Curves — Coarse Literal vs Fine-Grained

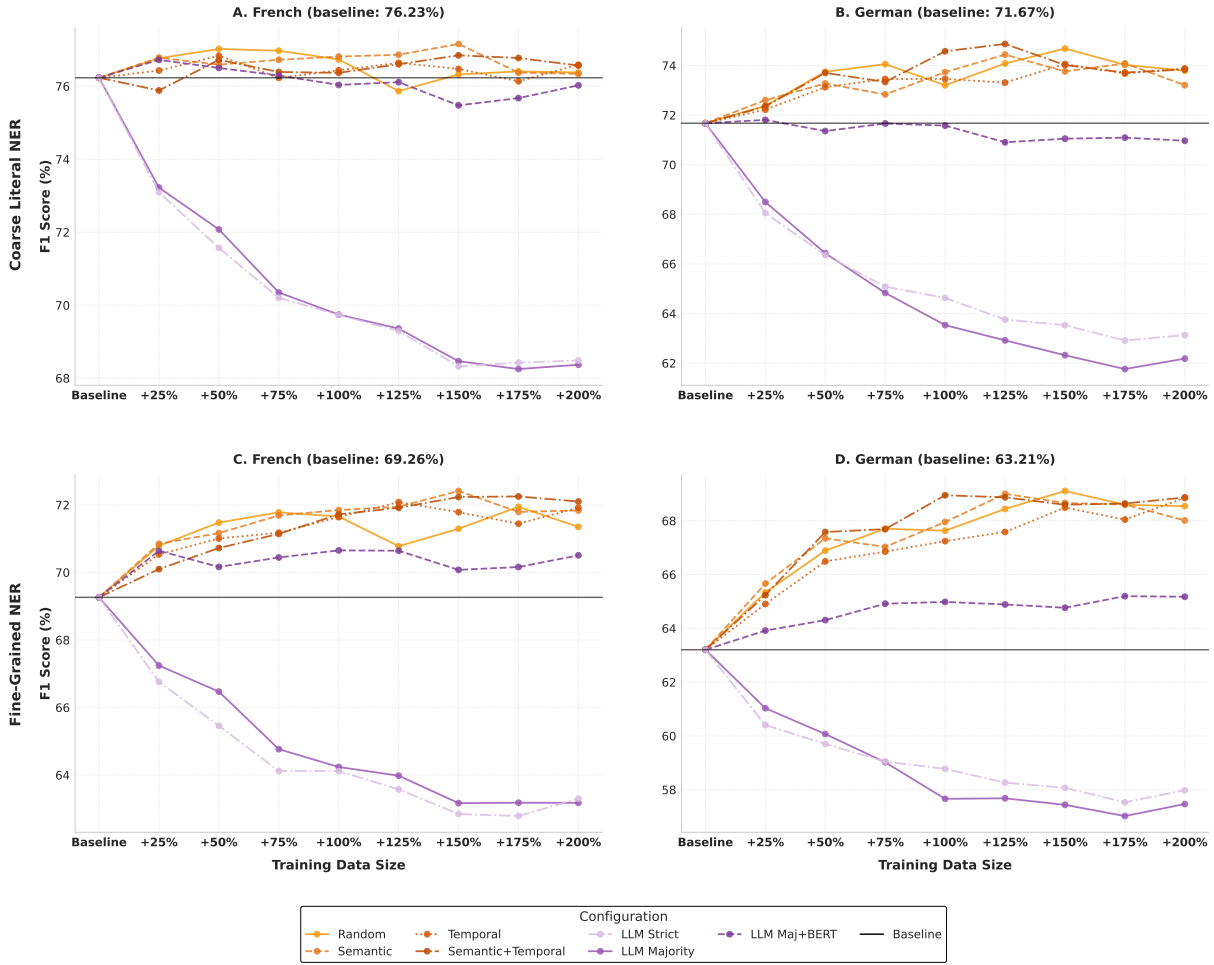


Figure 1: NER F1 as a function of augmentation level for French (left) and German (right). Top: coarse-grained NER; bottom: fine-grained NER. The horizontal line indicates the baseline.

from both methods concentrate on the types with the lowest baseline performance, that is, `ORG` (baseline 50.3; Δ best MR: +8.0 pp, Δ best LCA: +3.4 pp) and `PROD` (baseline 64.3; Δ best MR: +10.1 pp, Δ best LCA: +2.5 pp), while `LOC` and `PERS` remain largely unchanged. In German, `PROD`, despite its lower baseline, is essentially unchanged under MR (+0.1 pp) but is severely degraded by LCA (-6.5 pp). The largest gains in German concentrate on `ORG` and `TIME`, the latter showing the strongest MR improvement (+11.1 pp).

At the fine-grained level (Table 5), MR consistently maintains or improves performance across all subtypes in both languages. The largest gains occur for rare subtypes, including `ADM.SUP` (+33.8 pp FR, +77.6 pp DE) and `ENT.PRESSAGENCY` (+15.4 pp FR, +49.8 pp DE). By contrast, the most frequent subtypes, which already showed

the strongest baseline performance, change little: `PERS.IND` and `ADM.TOWN` remain stable or improve only marginally in both languages.

LCA results show a more uneven pattern. Some rare subtypes benefit substantially: `PHYS.HYDRO` gains +21.8 pp in French and +10.2 pp in German, while `ADM.SUP` improves by +15.4 pp and +6.1 pp, respectively. However, other subtypes degrade sharply, including `LOC.FAC` in French (-5.5 pp) and `PROD.MEDIA` in German (-10.1 pp). This disparity suggests that LLM annotation quality varies across entity types, with some categories being more prone to systematic errors.

7 Discussion

Why mention replacement is consistently effective. The robustness of MR across task granularity levels, languages, and augmentation levels

Parent	Subtype	n		HIST-base		Δ MR		Δ LCA	
		FR	DE	FR	DE	FR	DE	FR	DE
LOC	ADM.TOWN	450	257	78.6	76.7	+0.7	+3.5	-0.1	+0.8
	ADM.NAT	151	161	81.7	82.3	+3.5	+3.6	+2.7	+2.1
	ADM.REG	147	84	38.3	35.9	+14.2	+19.1	+5.5	+7.4
	PHYS.GEO	28	14	37.5	0.0	+11.7	+31.4	-5.4	+7.9
	PHYS.HYDRO	23	29	38.0	30.5	+31.6	+16.3	+21.8	+10.2
	ADM.SUP	19	21	46.3	3.6	+33.8	+77.6	+15.4	+6.1
	ORO [†]	19	5	66.3	56.6	+0.4	+6.0	-0.2	-24.2
	FAC	18	14	16.1	2.2	+10.9	+29.7	-5.5	-1.3
	PHYS.ASTRO [†]	—	10	—	0.0	—	0.0	—	0.0
UNK [†]	3	1	0.0	0.0	0.0	0.0	0.0	0.0	
ORG	ENT	69	85	48.0	39.9	+5.6	+5.9	+2.8	+7.5
	ADM	43	29	47.5	37.1	+10.0	+10.0	+4.0	+10.0
	ENT.PRESSAGENCY	20	18	64.0	25.0	+15.4	+49.8	+0.1	+24.0
PERS	IND	519	330	76.3	68.3	-0.3	+2.9	-0.2	-0.2
	IND.ARTICLEAUTHOR [†]	13	3	13.0	0.0	+19.8	0.0	-1.6	0.0
	COLL [†]	5	—	0.0	—	0.0	—	0.0	—
PROD	MEDIA	58	61	66.3	47.9	+9.6	+2.0	+2.8	-10.1
	DOCTR [†]	3	5	28.6	10.0	+26.0	+56.2	+0.7	+14.7

Table 5: Δ F1 (pp) per fine-grained subtype for all subtypes, sorted by n (test set entity mention count, descending) within each parent type. MR = best per language (FR: Semantic +150%; DE: Random +150%); LCA = best LLM-Majority+BERT per language (FR: +100%; DE: +175%). [†]Test set too small for reliable evaluation.

can be attributed to two complementary properties. First, label consistency and distributional faithfulness to the original corpus are guaranteed by construction. Second, the near-identical performance of all MR variants (Section 6) suggests that surface form diversity alone drives the improvement, with semantic or temporal filtering providing no measurable benefit. This simplicity is practically significant: random mention replacement is sufficient and requires no additional resources beyond the training set itself.

Why LLM annotation degrades with scale.

The monotonic degradation of LLM-Strict and LLM-Majority points to systematic annotation errors rather than random noise. We attribute this primarily to the complexity of the HIPE annotation guidelines: 21 fine-grained types and metonymic annotations make it difficult for LLMs to produce consistent annotations, leading to systematic type assignment errors and missed entities. The BERT validation step in LLM-Majority+BERT partially mitigates this effect by filtering pseudo-labels through a domain-aware model trained on gold data, but does not eliminate it entirely. Crucially, as more pseudo-labeled data is introduced, these residual errors accumulate in the training signal, explaining the continuous performance decline observed beyond moderate augmentation levels.

Practical implications and limitations. These results carry clear practical implications for historical NER practitioners: mention replacement is currently the more reliable augmentation strategy, requiring only a gold-annotated training set and yielding consistent gains across all experimental conditions. LLM-based augmentation may complement MR at moderate scale when validated by BERT, but should not be relied upon as a primary augmentation strategy.

This work is subject to several limitations: results are obtained with a single model architecture and dataset (HIPE-2020, French and German), and may not generalize to other historical languages, periods, or annotation schemes. The closed-world assumption of MR means it cannot introduce entity surface forms absent from the training data, potentially limiting its effectiveness in very low-resource settings.

8 Conclusion

This article presented a systematic comparison of two data augmentation strategies for historical NER on French and German historical newspapers. Mention replacement (MR) proved consistently effective and robust across all experimental conditions, yielding gains of +0.93 to +5.90 F1 points depending on the language and task granularity. LLM-based corpus annotation (LCA) was beneficial only

under strict validation and at moderate scale, with smaller gains of up to +1.99 F1 points; performance deteriorated as more pseudo-labeled data was added. These results show that simple label-preserving augmentation is currently the more reliable strategy, while LLM-based approaches remain limited by annotation noise, guideline complexity, and distributional mismatch with historical text.

Several directions for future work emerge from these findings. First, mention replacement could be improved through frequency-weighted candidate sampling, prioritizing rare entities and entity forms underrepresented in the training data. Second, LLM-based annotation would benefit from more sophisticated approaches, including prompt revision to reduce entity hallucination and improve handling of OCR noise, dynamic few-shot selection, and per-type inference. Third, hybrid strategies combining the robustness of mention replacement with the broader entity coverage of externally annotated silver data warrant investigation. Finally, both strategies should be evaluated on broader historical collections covering additional languages, periods, and annotation schemes, to assess the generalizability of the findings beyond the hi-pe-2020 subset of the HIPE benchmark.

References

- Hildelies Balk and Aly Conteh. 2011. [IMPACT: Centre of Competence in Text Digitisation](#). In *Proceedings of the 2011 Workshop on Historical Document Imaging and Processing*, HIP '11, pages 155–160, Beijing, China, USA. ACM.
- Ilaria Bartolini, Vincenzo Moscato, Marco Postiglione, Giancarlo Sperli, and Andrea Vignali. 2023. [Data augmentation via context similarity: An application to biomedical Named Entity Recognition](#). *Information Systems*, 119:102291.
- Léa Blinière. 2026. [Data augmentation strategies for historical named entity recognition](#). Master's thesis, École Polytechnique Fédérale de Lausanne, February.
- Emanuela Boros, Ahmed Hamdi, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Jose G. Moreno, Nicolas Sidere, and Antoine Doucet. 2020. [Alleviating digitization errors in named entity recognition for historical documents](#). In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 431–441, Online. Association for Computational Linguistics.
- Estelle Bunout, Maud Ehrmann, and Frédéric Clavert, editors. 2023. [Digitized Newspapers - A New Eldorado for Historians? Reflections on Tools, Methods and Epistemology](#). Studies in Digital History and Hermeneutics. De Gruyter Oldenbourg, Berlin, Germany.
- Xiang Dai and Heike Adel. 2020. [An Analysis of Simple Data Augmentation for Named Entity Recognition](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3861–3867, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- An Dao, Hiroki Teranishi, Yuji Matsumoto, Florian Boudin, and Akiko Aizawa. 2025. [Overcoming data scarcity in named entity recognition: Synthetic data generation with large language models](#). In *Proceedings of the 24th Workshop on Biomedical Language Processing*, pages 328–340, Vienna, Austria. Association for Computational Linguistics.
- Marten Düring, Matteo Romanello, Maud Ehrmann, Kaspar Beelen, Daniele Guido, Brecht Deseure, Estelle Bunout, Jana Keck, and Petros Apostolopoulos. 2023. [Impresso Text Reuse at Scale. An interface for the exploration of text reuse data in semantically enriched historical newspapers](#). *Frontiers in Big Data*, 6.
- Ehrmann, Watter, Romanello, Clematide, and Flückiger. 2020. [Impresso Named Entity Annotation Guidelines](#).
- Maud Ehrmann, Marten Düring, Clemens Neudecker, and Antoine Doucet. 2023a. [Computational Approaches to Digitised Historical Newspapers \(Dagstuhl Seminar 22292\)](#). *Dagstuhl Reports*, 12(7):112–179.
- Maud Ehrmann, Ahmed Hamdi, Elvys Linhares Pontes, Matteo Romanello, and Antoine Doucet. 2023b. [Named Entity Recognition and Classification in Historical Documents: A Survey](#). *ACM Computing Surveys*, 56(2):27:1–27:47.
- Maud Ehrmann, Matteo Romanello, Alex Flückiger, and Simon Clematide. 2020. Extended overview of CLEF HIPE 2020: Named entity processing on historical newspapers. In *CEUR Workshop Proceedings*, 2696. CEUR-WS.
- Maud Ehrmann, Matteo Romanello, Sven Najem-Meyer, Antoine Doucet, and Simon Clematide. 2022. [Extended Overview of HIPE-2022: Named Entity Recognition and Linking in Multilingual Historical Documents](#). In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*. CEUR-WS.
- Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. [A Survey of Data Augmentation Approaches for NLP](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

- Antske Fokkens, Serge ter Braake, Ronald Sluijter, Paul Longley Arthur, and Eveline Wandl-Vogt, editors. 2018. *Proceedings of the Second Conference on Biographical Data in a Digital World 2017*, volume 2119 of *CEUR Workshop Proceedings*. CEUR-WS.org, Linz, Austria.
- Paul Gooding. 2016. Exploring the information behaviour of users of Welsh Newspapers Online through web log analysis. *Journal of Documentation*, 72(2):232–246.
- Ahmed Hamdi, Axel Jean-Caurant, Nicolas Sidère, Mickaël Coustaty, and Antoine Doucet. 2020. Assessing and Minimizing the Impact of OCR Quality on Named Entity Recognition. In *Digital Libraries for Open Knowledge*, Lecture Notes in Computer Science, pages 87–101, Cham. Springer International Publishing.
- Yi Huang, Yuhan Gao, and Chengjuan Ren. 2025. A survey of data augmentation in named entity recognition. *Neurocomputing*, 651:130856.
- Gaurav Kamath and Sowmya Vajjala. 2025. Does Synthetic Data Help Named Entity Recognition for Low-Resource Languages? In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 159–167, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.
- Marijn Koolen, Rik Hoekstra, Joris Oddens, Ronald Sluijter, Rutger Van Koert, Gijsjan Brouwer, and Hennie Brugman. 2023. The Value of Preexisting Structures for Digital Access: Modelling the Resolutions of the Dutch States General. *J. Comput. Cult. Herit.*, 16(1):1:1–1:24.
- Andrianos Michail, Juri Opitz, Yining Wang, Robin Meister, Rico Sennrich, and Simon Clematide. 2025. Cheap Character Noise for OCR-Robust Multilingual Embeddings. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11705–11716, Vienna, Austria. Association for Computational Linguistics.
- Yuji Naraki, Ryosuke Yamaki, Yoshikazu Ikeda, Takafumi Horie, Kotaro Yoshida, Ryotaro Shimizu, and Hiroki Naganuma. 2024. Augmenting NER Datasets with LLMs: Towards Automated and Refined Annotation. *Preprint*, arXiv:2404.01334.
- Clemens Neudecker. 2022. Cultural Heritage as Data: Digital Curation and Artificial Intelligence in Libraries. In *Proceedings of the Third Conference on Digital Curation Technologies (Qurator 2022)*, volume 3234 of *CEUR Workshop Proceedings*, Berlin, Germany. CEUR.
- Clemens Neudecker and Apostolos Antonacopoulos. 2016. Making Europe’s Historical Newspapers Searchable. In *Proceedings of the 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 405–410, Santorini, Greece. IEEE.
- Juri Opitz, Corina Raclé, Emanuela Boros, Andrianos Michail, Matteo Romanello, Maud Ehrmann, and Simon Clematide. 2026. CLEF HIPE-2026: Evaluating Accurate and Efficient Person-Place Relation Extraction from Multilingual Historical Texts. *Preprint*, arXiv:2602.17663.
- Uyen Phan and Nhung Nguyen. 2022. Simple semantic-based data augmentation for named entity recognition in biomedical texts. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 123–129, Dublin, Ireland. Association for Computational Linguistics.
- Sophie Schneider, Ulrike Förstel, Kai Labusch, Jörg Lehmann, and Clemens Neudecker. 2025. ZEFYS2025: A German Historical Newspaper Dataset for Named Entity Recognition and Entity Linking. In *Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long and Short Papers*, pages 48–58, Hannover, Germany. HsH Applied Academics.
- Stefan Schweter. 2022. Hugging Face model - dbmdz/bert-base-historic-multilingual-cased.
- Stefan Schweter and Johannes Baiter. 2019. Towards Robust Named Entity Recognition for Historic German. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepLANLP-2019)*, pages 96–103, Florence, Italy. Association for Computational Linguistics.
- Stefan Schweter, Luisa März, Katharina Schmid, and Erion Çano. 2022. hmbERT: Historical Multilingual Language Models for Named Entity Recognition. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, volume 3180 of *CEUR Workshop Proceedings*, pages 1109–1129, Bologna, Italy. CEUR.
- Konstantin Todorov and Giovanni Colavizza. 2020. Transfer learning for named entity recognition in historical corpora. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, volume 2696, pages 1–12, Thessaloniki, Greece. CEUR-WS.
- Daniel van Strien, Kaspar Beelen, Mariona Ardanuy, Kasra Hosseini, Barbara McGillivray, and Giovanni Colavizza. 2020. Assessing the Impact of OCR Quality on Downstream NLP Tasks. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence*, pages 484–496, Valletta, Malta. SCITEPRESS - Science and Technology Publications.
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. GPT-NER: Named Entity Recognition via Large Language Models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4257–4275, Albuquerque, New Mexico. Association for Computational Linguistics.

A HIPE Dataset Statistics

Tables 6–8 provide an overview of the hipe-2020 dataset used in our experiments. Table 6 reports document, sentence, and entity mention counts per split and language. Table 7 shows the coarse-grained entity type distribution as a percentage of total mentions. Table 8 details the fine-grained subtype distribution, reporting mention counts and corpus-wide ranks across languages and splits.

Lang.	Split	Docs	Sents	Mentions
FR	Train	158	5,743	7,138
	Dev	43	1,244	1,746
	Test	43	1,462	1,642
DE	Train	103	3,472	3,655
	Dev	33	1,202	1,279
	Test	49	1,217	1,176

Table 6: Document, sentence, and entity mention counts per split for French and German hipe-2020.

Lang.	Split	LOC	PERS	ORG	PROD	TIME
FR	Train	43.5%	37.6%	12.2%	2.8%	4.0%
	Dev	44.4%	39.7%	9.2%	2.8%	3.9%
	Test	52.3%	32.7%	8.0%	3.7%	3.3%
DE	Train	47.6%	35.6%	9.9%	3.5%	3.4%
	Dev	46.0%	31.5%	13.1%	4.0%	5.5%
	Test	50.7%	28.3%	11.2%	5.6%	4.2%

Table 7: Coarse entity type distribution (% of total entities) per split for French and German hipe-2020.

B Training Configuration

HIST-base models are trained using the following hyperparameters, selected based on preliminary experiments (Boros et al., 2020):

Table 9: Training hyperparameters

Hyperparameter	Value
Batch size	32
Learning rate	5e-5
Optimizer	AdamW
ϵ (AdamW)	1e-8
Weight decay	0.0
Epochs	3
Gradient clipping	Max norm 1.0
Warmup steps	0
Learning rate schedule	Constant

Models are trained on NVIDIA A100 GPUs

Parent	Subtype	FR				DE			
		Train		Test		Train		Test	
		<i>n</i>	Rank	<i>n</i>	Rank	<i>n</i>	Rank	<i>n</i>	Rank
LOC FR: 3,106 / 858 DE: 1,741 / 596	ADM.TOWN	1,695	#2	450	#2	687	#2	257	#2
	ADM.NAT	648	#3	151	#3	564	#3	161	#3
	ADM.REG	382	#5	147	#4	199	#5	84	#5
	ORO	91	#10	19	#12	52	#11	5	#15
	PHYS.GEO	85	#11	28	#9	58	#10	14	#12
	PHYS.HYDRO	73	#12	23	#10	88	#9	29	#8
	ADM.SUP	63	#13	19	#12	42	#13	21	#10
	FAC	59	#14	18	#14	46	#12	14	#12
	PHYS.ASTRO	—	—	—	—	2	#17	10	#14
	ADD.PHYS	4	#18	—	—	2	#17	—	—
	ADD.ELEC	3	#19	—	—	—	—	—	—
	UNK	3	#19	3	#17	1	#19	1	#18
ORG FR: 868 / 132 DE: 362 / 132	ENT	599	#4	69	#5	200	#4	85	#4
	ADM	212	#7	43	#8	136	#6	29	#8
	ENT.PRESSAGENCY	57	#15	20	#11	26	#14	18	#11
PERS FR: 2,682 / 537 DE: 1,302 / 333	IND	2,553	#1	519	#1	1,288	#1	330	#1
	IND.ARTICLEAUTHOR	109	#9	13	#15	—	—	3	#17
	COLL	20	#17	5	#16	14	#16	—	—
PROD FR: 200 / 61 DE: 127 / 66	MEDIA	152	#8	58	#6	108	#8	61	#6
	DOCTR	48	#16	3	#17	19	#15	5	#15
TIME FR: 282 / 54 DE: 123 / 49	DATE.ABS	282	#6	54	#7	123	#7	49	#7

Table 8: Entity type distribution across train and test splits for French (FR) and German (DE) hi-pe-2020. For each fine-grained subtype, mention count n and corpus-wide rank are reported (by descending frequency, per language and split). Coarse parent totals are shown as train/test counts. “—” indicates subtypes absent from a given split.

(40GB VRAM). Training duration per configuration ranges from 15–20 minutes.

C LLM Annotation Prompt

All three LLMs received an identical system prompt specifying the annotation task and guidelines. The prompt covers: (1) task definition and entity types; (2) entity boundary rules, including modifiers, appositions, and coordination; (3) handling of OCR errors (annotate text as-is, without correction); (4) metonymic annotation criteria; and (5) output format requirements (JSON with character-level offsets).

The full prompt is available at <https://github.com/impresso/impresso-named-entity-data-augmentation/tree/main/prompts>.

The following excerpt illustrates the OCR handling instructions, which are particularly critical for historical text:

8. GESTION DES ERREURS OCR Principe fondamental : TOUJOURS annoter le texte TEL QUEL,

SANS CORRECTION.

Les offsets (start/end) doivent pointer exactement sur les chaînes bruitées dans le texte.

Exemples : “Léo Blanchard” → annoter la chaîne entière ; “L . Bridel” → annoter avec espaces ; “Ocan Atlantique” pour “Océan Atlantique” → annoter tel quel.

D Full Results per Configuration

Table 10 identifies the best augmentation variant and level for each task and language, corresponding to the peak performances reported in Table 3. Tables 11–16 then report $\Delta F1$ (pp) vs. baseline for every configuration, across all tasks and languages. **Bold** indicates the best Mention Replacement and best LCA configuration in each table, as listed in Table 10.

Task	Lang.	Best MR	Best LCA
Coarse	FR	Semantic +150%	LLM-Maj+BERT +25%
	DE	Sem+Temp +125%	LLM-Maj+BERT +25%
Fine	FR	Semantic +150%	LLM-Maj+BERT +100%
	DE	Random +150%	LLM-Maj+BERT +175%
Meto	FR	Semantic +125%	LLM-Maj+BERT +25%
	DE	Random +150%	LLM-Maj+BERT +75%

Table 10: Best augmentation configuration (variant and level) for each cell in Table 3.

E Augmentation Data Statistics

Figures 2–5 report augmentation data statistics across all methods, levels, and languages. Figure 2 shows mean entity density (entities per sentence) and Figure 3 total entity volume ($\Delta\%$ vs. baseline) for all augmentation levels. Figure 4 shows entity type distribution evolution across augmentation levels, and Figure 5 reports the proportion of novel entities (surface forms absent from the original training data) introduced by each LCA configuration.

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+0.53	+0.55	+0.20	-0.35	-3.14	-3.01	+0.49
+50%	+0.79	+0.36	+0.60	+0.49	-4.66	-4.16	+0.27
+75%	+0.74	+0.49	0.00	+0.16	-6.03	-5.89	+0.06
+100%	+0.50	+0.58	+0.20	+0.13	-6.50	-6.49	-0.20
+125%	-0.37	+0.63	+0.41	+0.37	-6.94	-6.87	-0.12
+150%	+0.09	+0.93	+0.24	+0.61	-7.91	-7.77	-0.76
+175%	+0.17	+0.14	-0.10	+0.54	-7.81	-7.98	-0.56
+200%	+0.14	+0.11	+0.36	+0.34	-7.75	-7.86	-0.21

Table 11: $\Delta F1$ (pp) per configuration — **Coarse, French**. Baseline: 76.23 ± 0.53 .

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+0.71	+0.93	+0.55	+0.68	-3.62	-3.18	+0.13
+50%	+2.08	+1.60	+1.46	+2.03	-5.31	-5.24	-0.32
+75%	+2.38	+1.16	+1.78	+1.66	-6.59	-6.84	-0.01
+100%	+1.53	+2.06	+1.79	+2.90	-7.04	-8.14	-0.10
+125%	+2.41	+2.77	+1.64	+3.20	-7.91	-8.75	-0.77
+150%	+3.01	+2.09	+2.39	+2.35	-8.14	-9.35	-0.62
+175%	+2.36	+2.41	+2.00	+2.04	-8.76	-9.91	-0.58
+200%	+2.12	+1.54	+2.21	+2.17	-8.54	-9.49	-0.71

Table 12: $\Delta F1$ (pp) per configuration — **Coarse, German**. Baseline: 71.67 ± 0.82 .

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+1.53	+1.59	+1.27	+0.84	-2.51	-2.02	+1.38
+50%	+2.21	+1.90	+1.74	+1.46	-3.80	-2.79	+0.90
+75%	+2.52	+2.43	+1.91	+1.88	-5.14	-4.50	+1.18
+100%	+2.39	+2.59	+2.38	+2.46	-5.15	-5.02	+1.39
+125%	+1.51	+2.68	+2.82	+2.65	-5.68	-5.28	+1.38
+150%	+2.03	+3.15	+2.52	+2.97	-6.41	-6.09	+0.81
+175%	+2.68	+2.53	+2.18	+2.99	-6.48	-6.08	+0.90
+200%	+2.09	+2.57	+2.65	+2.84	-5.96	-6.09	+1.24

Table 13: $\Delta F1$ (pp) per configuration — **Fine-Grained, French**. Baseline: 69.26 ± 0.72 .

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+2.13	+2.46	+1.70	+2.03	-2.80	-2.17	+0.71
+50%	+3.69	+4.14	+3.28	+4.38	-3.50	-3.13	+1.10
+75%	+4.50	+3.82	+3.65	+4.48	-4.16	-4.18	+1.71
+100%	+4.42	+4.75	+4.04	+5.74	-4.43	-5.54	+1.78
+125%	+5.23	+5.80	+4.38	+5.67	-4.94	-5.52	+1.68
+150%	+5.90	+5.46	+5.28	+5.39	-5.13	-5.77	+1.56
+175%	+5.39	+5.40	+4.84	+5.44	-5.67	-6.18	+1.99
+200%	+5.34	+4.81	+5.65	+5.66	-5.22	-5.74	+1.97

Table 14: $\Delta F1$ (pp) per configuration — **Fine-Grained, German**. Baseline: 63.21 ± 0.70 .

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+0.76	+0.68	+0.27	-0.23	-2.90	-2.81	+0.79
+50%	+0.94	+0.53	+0.80	+0.37	-4.54	-3.95	+0.46
+75%	+0.87	+0.72	+0.11	+0.23	-5.91	-5.70	+0.40
+100%	+0.72	+0.89	+0.33	+0.43	-6.41	-6.33	+0.08
+125%	-0.17	+1.04	+0.44	+0.61	-6.89	-6.82	+0.18
+150%	+0.17	+1.04	+0.37	+0.91	-7.89	-7.52	-0.50
+175%	+0.30	+0.18	+0.04	+0.85	-8.03	-7.95	-0.16
+200%	+0.17	+0.26	+0.48	+0.49	-7.73	-7.79	+0.07

Table 15: Δ F1 (pp) per configuration — **Metonymic, French**. Baseline: 73.59 ± 0.28 .

Level	Mention Replacement				LLM Corpus Annotation		
	Rand.	Sem.	Temp.	S+T	LLM-S	LLM-M	LLM-M+B
+25%	+0.77	+1.17	-0.04	+0.81	-3.28	-3.06	+0.17
+50%	+2.02	+1.38	+1.39	+1.83	-5.10	-5.13	-0.21
+75%	+2.11	+0.83	+1.53	+1.46	-6.28	-6.87	+0.27
+100%	+1.09	+1.66	+1.66	+2.75	-6.71	-8.03	+0.11
+125%	+2.38	+2.45	+1.71	+2.78	-7.56	-9.05	-0.35
+150%	+3.16	+1.89	+2.42	+2.06	-7.80	-9.45	-0.39
+175%	+2.49	+2.25	+1.86	+1.80	-8.51	-9.87	-0.27
+200%	+2.14	+1.38	+2.22	+1.81	-8.22	-9.55	-0.52

Table 16: Δ F1 (pp) per configuration — **Metonymic, German**. Baseline: 67.62 ± 0.42 .

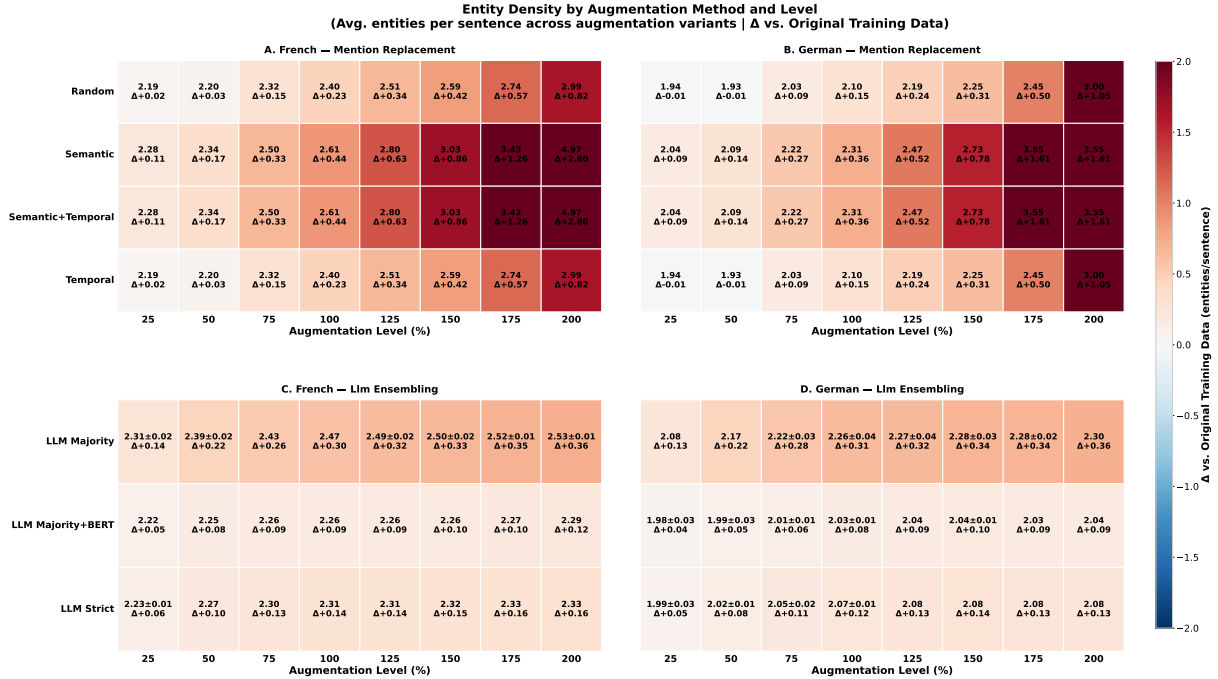


Figure 2: Mean entity density (entities/sentence, Δ vs. baseline) per augmentation method and level, for French (A, C) and German (B, D). Top: Mention Replacement; bottom: LLM Ensembling.

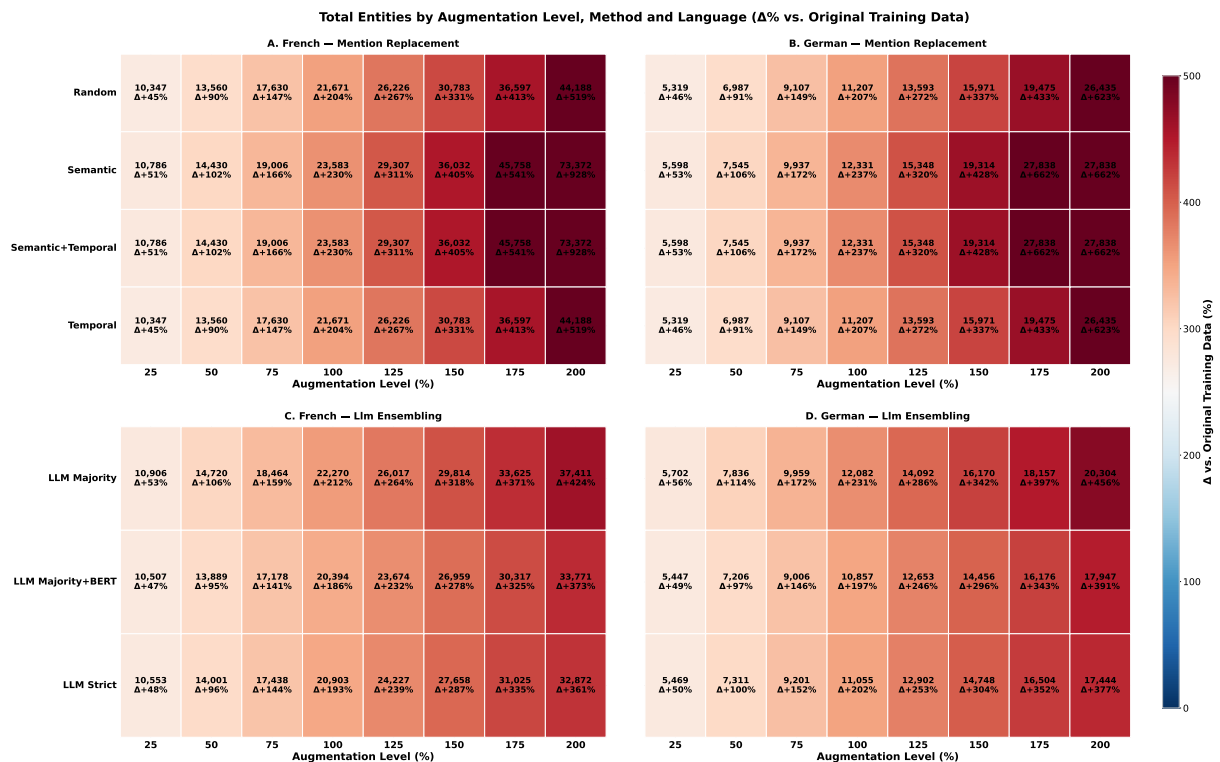


Figure 3: Total entity volume ($\Delta\%$ vs. baseline) per augmentation method and level, for French (A, C) and German (B, D). Top: Mention Replacement; bottom: LLM Ensembling.

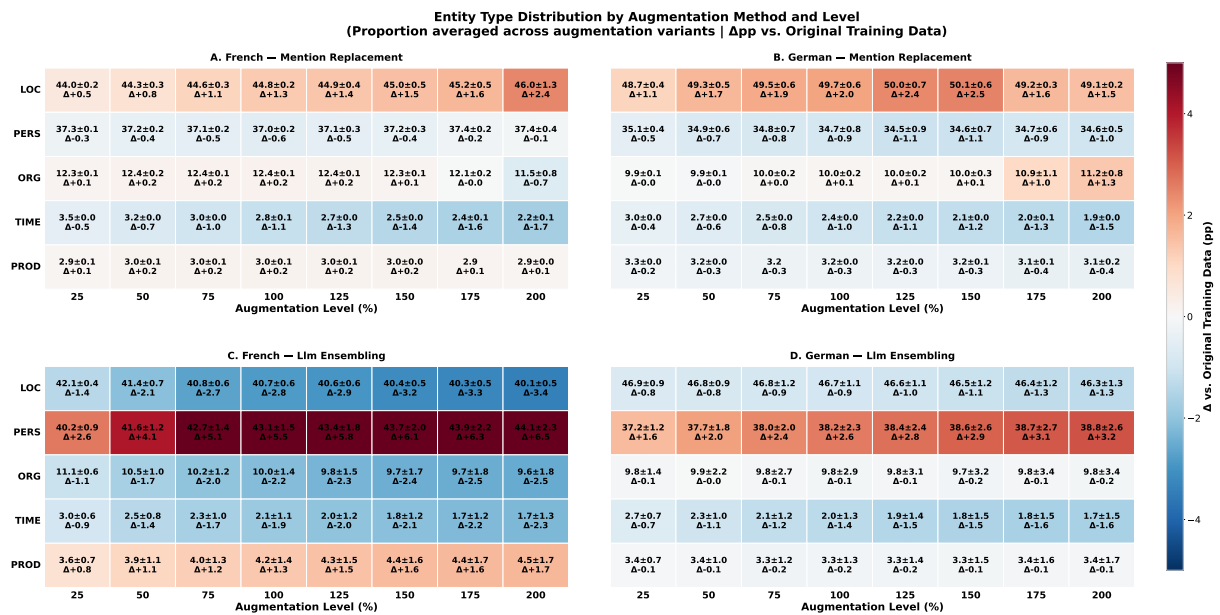


Figure 4: Entity type distribution evolution (Δpp vs. baseline) per augmentation method and level, for French (A, C) and German (B, D). Top: Mention Replacement; bottom: LLM Ensembling.

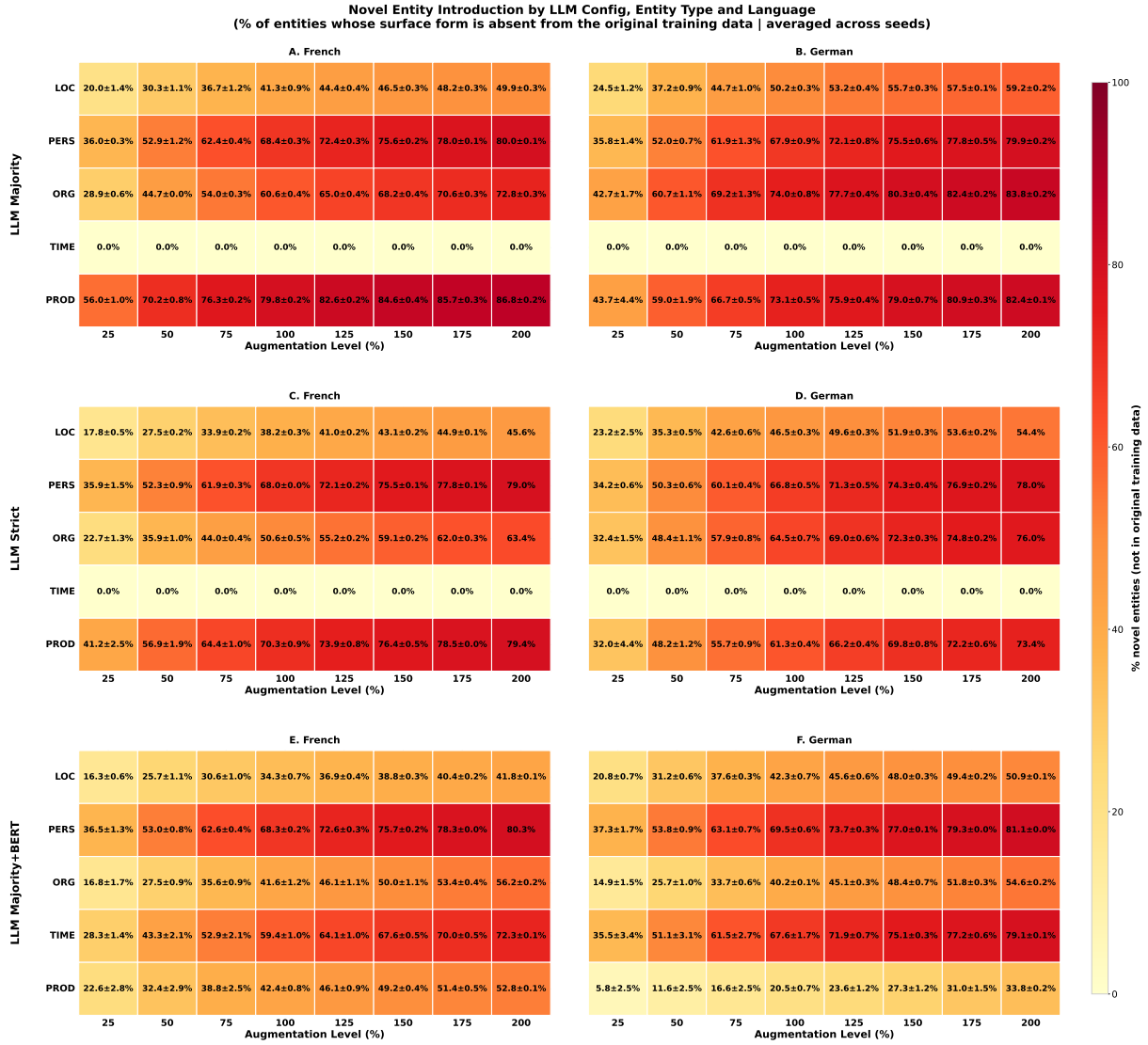


Figure 5: Proportion of novel entities (% of entity surface forms absent from the original training data) introduced by each LCA configuration (LLM-Majority, LLM-Strict, LLM-Majority+BERT), per entity type and augmentation level, for French (A, C, E) and German (B, D, F).

Code-Switching Detection in Multilingual Child Speech with SwissBERT

Agnese D’Angelo^{1,*} Sina Ahmadi² Moritz M. Daum¹ Stephanie Wermelinger¹

¹Department of Psychology, University of Zurich

²Department of Computational Linguistics, University of Zurich

*a.dangelo@psychologie.uzh.ch

Abstract

Code-switching is widespread in multilingual speech, yet its automatic detection remains challenging, especially for low-resource languages. In Switzerland, a context with multiple languages and Swiss German varieties, these challenges are amplified by variable orthography and limited annotated data. We present a supervised word-level language-identification system for code-switching detection in multilingual everyday child and adult speech, obtained by fine-tuning SwissBERT. We constructed a dataset of four languages and an *other* category, implemented controlled subword-label alignment, and evaluated performance using token-level F1. To contextualize SwissBERT’s performance, we additionally fine-tuned mBERT as a multilingual baseline. SwissBERT achieves robust word-level predictions and outperforms mBERT. We release the full training pipeline and evaluation scripts to facilitate reproducibility.

📄 | ZurichNLP/SwissBERT-CS

1 Introduction

Multilingualism shapes Switzerland’s identity: one-third of the population regularly uses more than one language (Bundesamt für Statistik, 2021). In multilingual settings, speakers often engage in code-switching (CS), the alternation between two or more languages within a conversation. CS emerges early in development (Smolak et al., 2020) and is a natural, systematic feature of multilingual communication rather than a sign of confusion or delay. CS occurs across sentences (i.e., intersentential; “I went to the park yesterday! *Ich ha Spass gha.*”) or within the same sentence (i.e., intrasentential; “Could you give me *s Buech bitte?*”). Swiss German presents unique challenges for computational models: it lacks a standardized orthography, exhibits substantial regional variation, and

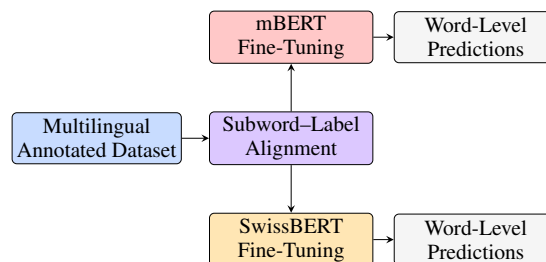


Figure 1: Overview of the methodology. A multilingual annotated dataset is aligned at the subword level and used to fine-tune SwissBERT and mBERT.

is underrepresented in existing multilingual pre-trained models.

These difficulties are amplified in child speech. Young children frequently produce phonological reductions, truncations, and non-standard word forms that deviate from adult speech (Dodd et al., 2003), making word boundaries and lexical identity less predictable. In multilingual contexts, children may additionally blend features from multiple languages within a single word or use language-specific morphology in non-target-language utterances (Paradis and Genesee, 1996), further complicating word-level language identification. Moreover, child-directed speech from caregivers often contains simplified or exaggerated forms that differ from standard adult registers (Foulkes et al., 2005), adding another layer of variation. Unlike social-media text, where non-standard spelling follows some community conventions that models can learn, the variability in child speech is less systematic and harder to anticipate from pre-training data alone.

In this work, we address these gaps by constructing a multilingual, manually annotated dataset across four languages and by fine-tuning SwissBERT (Vamvas et al., 2023)—a transformer model pre-trained on Swiss national languages and dialects—for word-level CS detection. Because SwissBERT is pre-trained on Swiss German, we hy-

pothesize that it will outperform other multilingual models. To test this hypothesis, we additionally fine-tune mBERT (Wu and Dredze 2020) as a contrastive baseline using the same training pipeline.

To our knowledge, this is the first work to apply transformer-based word-level CS detection to multilingual child speech involving Swiss German.

2 Related Work

Previous work on CS relied heavily on manual annotation to identify language boundaries (e.g., Lachemat et al., 2025). Most research on CS detection has focused on social-media text. The Second Shared Task on Language Identification in Code-Switched Data (Molina et al., 2016) provided one of the first large-scale word-level benchmarks for bilingual social-media corpora, highlighting persistent challenges, including noisy orthography, short words, and limited annotated data. Rijhwani et al. (2017) proposed a generalized word-level language identification model based on a Hidden Markov Model, enabling multilingual CS detection without manually annotated training data and demonstrating its effectiveness on Twitter data.

Recent work has increasingly turned to transformer-based architectures. Multilingual models like mBERT (Wu and Dredze, 2020) and XLM-R (Conneau et al., 2020) outperform traditional sequence models on CS language identification, especially when fine-tuned on mixed-language data (e.g., Aguilar et al., 2020; Khanuja et al., 2020; Das et al., 2023). Other studies have explored syntactic and discourse information: Sterner and Teufel (2025) showed that syntactic structure alone can support human-level CS acceptability judgments in a graph-neural-network model (CSntax-GNN), with patterns generalizing across unseen language pairs.

Data-augmentation strategies have also been proposed to improve CS language modeling. Hu et al. (2020) combined monolingual sentence selection, syntactic-constraint substitution, and a pointer-generator network, achieving substantial perplexity reductions on Mandarin–English CS corpora.

A comprehensive survey by Winata et al. (2023) reviewed more than 400 CS studies and documented a rapid increase in the number of publications. However, two gaps remain: (1) most work targets high-resource language pairs (e.g., Spanish–English, Mandarin–English), and (2)

Language	Words
Swiss German	50,733
English	49,658
Italian	48,844
French	46,281
Other	9,956

Table 1: Distribution of the 205,472 manually labeled words across Swiss German, English, French, Italian and *other*.

fine-grained word-level CS detection remains difficult for informal registers with high variability. Furthermore, automatic CS detection in child speech, particularly for low-resource languages, remains unexplored. This gap is especially relevant for developmental psychology, where CS annotation is typically performed manually. Automating this process provides a methodological bridge between computational linguistics and developmental research, enabling scalable, reproducible analyses of multilingual child language.

3 Data

Existing CS datasets and benchmarks focus primarily on adult speakers, social-media text, or high-resource language pairs. No existing resources cover multilingual child speech in Swiss German, so we created a manually annotated dataset across four languages (see Table 1), including an *other* category.

3.1 Sources

In total, we assembled a corpus of 205,472 manually labeled words across five labels (Swiss German, English, French, Italian, Other). The dataset contains 46,501 utterances with a mean utterance length of 4.42 words (SD = 5.16), ranging from 1 to 127 words per utterance. This level of variability is characteristic of spontaneous child speech, where children often produce one-word utterances while caregivers may produce much longer utterances.

English, French, and Italian words were extracted from publicly available transcribed child-speech corpora in the CHILDES database (MacWhinney, 2000), including Antelmi and Morlacchi (2005); Stine and Bohannon (1983); Burgado (2025); Hamann et al. (2003); Genesee et al. (2004); Pizzuto (2004); Tonelli (2004); Watkins (2004), both monolingual corpora and multilingual corpora with CS. More than one third (17,624 words) of the Swiss German words were obtained

from unpublished child speech data collected as part of a pre-registered¹ project in our research unit. The dataset consists of spontaneous everyday speech produced by three-year-old children and their interlocutors (i.e., caregivers, siblings, friends). The recordings were collected using microphones that the children wore for approximately 12 hours across one week in their everyday environments. Because no open-source Swiss German child speech corpora currently exist, we expanded the Swiss German portion of the dataset with an additional 33,109 tokens from the SwissDial corpus (Dogan-Schönberger et al., 2021). Although SwissDial is not child speech, its inclusion increases lexical diversity and improves coverage of Swiss German orthographic variation. Because the dataset was constructed over an extended period, the earliest stage involved artificially enriching code-switching patterns. Specifically, within a segment of 1,619 monolingual Swiss German words, we added 139 English words by translating parts of existing utterance to simulate code-switching. This approach was only used in the initial phase; as more recordings were transcribed and real-life instances of child code-switching became available, the dataset was expanded using real instances instead of artificial ones.

3.2 Preprocessing

Before word annotation, the data were preprocessed to ensure consistent tokenization across heterogeneous sources. This included: (i) splitting contractions in English, French, and Italian (e.g., *c'est*, *don't*) to ensure that each meaningful unit receives an independent label; (ii) separating Swiss German clitics (e.g., *s'Auto*); (iii) removing punctuation; (iv) normalizing white-space and removing transcription artifacts. These steps ensured that word boundaries aligned with meaningful linguistic units, which is essential for reliable word-level CS detection.

3.3 Annotation

After preprocessing, the dataset was manually labeled at the word level. The annotator assigned one of four languages (Swiss German, French, English, or Italian) to each token based on its lexical form. Proper names, place names, interjections, fillers, and words whose language could not be reliably assigned to any of the four languages (e.g., *super*,

¹https://osf.io/57wt3/overview?view_only=3fb372b514e4413ca8dbbde2056f6011

Comparison	Percent Agreement	Cohen's κ
A vs. B	0.99	0.98
A vs. GOLD	0.98	0.97
B vs. GOLD	0.98	0.97

Table 2: Inter-annotator agreement for language-label annotations.

okay) were assigned to an additional *other* category. This category prevents false code-switch detections for language-neutral or ambiguous lexical items that do not clearly belong to a single language. All annotations were first performed by an annotator fluent in all languages in the dataset.

To calculate inter-annotator agreement (IAA), we extracted a total of 200 utterances (1,831 words), equally divided into 100 monolingual utterances (balanced across Swiss German, English, French, and Italian) and 100 multilingual utterances. Only utterances containing at least three words were included, as shorter utterances do not provide enough lexical or contextual information to be reliably categorized. Two annotators (A and B), both fluent in all languages of the dataset, completed the task. Before the main annotation, they annotated 30 utterances as a training exercise, after which they proceeded independently following the annotation guidelines.

IAA was computed between the two annotators and relative to the gold standard (i.e., the labeled dataset). Percent agreement and Cohen's κ are reported in Table 2. The agreement values are extremely high, indicating that the task is straightforward and that the annotation scheme is well-defined. Disagreements were mostly found in short utterances, where the lack of contextual and lexical cues made language identification more difficult.

3.4 Sentence IDs

Each label word in the dataset is associated with a sentence identifier (i.e., Sentence ID), allowing for the reconstruction of utterance boundaries after preprocessing and tokenization. Maintaining utterance structure is crucial for analyzing code-switching patterns, as CS can occur both within and across utterances.

4 Methodology

Our approach combines manual dataset construction, controlled preprocessing, and transformer-based token classification. Figure 1 provides an overview of the methodology.

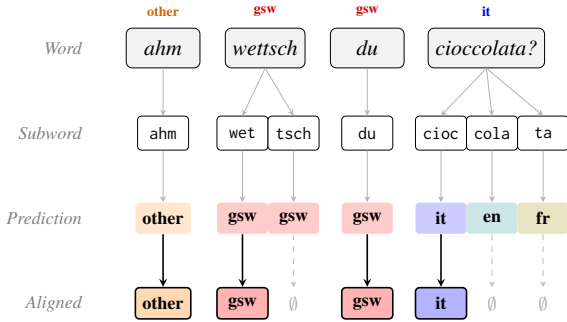


Figure 2: Example of WordPiece tokenization with corresponding word-level labels, model predictions, and loss masking. Only the first subword of each word contributes to the supervised loss.

4.1 Model Architecture

We fine-tuned SwissBERT (Vamvas et al., 2023), a multilingual transformer pre-trained on the Swiss national languages (Standard German, French, Italian, and Romansh), and more importantly, including recently-added Swiss German. A linear classification head predicts one of the five labels (Swiss German, English, French, Italian or *other*) for each word.

4.2 Baseline Model

To contextualize the performance of SwissBERT, we additionally fine-tuned mBERT (Devlin et al., 2019) using the same preprocessing, subword-label alignment, and training pipeline. mBERT serves as a contrastive baseline: unlike SwissBERT, it is not pre-trained on Swiss German. This comparison allows us to isolate the contribution of SwissBERT’s dialect-specific pretraining and assess whether exposure to Swiss German during pretraining yields measurable improvements in word-level CS detection.

4.3 Subword Tokenization and Alignment

SwissBERT uses WordPiece tokenization, which frequently splits words into multiple subword units. However, code-switching happens at the *word* level, and accurate CS detection requires a single language label per word. To reconcile word-level labels with subword tokenization, we adopt a first-subword labeling strategy: the first subword inherits the word’s language label, while all subsequent subwords are ignored during loss computation. This alignment ensures that the model learns word-level language boundaries while remaining compatible with subword-based transformer architectures.

4.4 Inference

At inference time, the model applies the same preprocessing and WordPiece tokenization used during training. As illustrated in Figure 2, SwissBERT outputs a label for every subword token. To obtain word-level predictions, we use the same strategy as in training: only the first subword is kept, and all later subwords are ignored. This mirrors the loss-masking scheme showed in Figure 2, where non-initial subwords receive a null label (\emptyset), ensuring that each word receives exactly one linguistically meaningful prediction.

4.5 Training Procedure

We fine-tuned SwissBERT with a classification head using the Hugging Face Trainer API (Wolf et al., 2020) for five epochs, a batch size of 8, and a learning rate of 5×10^{-5} . We used the AdamW optimizer with weight decay 0.01. Validation was performed at the end of each epoch using the F1 metric computed over non-masked tokens, and the best-performing checkpoint was selected.

To train the model, we randomly split the annotated dataset into training and validation sets using a 90/10 ratio. The split was performed at the utterance (sentence ID) level, ensuring that all tokens within the same utterance remained in the same partition. This prevents contextual leakage across splits and avoids artificially inflated performance. Each utterance was truncated to a maximum length of 128 tokens, which standardizes input size and ensures consistent batching during training.

We evaluate model performance using the standard F1 score, computed with the Seqeval library (Nakayama, 2018). Seqeval is widely used for sequence-labeling tasks such as named entity recognition and token-level classification, and provides a reliable implementation of precision, recall, and F1 over label sequences. We report token-level F1 computed only on non-masked tokens (i.e., the first subwords), thereby aligning the evaluation with our word-level labeling scheme. Following common practice in token-level evaluation, we use Seqeval’s micro-averaged F1 across all tokens to select the best model checkpoint during training.

5 Results

5.1 Test Setup

The models were evaluated on an independent dataset of 15,819 manually labeled words grouped into 3,420 utterances. This dataset was collected

Child	Languages	Words
A	French, Swiss German	3,442
B	Italian, Swiss German	1,960
C	English, Swiss German	5,337
D	Italian, French, English	5,080
Total		15,819

Table 3: Composition of the evaluation dataset, based on transcriptions of naturalistic everyday interactions between multilingual children and their caregivers.

Language	SwissBERT			mBERT		
	P	R	F1	P	R	F1
English	0.87	0.99	0.93	0.82	0.92	0.87
French	0.98	1.00	0.99	0.88	0.90	0.89
Italian	0.91	0.90	0.90	0.83	0.88	0.85
Swiss German	0.98	0.95	0.96	0.92	0.85	0.88
Other	0.99	0.95	0.97	0.90	0.87	0.89
Overall	0.96	0.96	0.96	0.88	0.88	0.88

Table 4: Token-level precision (P), recall (R), and F1 for SwissBERT and mBERT on the external evaluation set. SwissBERT consistently outperforms mBERT across all languages.

within the same pre-registered² project described in Section 3.1. Because it originates from a different population and recording context than the training corpus, it provides an ecologically valid estimate of model generalization. The composition of the evaluation set is provided in Table 3.

5.2 Token-level Prediction

Table 4 reports precision, recall, and F1 for each language category for both models. Across the test dataset, the fine-tuned SwissBERT model achieved an overall F1 score of 0.96. Performance was high across all languages, with F1 scores of 0.93 (English), 0.99 (French), 0.90 (Italian), 0.97 (Other), and 0.96 (Swiss German). These results indicate that SwissBERT generalizes well to naturalistic multilingual child speech despite dialectal variation and informal registers.

To assess the contribution of dialect-specific pre-training, we compared SwissBERT with mBERT, both fine-tuned on the same training data and alignment strategy. mBERT achieved an overall F1 score of 0.88. While mBERT performed reasonably well, SwissBERT outperformed it across all languages.

Overall, the results show that SwissBERT’s pre-training on Swiss German leads to better subword representations of dialectal spellings.

²<https://osf.io>

Metric	Value
Gold switch points	736
Predicted switch points	862
Precision	0.78
Recall	0.91
F1 score	0.84

Table 5: Switch-point detection performance. A switch is defined as a change in language label between consecutive tokens, excluding the label *other*.

5.3 Switch-Point Detection

To further evaluate the performance of our model, we compute its ability to correctly identify *switch-points*. These correspond to transitions between two consecutive words where the language label changes, excluding any words labeled *other*. The *other* label is treated as non-linguistic and is removed from the evaluation stream prior to computing transitions. This ensures that sequences such as English → other → Swiss German are treated as a direct English → Swiss German transition.

Switch-points are therefore computed on a cleaned word sequence in which all words with the label *other* have been removed. A predicted switch is counted as correct only if it occurs at the same position in the global sequence and exhibits the same direction of transition (e.g., Swiss German → English) as in the gold annotation. Experiments on the switch-point detection, as summarized in Table 5, show that the model achieves a switch-point precision of 0.78, recall of 0.91, and F1 score of 0.84. These results suggest that the model captures the majority of true code-switching transitions, although it tends to over-predict switch points. This additionally indicates that even though the model achieves high token-level F1, as discussed in Section 5.2, it is less effective at identifying the exact locations of language transitions.

6 Discussion & Conclusion

Our aim was to develop an automatic approach to detect code-switching in multilingual child speech. We hypothesized that SwissBERT’s pre-training on Swiss German and the Swiss national languages would provide an advantage for multilingual word-level classification. The evaluation confirmed this: SwissBERT outperformed mBERT across all languages.

A likely explanation for this performance difference lies in the training data and model specialization. SwissBERT is trained with a focus on language varieties relevant to Switzerland, allowing it to better capture linguistic nuances, vocabulary, and orthographic variation present in the dataset. In contrast, mBERT is designed as a broadly multilingual model trained on a wide range of languages, which can limit its ability to model specific regional language varieties.

The strong performance of both models may also reflect that the languages in our dataset are linguistically distant, which simplifies the classification problem. Future work with more closely related language pairs (e.g., Standard German vs. Swiss German) may pose greater challenges, because lexical overlap and shared morphology reduce the distinctiveness of subword patterns. Moreover, code-switching frequency varies widely across speakers and contexts, introducing additional variability that models must learn to handle.

A qualitative analysis also highlighted a structural limitation shared by both models: words that are orthographically identical across languages are difficult to classify reliably when they appear in isolation or in contexts with limited syntactic information. For example, the word *da* (Italian preposition and Swiss German adverb) is sometimes misclassified when isolated. In such cases, the model has no access to phonetic cues (which would distinguish the two pronunciations) or semantic cues (which would clarify the intended meaning), making misclassifications almost unavoidable. This limitation is not specific to our models but reflects an inherent ambiguity in written speech transcripts.

Finally, although the training pipeline is fully generalizable, the current model's advantage stems from SwissBERT's region-specific pre-training and is therefore limited to the Swiss context. As our research unit continues to collect more diverse multilingual child-speech data, the same corpus can be extended to train classifiers that recognize an increasingly broad set of languages, enabling progressively richer and more inclusive analyses of code-switching.

Limitations

The dataset is imbalanced. The *other* category is heterogeneous, grouping together diverse linguistic phenomena that are not easily comparable. The Swiss German portion of the dataset is not exclu-

sively composed of child-speech data, as it also includes corpus data from SwissDial that differ substantially from naturalistic child language. This mismatch introduces distributional differences that may affect model performance and limit the validity of the results. Our long-term goal is to re-train the model on a corpus composed exclusively of child-speech data as soon as additional recordings become available. Although the current model shows high performance and strong inter-annotator agreement, these metrics may shift as more closely related languages are added. This is especially true for distinctions between language varieties (e.g., Swiss German vs. Standard German).

Manual annotation is time-consuming and introduces the possibility of bias, particularly for ambiguous or context-dependent words. These challenges are amplified by the absence of phonetic information, which limits the ability to disambiguate homographs or reduced child-speech forms. Future work should integrate both phonetic and orthographic information to improve the robustness of code-switching detection.

Ethical Considerations

All data involving families were collected in accordance with the ethical standards of the 1964 Helsinki Declaration and its later amendments. The recordings contain sensitive everyday conversations between children and their caregivers; therefore, raw audio and transcripts cannot be publicly released. The dataset reflects multilingual families in the Zurich area, which may limit generalizability and introduce demographic biases. Our work is intended solely for research on early multilingual development. The authors declare no conflicts of interest.

Acknowledgments

The project is financed by the Swiss National Science Foundation (reference number 10001585). Sina Ahmadi gratefully thanks the support of the UZH Grant (reference number 269093). We also thank the Kleine Weltentdecker*innen Lab and the Jacobs Center for Productive Youth Development at Universität Zürich for the support and all children and their caregivers for their participation in our studies.

References

- Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. [LinCE: A centralized benchmark for linguistic code-switching evaluation](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1803–1813, Marseille, France. European Language Resources Association.
- Donatella Antelmi and Alessandra Morlacchi. 2005. L'interpretazione del linguaggio figurato nel ritardo mentale. *Rassegna Italiana di Linguistica Applicata*, 37(2):355–380.
- Bundesamt für Statistik. 2021. Zunahme der mehrsprachigkeit in der schweiz: 68% verwenden regelmässig mehr als eine sprache. Swiss Federal Statistical Office report.
- Laura Burgato. 2025. CHILDES Italian Burgato Corpus. CHILDES Database. 30 samples. Corpus date: 2025.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Richeek Das, Sahasra Ranjan, Shreya Pathak, and Preethi Jyothi. 2023. Improving pretraining techniques for code-switched NLP. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, volume 1, pages 1176–1191, Toronto, Canada. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Barbara Dodd, Alison Holm, Zhu Hua, and Sharon Crossbie. 2003. [Phonological development: a normative study of british english-speaking children](#). *Clinical Linguistics & Phonetics*, 17(8):617–643.
- Pelin Dogan-Schönberger, Julian Mäder, and Thomas Hofmann. 2021. [SwissDial: Parallel multidialectal corpus of spoken Swiss German](#). *CoRR*, abs/2103.11401.
- Paul Foulkes, Gerard Docherty, and Dominic Watt. 2005. [Phonological variation in child-directed speech](#). *Language*, 81(1):177–206.
- Fred Genesee, Elena Nicoladis, and Johanne Paradis. 2004. [Chilides French–English gnp corpus](#). CHILDES Database. French–English bilingual children in Montreal. Corpus date: 2004-03-30.
- Cornelia Hamann, Sharon Ohayon, Sophie Dubé, Ulrich H. Frauenfelder, Luigi Rizzi, Michal Starke, and Pascal Zesiger. 2003. Aspects of grammatical development in young French children with sli. *Developmental Science*, 6(2):151–158.
- Xuenan Hu, Qi Zhang, Liner Yang, Bin Gu, and Xun Xu. 2020. [Data augmentation for code-switch language modeling by fusing multiple text generation methods](#). In *Proceedings of Interspeech 2020*, pages 1062–1066.
- Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020. [GLUECoS: An evaluation benchmark for code-switched NLP](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3575–3585, Online. Association for Computational Linguistics.
- Houssam Eddine-Othman Lachemat, Abbas Akli, Nourredine Oukas, Yassine El Kheir, Samia Haboussi, and Shammur Absar Chowdhury. 2025. [CAFE: Spontaneous code-switching speech dataset in Algerian dialect, French and English](#). *Data in Brief*, 63:112150.
- Brian MacWhinney. 2000. *The CHILDES Project: Tools for Analyzing Talk: The Database*, 3rd edition. Lawrence Erlbaum Associates, Mahwah, NJ.
- Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Thamar Solorio. 2016. [Overview for the second shared task on language identification in code-switched data](#). In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49, Austin, Texas. Association for Computational Linguistics.
- Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from <https://github.com/chakki-works/seqeval>.
- Johanne Paradis and Fred Genesee. 1996. [Syntactic acquisition in bilingual children: Autonomous or interdependent?](#) *Studies in Second Language Acquisition*, 18(1):1–25.
- Elena Pizzuto. 2004. [CHILDES Italian Roma Corpus](#). CHILDES Database. Longitudinal study of a single Italian child. Corpus date: 2004-04-02.
- Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali, and Chandra Shekhar Maddila. 2017. [Estimating code-switching on Twitter with a novel generalized word-level language detection technique](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1971–1982, Vancouver, Canada. Association for Computational Linguistics.

- Emily Smolak, Stephanie de Anda, Bianca Enriquez, Diane Poulin-Dubois, and Margaret Friend. 2020. [Code-switching in young bilingual toddlers: A longitudinal, cross-language investigation](#). *Bilingualism: Language and Cognition*, 23(3):500–518.
- Igor Sterner and Simone Teufel. 2025. [Code-switching and syntax: A large-scale experiment](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11526–11533, Vienna, Austria. Association for Computational Linguistics.
- Elaine L. Stine and John N. III Bohannon. 1983. Imitations, interactions, and language acquisition. *Journal of Child Language*, 10(3):589–603.
- Livia Tonelli. 2004. [ChilDES Italian Tonelli corpus](#). CHILDES Database. Transcripts from three Italian children. Corpus date: 2004-04-01.
- Jannis Vamvas, Johannes Graën, and Rico Sennrich. 2023. [SwissBERT: The multilingual language model for Switzerland](#). In *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 54–69, Neuchâtel, Switzerland. Association for Computational Linguistics.
- Charles Watkins. 2004. [ChilDES French–English Watkins corpus](#). CHILDES Database. French–English bilingual children studied for their use of deixis. Corpus created at Université de Paris XIII.
- Genta Winata, Alham Fikri Aji, Zheng Xin Yong, and Thamar Solorio. 2023. [The decades progress on code-switching research in NLP: A systematic survey on trends and challenges](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2936–2978, Toronto, Canada. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Shijie Wu and Mark Dredze. 2020. [Are all languages created equal in multilingual BERT?](#) In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Which Skills Debate Reaches the Public? Comparing Scientific Literature and Media Coverage of AI and LLM Skill Impacts (2022–2025)

Maud Reveilhac¹, Gerold Schneider², Bohdan Trembovelskyi³
Nisha Yadav¹, Vlada Druta³, Aurelia Tamò-Larrieux³, Simon Mayer⁴
Clement Guitton⁴, Joshua C. Yang⁵

¹LUT University (Finland); ²University of Zurich (Switzerland); ³University of Lausanne (Switzerland);

⁴University of St-Gallen (Switzerland); ⁵ETH Zurich (Switzerland)

Correspondence: maud.reveilhac@lut.fi

Abstract

As large language models (LLMs) rapidly enter classrooms and workplaces, the public discourse about their impact on human skills has significant consequences for public understanding, policy, and education. We analyze this discourse through a comparison of the scientific literature and Swiss multilingual media coverage in German and French on how LLMs affect skills in education and work from 2022 to 2025. Using a reproducible pipeline that combines conceptual mapping and BERTopic modeling and proposing and implementing a tripartite analytical framework – amplified, simplified, absent – for examining how scientific framings are recontextualized in public discourse, we find a sharp divergence in thematic structure. We use these three categories to show that media coverage amplifies concrete use cases, simplifies differentiated scientific debates about skills, and leaves several central scientific themes less visible. In education reviews, a single pedagogical topic accounts for the majority of the corpus; in workplace reviews, it remains dominant at 51.28%. In Swiss media, however, the dominant topic is a broad and generalized AI-skills discourse (52.99%), while the education-centered topic accounts for only 1.52%. Conceptual maps show in greater detail that media coverage foregrounds AI capabilities, job loss, and replacement, while giving limited attention to themes central in the literature, including AI literacy, reflective use, metacognition, and pedagogical integration.

1 Introduction

Since the public release of ChatGPT in late 2022, LLMs have become central to debates about education, work, and the future of human competence. In education, recent reviews describe a rapidly expanding research field that links LLM use to writing, feedback, assessment, AI literacy, critical thinking, and broader pedagogical change (Kasneci et al., 2023; Yan et al., 2024; Giannakos et al.,

2025). In workplace research, LLMs are discussed both as productivity-enhancing tools and as drivers of changing task structures, skill requirements, and job quality (Brynjolfsson et al., 2025; Gmyrek et al., 2023; Salari et al., 2025).

Yet, scientific knowledge does not enter the public sphere directly. For most citizens, emerging technologies are encountered through media discourse rather than through review articles or research syntheses. Media therefore matters not as a measure of public opinion itself, but as a site where issues are framed, simplified, dramatized, or normalized, thereby shaping the interpretive resources available to citizens when they form opinions about emerging technologies. Research on public attitudes toward AI shows that media use is associated with the kinds of interpretive frames people adopt when thinking about AI, while deliberative research suggests that public support for AI-related applications depends strongly on how issues are explained and contextualized (Brewer et al., 2022; Arnesen et al., 2025).

This matters especially for the notion of skills. In the scientific literature, “skills” is not a single, stable category. It can refer to cognitive skills, professional competences, (AI) literacy, metacognitive regulation, communication abilities, or new forms of human–AI collaboration (Acemoglu and Autor, 2011; Zhang et al., 2025; Sidra and Mason, 2024). In the public discourse, however, these distinctions may be compressed into narratives about replacement, efficiency, or the need to “adapt”.

We thus ask: which dimensions of skill transformation associated with LLMs in scientific literature reviews are amplified, simplified, or absent in the discourse on public media? To answer, we compare two corpora from the same period, 2022–2025: (1) literature reviews in English on the effects of LLMs on skills in education and workplace domains, and (2) Swiss German- and French-language media articles retrieved from Swissdox.

This paper makes four contributions. *Empirically*, we provide a comparison of scientific and media discourse on LLMs and skills in a multilingual Swiss context, drawing on 246 English-language review documents and 4,610 German- and French-language news passages from Swissdox covering the 2022–2025 period. *Methodologically*, we develop a reproducible pipeline that combines BERTopic modeling with conceptual mapping via a translation-based pivot, enabling cross-arena and cross-lingual comparison of how the same problem space is structured in different discursive settings. *Conceptually*, we propose a tripartite analytical framework - *amplified, simplified, absent* - for examining how scientific framings of skill transformation are recontextualized in public discourse. *Theoretically*, we argue that skill transformation due to AI should be treated not only as an educational or labor-market issue, but as a public-opinion issue.

2 Theoretical background

2.1 Skills, tasks, and technological change

To discuss expected skill transformation through LLMs and their use, we start by introducing a distinction between tasks, skills, and technologies. Acemoglu and Autor (2011) argue that technological change does not mainly replace occupations; rather it reorganizes task structures and alters the value of particular skills. This perspective remains relevant for LLMs, which intervene directly in language-based and knowledge-intensive tasks.

Recent research on generative AI extends this view in two directions. First, workplace studies emphasize that LLMs might augment performance unevenly, often benefiting less experienced workers while also reshaping expectations of expertise, autonomy, and evaluation (Brynjolfsson et al., 2025; Woodruff et al., 2024). Second, education research stresses that the relevant outcome is not simply performance improvement, but the transformation of competences such as writing, reflection, prompting, critical thinking, and digital literacy (Kasneci et al., 2023; Yan et al., 2024; Daniel et al., 2025).

This literature also shows that LLM-related skill change is normatively ambivalent. Some studies highlight productivity gains, scaffolding, personalization, and support for higher-order learning, while others warn against overreliance, cognitive offloading, shallow engagement, and unequal access to AI competence (Giannakos et al., 2025;

Wang and Fan, 2025; Wiczorek et al., 2025).

2.2 Framing, mediation, and public understanding

In this paper, we draw on a science-communication view of media as a site of recontextualization rather than mere transmission. Scientific reviews synthesize evidence and stabilize conceptual distinctions; journalism selects, condenses, and narrativizes. As a result, the issue is not whether media “covers” the same material as the literature in a one-to-one way, but how scientific problem definitions are transformed into public-facing frames.

Brewer et al. (2022) show that public attitudes toward AI are associated with different “frames in mind,” including optimistic and risk-oriented interpretations, and that news use predicts these framings. Arnesen et al. (2025) similarly show that expressed AI support is sensitive to deliberation and knowledge conditions, which underscores the importance of information quality for democratic opinion formation.

In the specific case of LLMs, one likely mechanism is product-centric framing: public discourse often treats a highly visible application such as ChatGPT as shorthand for AI more generally. From the standpoint of skills discourse, this matters because product-centered coverage may foreground spectacle, capability, and disruption while backgrounding slower-moving questions about pedagogy, competence formation, metacognition, or professional practice. Deeper scientific grounding of relevant public discourse is further hindered by polarization tendencies of several public media, specifically on social media platforms (cf. (Lorenz-Spreen et al., 2023; Nguyen and Hekman, 2024)).

2.3 Skills as a public issue

This leads to this paper’s core theoretical positioning: We argue that skill transformation should not be treated only as educational or labor-market issue, but as a public-opinion one. If a framed media discourse selectively highlights some skill dimensions and neglects others, the public debate around LLMs may be structured by partial visibility. For example, “job loss” may become publicly salient while “AI literacy” or “reflective human–AI collaboration” remains largely scientific vocabulary. The literature on AI literacy already suggests that the concept itself is heterogeneous and evolving, spanning technical understanding, evaluation, ethics, and metacognitive awareness (Zhang et al., 2025;

Sidra and Mason, 2024).

This positioning has two implications for our study. *Theoretically*, it reframes skill transformation as an object of public-opinion research rather than only an educational or labor-market concern: what counts as a “skill at risk” or a “skill to acquire” is partly constituted by the discursive frames through which the issue becomes publicly visible. *Empirically*, it motivates a comparative design in which the same problem space – the intersection of LLMs and skills – is examined across two arenas (scientific synthesis and public-facing media) in order to identify which dimensions travel, which are simplified, and which remain largely invisible.

3 Methods

3.1 Corpus design

The study compares two corpora covering the same time period (2022–2025): a corpus of scientific literature reviews in English and a corpus of bilingual Swiss news media articles (French and German). The comparison aims to analyze how the concept of skill transformation due to LLMs is represented across different discourse arenas.

The first corpus consists of English literature reviews on LLMs and skills in two domains: education and the workplace. Literature reviews were selected because they synthesize multiple empirical studies and therefore provide a condensed representation of the scientific debate. As such, they serve as a useful reference space for identifying the main dimensions through which LLM-related skill transformation is conceptualized in research. The literature corpus was collected using two domain-specific queries combining AI and skill. The exact query is in the Appendix A1. The query does not search only for ChatGPT or GPT, but includes broader generic terms for AI, generative AI, and large language models in English, German, and French. Nevertheless, because the terms GPT/ChatGPT are highly visible in public discourse, their inclusion may increase the visibility of OpenAI-related coverage.

The second corpus consists of Swiss news media articles retrieved from the Swissdox database.¹ The retrieval logic was aligned conceptually with the literature corpus by combining LLM-related and skill-related terms. The query includes English, German, and French variants to capture multilingual coverage, see the exact query in the Appendix

¹See <https://swissdox.ch/> (last accessed: 30.04.2026)

A1. In the Swissdox query language, this logic was implemented using token and sequence matching for multilingual expressions of both AI technologies and skill-related concepts. The extraction initially returned 6062 German and 3478 French articles. To focus on relevant passages only, and also because BERTopics has limited context, we chunk the articles into passages as follows: After every 200 tokens we split at the next full stop. The 200-token threshold was chosen as a pragmatic compromise: it preserves enough local context for semantic embedding while avoiding very long passages that may contain several unrelated article sections. This procedure does not fully preserve journalistic structure such as headlines, leads, quotations, or section boundaries. We therefore interpret the resulting units as local thematic passages rather than complete articles. Only news passages that corresponded to the search query were kept. This led to 2915 German and 1695 French, in total 4610 passages from Swissdox. This design does not eliminate genre differences, but it ensures that both datasets are sampled from the same semantic problem space: the intersection of AI and skills. The English literature reviews’ titles and abstracts are shorter, so we did not need to chunk them. The literature corpus contains 78 work- and 168 education-related documents.

3.2 Analytical logic

The analytical logic is asymmetric. The literature-review corpus is not treated as a gold standard in any normative sense, but as a structured representation of the scientific conceptual field. The media corpus is treated as the public mediation field in which some of these dimensions may be amplified, generalized, displaced, or omitted. This means the study is not asking whether the media reproduces the literature faithfully. Instead, it asks how the skill dimensions present in scientific discourse are recontextualized when they become part of public communication.

3.3 Pre-processing and representation

Swissdox XML files were converted into structured text data through a Python–R pipeline. Media texts in German and French were normalized for multilingual comparison with the English-language review corpus. We used the multilingual model paraphrase-multilingual-mpnet-base-v2 for

Topic Modeling (Reimers and Gurevych, 2019)². Because the analysis focuses on conceptual proximity rather than quotation-level interpretation, semantic text representations are especially important. Embedding-based approaches are appropriate here because they are able to capture latent patterns or semantic similarity beyond simple word overlap, which is crucial when comparing journalistic and scientific language that may refer to similar issues with different lexical choices (Reimers and Gurevych, 2019). We have also used a translated version (see below) and obtained similar Topic Modeling results.

3.4 Conceptual mapping

From this structured text data, we constructed a visual semantic map by extracting salient terms from the corpora and positioning them in two-dimensional space according to their co-occurrence proximity (i.e., terms that frequently appear together are also rendered closer to one another). The result is a graph in which clusters correspond to conceptual neighborhoods, and the distance between nodes reflects the degree of lexical association. This enables an exploratory view of which concepts are corpus-specific, which are shared, and which form bridging zones between scientific and media discourse. This step follows a broader text-as-data approach in which texts are analyzed not only as containers of content but as structured signals of social and communicative processes (Stewart et al., 2022). It is especially useful here because the goal is to identify dimensions of framing rather than just keyword frequencies.

As this method does not support multilingual embeddings, we translated the passages automatically to English using OPUS-MT (Tiedemann et al., 2023)³. We then calculate term-term matrices with *textplot* (McClure, 2016). Finally we use the spring attraction algorithm *ForceAtlas2* (Jacomy et al., 2014) in the graph visualizer *gephi*.

3.5 Topic modeling

To validate and refine the conceptual map, we use topic modeling, with BERTopic as the main modeling framework. BERTopic is well suited to heterogeneous corpora because it combines

transformer-based embeddings, dimensionality reduction, clustering, and class-based TF-IDF topic representations (Grootendorst, 2022; Alammari and Grootendorst, 2024). Alternative topic-modeling paradigms such as Top2Vec provide useful methodological points of comparison because they also rely on semantic embeddings rather than purely bag-of-words assumptions (Angelov, 2020).

Topic modeling serves two purposes in our design. First, it checks whether the conceptual distinctions observed in the map persist at the level of document clusters. Second, it helps operationalize the paper’s core categories: i) *amplified*: themes proportionally more prominent in media than in the review corpus; ii) *simplified*: themes present in both corpora but realized through less differentiated vocabulary or narrower narrative associations in media; iii) *absent*: themes clearly established in the review corpus but weakly represented or missing in media.

3.6 Reproducibility

The study is conducted with aligned query logic, explicit corpus boundaries, scripted preprocessing, embedding-based semantic analysis, and topic-model validation. This is also relevant to the SwissText 2026 theme of reproducible NLP, because comparative discourse analysis can otherwise become difficult to audit when corpus construction and preprocessing decisions remain implicit.

4 Results

To ensure analytical clarity, the results are presented in two strictly separated parts: (1) topic modeling across corpora and (2) temporal-semantic patterns in the media corpus. No cross-references are made between these analytical layers.

4.1 Topic modeling across corpora

This subsection reports how the main BERTopic clusters are distributed across the three corpora and how they should be interpreted in relation to one another. We first clarify the topic labels, then compare their relative prominence across scientific education reviews, scientific workplace reviews, and Swiss media passages. In BERTopic, Topic -1 is a standard outlier or residual category: it contains documents that the clustering procedure does not assign to a more coherent topic. We therefore do not interpret Topic -1 as a substantive topic in the same way as Topics 0, 1, 2, or 3, but as an indicator of thematic heterogeneity and weak clustering.

²<https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2>

³<https://huggingface.co/Helsinki-NLP/opus-mt-roa-en> for French and Helsinki-NLP/opus-mt-de-en for German to English

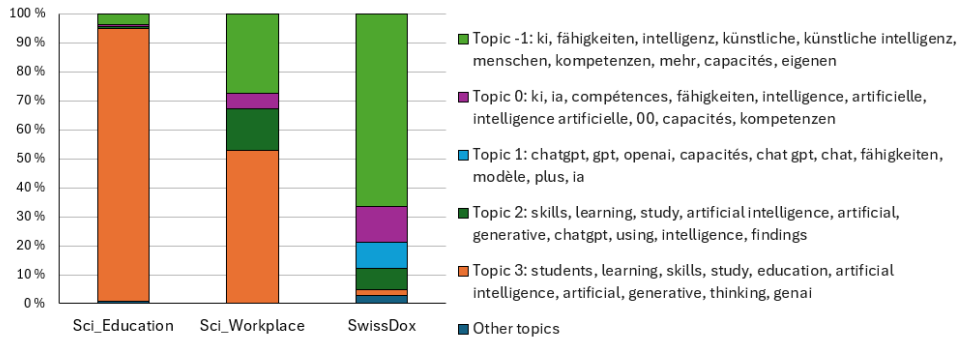


Figure 1: Distribution of the main topics across corpora.

Topic 3 is the education-centered topic, characterized by terms such as students, learning, skills, and education. Topic 0 captures a broad AI-capabilities discourse, often centered on what AI systems can do and how surprising these abilities appear. Topic 1 and Topic 2 capture more applied or practice-oriented discussions, including uses, tasks, competences, and work-related implications. Topic -1, as noted above, is the residual category.

We first analyze how topics are distributed across the three corpora. In the scientific education corpus (*Sci_Education* in Figure 1), the distribution is highly concentrated: Topic 3 (students, learning, skills, education) accounts for the majority of all content. This indicates a unified and pedagogically anchored discourse, with minimal contribution from other topics. In the scientific workplace corpus (*Sci_Workplace*), the distribution is more heterogeneous. Topic 3 remains dominant (51.28%), but Topic -1 (26.92%) and Topic 2 (14.10%) contribute substantially. This reflects a broader thematic scope that combines general AI discourse with applied and practice-oriented perspectives on skills. In contrast, the media corpus (*SwissDox*) is markedly fragmented. Topic -1 dominates with 52.99%, followed by Topic 0 (9.70%), Topic 1 (7.22%), and Topic 2 (6.01%). Topic 3, i.e., the education-centered topic, accounts for only 1.52%, indicating that pedagogical perspectives are largely absent from media coverage. Topics -1 to 3 cover the vast mass of the textual data, but it is insightful to explore further topics, see the Appendix A2.

4.2 Temporal patterns in the media corpus

We now analyze the conceptual map of the media corpus over time (Figure 2). In this analysis, the publication year (2022–2025) functions as a meta-variable for comparing corpus slices and structur-

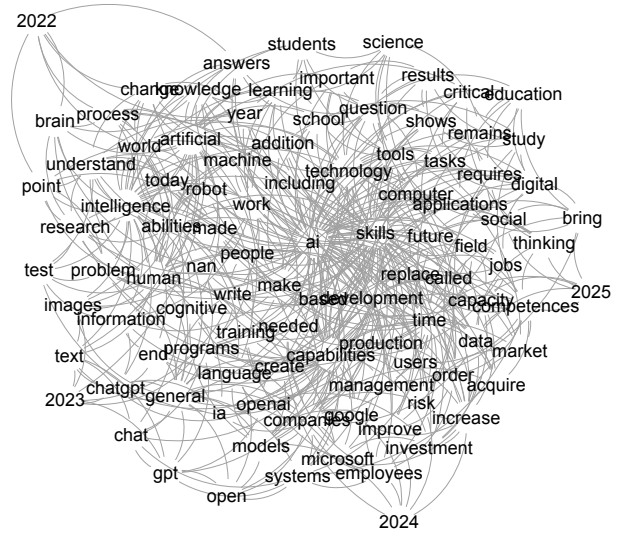


Figure 2: Conceptual map showing German and French translated news articles over time (2022–2025).

ing the semantic space. Figure 2 shows that media vocabulary shifts from early technological novelty in the year 2022 toward later labor-market and societal implications.

The map shows a clear temporal progression of themes. In 2022, the full form "artificial intelligence" is still used frequently, and the hope that we can "understand" the "brain" due to AI "research" (terms appearing in the maps are in straight quotes). In 2022–2023, the discourse is centered on technological emergence and proliferation of use, with clusters around terms such as "chatgpt," "models," "language," and "openai". The "abilities" of AI are surprising. This phase reflects the introduction and initial public engagement with generative AI.

In 2023–2024, the semantics shift toward use and application, with terms such as "tools," "applications," "tasks," "skills," and "users" becoming central. This indicates a transition from novelty to practical experimentation and integration.

In 2024–2025, the discourse turns more oriented toward societal and economic implications, with prominent clusters including "jobs," "market," "risk," "investment," and "competences." The debate focusses on labor-market transformation and future relevance of skills, particularly in 2025, where "social" impact, "jobs" and the question if AI will "replace" us moves into the center, and also which "skills" are required.

Across these phases, we have identified three distinct patterns in this progression:

First, we observe that the media discourse evolves from abstract technological concepts to more concrete socio-economic concerns. A first pattern is therefore *amplification through concrete use cases*. By this we mean that media coverage tends to make LLM-related skill transformation visible through recognizable examples: specific professions, companies, products, risks, public figures, or dramatic scenarios. This concreteness makes the issue more accessible, but it also amplifies those aspects of the debate that are easy to narrate, such as job loss, replacement, or spectacular AI capabilities. The media discourse appears to amplify the visible and dramatizing dimensions of skill transformation, especially those that can be narrated through recognizable actors, professions, or risks. This amplification intensifies in the later period (2024–2025), where labor-market and risk-related terms become more prominent.

The second pattern is *simplification through lexical compression*. Preliminary results indicate that scientific distinctions among skills, competences, literacy, reflection, and higher-order thinking are often collapsed in media discourse into broader and less specialized categories. Within the semantic map, this is reflected in densely connected clusters where heterogeneous terms co-occur without forming clearly separated conceptual domains. In this sense, the media does not simply omit scientific vocabulary; it often translates differentiated scientific debates into broader narratives about adaptation, employability, or disruption.

A third pattern is *absence or weak visibility*. Several dimensions that are prominent in the reviewed scientific literature currently seem underrepresented in media discourse: AI literacy as a structured competence, metacognitive awareness, reflective use of AI, pedagogical integration, and nuanced accounts of augmentation versus substitution. These dimensions do not appear as distinct or central clusters in the semantic structure of the

media corpus. We interpret this as weak public visibility: these concepts may appear occasionally, yet they do not organize the media discourse in the way they organize parts of the scientific literature.

These observations remain preliminary and will need to be tested against topic-model outputs and robustness checks, but they already point to a significant difference between the scientific conceptualization of skill transformation and the public framing available through media discourse.

4.3 Semantic patterns among the corpora including the meta-information

The semantic maps including the different corpora (see Figure 3 in Appendix A3) visualizes a dense, interconnected landscape centered on the transformative impact of artificial intelligence, specifically generative models, on education and the professional world, in distinction to the Swissdax news passages (see "French" and "German" on the left). The news are framing "people" and persons (e.g., Sam Altman and Lukas Walker), express surprise about the "abilities" of the models, often shortcutting methods and frameworks to individual products and companies ("gpt", "openai"). There are aspects relating to the local settings, "swiss" is of course on the side of Swiss news, while scientific concepts like "experimental," "methods," "significant," and "effects" are due to scientific practice.

The core of the map is dominated by technological terms such as "AI," "language models," "automation," and "code," which radiate outward to touch upon critical societal domains. Technological concepts do not exist in isolation; their proximity and repeated connections to terms such as "learning," "teaching," "students," "teachers," and "workplace" indicate that the mapped discourse frequently connects AI technologies with educational and professional activities. The graph indicates a research field that is actively mapping how tools like chatbots and LMMs are reshaping pedagogical methods, assessment strategies, and the very nature of "skills" required in the modern economy. The fact that "skills," "research" and "university" are central in the map indicates that also news media is reporting on the scientific discussion, and universities are seen in a bridging function.

A big portion of the map is dedicated to ethical and practical challenges arising from this integration. Terms like "ethical," "risk," "concerns," and "responsibly" are woven throughout the connections between AI and human systems, highlighting

a pervasive concern for the safe and fair deployment of these technologies. The map suggests that the conversation is not only about efficiency or innovation but equally on potentials for harm, the need for regulation, and the preservation of human agency. This is emphasized by the presence of words like “security,” “replace,” “affect” and “lack”, close to the constant, ambiguous need for “change,” “innovation” and “automation,” counterweights to more optimistic terms like “support,” “facilitate,” “promote.” All these terms are between media and science, but the positive ones closer to science, the negative ones closer to media.

The specific highlighting of “sciwork” and “sci-edu” points to a specialized focus on scientific work and science education, implying that the data underlying this map may originate from academic literature or research projects specifically targeting these domains. The dense clustering around “literature,” “research,” “study,” and “framework” confirms that this visualization likely represents a synthesis of scholarly inquiry rather than general public opinion. The intricate web of connections between “methods,” “evaluation,” “assessment,” and “feedback” suggests a rigorous approach to understanding how AI can be integrated into educational curricula and professional development without compromising academic integrity or professional standards. Ultimately, the map portrays a complex, evolving ecosystem where technology, education, and work are inextricably linked, requiring continuous adaptation and careful consideration of ethical implications.

5 Discussion

The discussion follows the three main findings identified above: amplification through concrete use cases, simplification of the science, and omission of key scientific themes. The preliminary findings support a view of media discourse not as a diluted copy of science, but as a selective reframing of scientific and technological developments for public communication. The analysis does not directly measure public opinion; rather, it identifies the media frames through which members of the public are likely to encounter debates about LLM-related skills.

5.1 Amplification through concrete use cases

First, media discourse amplifies LLM-related skill transformation by making it concrete through rec-

ognizable use cases, actors, products, professions, and risks. What becomes publicly salient is not the full conceptual architecture of the literature, but a narrower subset of dimensions that are easier to narrate: spectacular capability, labor-market disruption, expertise, and competition. This concreteness makes an abstract technological debate accessible, but it also gives disproportionate visibility to dramatic or easily personalized examples, such as ChatGPT, prominent companies, job loss, and replacement scenarios.

5.2 Simplification of the science

Second, media discourse simplifies the science by compressing differentiated concepts into broader and less specialized categories. This is visible in the media’s tendency to collapse distinct pedagogical concepts into single, high-frequency nodes, whereas the scientific networks maintain clearer separations between “learning,” “assessment,” “literacy,” and related concepts. This matters because the scientific literature treats skill transformation as a complex process involving learning design, literacy, metacognition, ethical judgment, and evolving professional competence (Yan et al., 2024; Zhang et al., 2025; Clear et al., 2025). In the scientific maps, terms such as “ethical,” “concerns,” and “responsible” connect AI to human agency; in the media corpus, these distinctions are more often absorbed into generalized narratives about adaptation, risk, disruption, or employability. Deskilling therefore tends to appear in media as a broad anxiety frame, while the literature differentiates between competences that may atrophy under overreliance and those that may become newly important, including prompt design, source evaluation, reflective monitoring, and human–AI coordination.

5.3 Omission of key scientific themes

Third, several key themes from the scientific literature are weakly visible or largely absent in media discourse. These include AI literacy as a structured competence, metacognitive awareness, reflective use of AI, pedagogical integration, and nuanced accounts of augmentation rather than simple substitution. Their weak visibility matters because these concepts would help the public understand what skillful adaptation to LLMs entails.

5.4 Implications for science communication research

These findings extend existing work on AI framing and public attitudes (Brewer et al., 2022; Arnesen et al., 2025) in three ways. First, the analysis quantifies the asymmetry between scientific and public framings within the same semantic problem space: a single pedagogical topic accounts for the majority of the education-review corpus but only 1.52% of the media corpus. Second, the temporal analysis suggests that media framing evolves from technological novelty (2022–2023), through use and application (2023–2024), to labor-market and societal implications (2024–2025), providing an empirical reference point for studies of how emerging-technology discourse stabilizes over time. Third, the persistent synecdochal use of “ChatGPT” as shorthand for AI more generally indicates a product-centered framing that may shape policy and educational debates by tying abstract questions about competence to a single commercial artifact.

From a public-opinion perspective, this is consequential. If citizens encounter LLM-related skills mainly through product-centric and disruption-oriented narratives, democratic debate about education and work may be shaped by a partial repertoire of interpretations. Rather than encountering the full range of scientific concepts used to understand learning, work, and human–AI collaboration, the public is more likely to encounter a narrower discourse centered on capability, disruption, and risk. Methodologically, the study shows the value of comparing media and review literature through semantic and topic-based methods rather than direct content matching: the goal is not literal equivalence, but a structured account of recontextualization across discourse arenas.

6 Conclusion

This study shows that scientific and media discourses construct different public meanings of LLM-related skill transformation. While the scientific literature emphasizes differentiated questions of AI literacy, cognitive skills, ethical control, metacognition, and human–AI collaboration (Brynjolfsson et al., 2025; Wang and Fan, 2025; Zhang et al., 2025), the Swiss media corpus more often foregrounds concrete and product-centered narratives of capability, disruption, replacement, and economic risk. Media discourse therefore does not simply transmit scientific knowledge; it recontextu-

alizes it into public-facing frames that shape which skill futures become publicly visible (Brewer et al., 2022; Arnesen et al., 2025).

Our main contribution is to make this recontextualization empirically visible through the framework of amplification, simplification, and absence. Media coverage amplifies concrete use cases, visible actors, and dramatic risks; simplifies scientific distinctions among skills, competences, literacy, reflection, and professional adaptation; and leaves key scientific themes such as pedagogical integration, metacognitive regulation, and reflective human–AI collaboration weakly visible. The frequent use of “ChatGPT” as shorthand for AI further reinforces a product-centered understanding.

Practically, the findings suggest that public debate about AI and skills may become reactive if it is organized mainly around job loss, technological spectacle, and replacement. Such a framing risks narrowing democratic deliberation and policy imagination: it makes risks visible, but less often shows how skills can be developed, protected, or transformed. More balanced science communication should therefore make visible not only disruption, but also the forms of learning, judgment, AI literacy, and human–AI coordination needed for skillful adaptation in education and work.

The study remains limited by corpus comparability, multilingual semantic alignment, its focus on reviews and Swiss media, and the absence of direct public-opinion or behavioral measures. The year-based analysis also does not yet use formal temporal models such as dynamic topic modeling or change-point detection. Future research should extend the comparison to empirical studies, policy documents, and additional media contexts, and should test how specific media frames shape public attitudes toward AI in education and work. It should also examine the conditions under which more differentiated scientific framings enter public discourse, and whether media coverage can move beyond the binary of “human versus machine” toward a more precise account of augmentation, adaptation, and responsible skill development.

Acknowledgments

The TA-SWISS Foundation is funding the project entitled "Sprachmodelle und Kompetenzverschiebungen," within the framework of which this study is being conducted (see: <https://www.ta-swiss.ch/grosse-sprachmodelle>).

References

- Daron Acemoglu and David Autor. 2011. **Skills, tasks and technologies: Implications for employment and earnings**. In David Card and Orley Ashenfelter, editors, *Handbook of Labor Economics*, volume 4, pages 1043–1171. Elsevier.
- Jay Alammar and Maarten Grootendorst. 2024. *Hands-On Large Language Models: Language Understanding and Generation*. O’Reilly Media.
- Dimo Angelov. 2020. **Top2Vec: Distributed representations of topics**. *Computing Research Repository*, arXiv:2008.09470.
- Svein Arnesen, Tanja S. Broderstad, James S. Fishkin, Mads P. Johannesson, and Alice Siu. 2025. **Knowledge and support for AI in the public sector: A deliberative poll experiment**. *AI & Society*, 40(5):3573–3589.
- Paul R. Brewer, John Bingaman, Albert Paintsil, David C. Wilson, and Will Dawson. 2022. **Media use, interpersonal communication, and attitudes toward artificial intelligence**. *Science Communication*, 44(5):632–658.
- Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. 2025. **Generative AI at work**. *The Quarterly Journal of Economics*, 140(2):889–942.
- Tony Clear, Åsa Cajander, Alison Clear, Roger McDermott, Mats Daniels, Anna Fjellkner, Arto Hellas, Päivi Kinnunen, Juho Leinonen, James Prather, Arnold Pears, Emma Saarinen, and Claudia Szabo. 2025. **AI integration in the IT professional workplace: A scoping review and interview study with implications for education and professional competencies**. In *Proceedings of the 2024 Working Group Reports on Innovation and Technology in Computer Science Education*, pages 34–67. Association for Computing Machinery.
- Kibona Daniel, Mussa M. Msambwa, and Zhe Wen. 2025. **Can generative AI revolutionise academic skills development in higher education? a systematic literature review**. *European Journal of Education*. Advance online publication.
- Michail Giannakos, Roger Azevedo, Peter Brusilovsky, Dragan Gašević, Inge Molenaar, Xavier Ochoa, and 1 others. 2025. **The promise and challenges of generative AI in education**. *Behaviour & Information Technology*, 44(11):2518–2544.
- Pawel Gmyrek, Janine Berg, and Daniel Bescond. 2023. **Generative AI and jobs: A global analysis of potential effects on job quantity and quality**. Working Paper 96, International Labour Organization.
- Maarten Grootendorst. 2022. **BERTopic: Neural topic modeling with a class-based TF-IDF procedure**. *Computing Research Repository*, arXiv:2203.05794.
- Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. 2014. **Forcelatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software**. *PLoS ONE*, 9(6):e98679.
- Enkelejda Kasneci, Katharina Sessler, Stefan Küchermann, Maria Bannert, Darya Dementieva, Frank Fischer, Ulrike Gasser, Georg Groh, Stephan Günemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tomer Michaeli, Christian Nerdel, Friedrich Pfeiffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, and 2 others. 2023. **ChatGPT for good? on opportunities and challenges of large language models for education**. *Learning and Individual Differences*, 103:102274.
- Philipp Lorenz-Spreen, Lisa Oswald, Stephan Lewandowsky, and Ralph Hertwig. 2023. **A systematic review of worldwide causal and correlational evidence on digital media and democracy**. *Nature Human Behaviour*, 7:74–101.
- David McClure. 2016. **textplot**. <https://github.com/davidmcclure/textplot>. GitHub repository.
- Dong Nguyen and Ekaterina Hekman. 2024. **The news framing of artificial intelligence: a critical exploration of how media discourses make sense of automation**. *AI & Society*, 39:437–451.
- Nils Reimers and Iryna Gurevych. 2019. **Sentence-BERT: Sentence embeddings using siamese BERT-networks**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992. Association for Computational Linguistics.
- Nader Salari, Mojtaba Beiromvand, Ali Hosseinian-Far, Masoumeh Mohammadi, Bahareh Fatahi, and Siti Shohaimi. 2025. **Impacts of generative artificial intelligence on the future of labor market: A systematic review**. *Smart Analytics, Artificial Intelligence and Sustainable Performance*, 1(1):100067.
- Sidra Sidra and Christine Mason. 2024. **Reconceptualizing AI literacy: The importance of metacognitive thinking in an artificial intelligence-enabled workforce**. In *2024 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*. IEEE.
- Brandon M. Stewart, Justin Grimmer, and Margaret E. Roberts. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press, Princeton, NJ.
- Jörg Tiedemann, Mikko Aulamo, Daria Bakshandaeva, Michele Boggia, Stig-Arne Grönroos, Tommi Nieminen, Alessandro Raganato Yves Scherrer, Raul Vazquez, and Sami Virpioja. 2023. **Democratizing neural machine translation with OPUS-MT**. *Language Resources and Evaluation*, (58):713–755.

- Jiaqi Wang and Wei Fan. 2025. The effect of ChatGPT on students' learning performance, learning perception, and higher-order thinking: Insights from a meta-analysis. *Humanities and Social Sciences Communications*, 12:744.
- Marta Wiczorek, Mohammad Hosseini, and Bert Gordijn. 2025. Unpacking the ethics of using AI in primary and secondary education: A systematic literature review. *AI and Ethics*. Advance online publication.
- Allison Woodruff, Renee Shelby, Patrick G. Kelley, Shiri Rousso-Schindler, Jeffrey Smith-Loud, and Lauren Wilcox. 2024. How knowledge workers think generative AI will (not) transform their industries. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–26. Association for Computing Machinery.
- Lixiang Yan, Lei Sha, Ling Zhao, Yi Li, Roberto Martinez-Maldonado, Guanliang Chen, Xiaoming Li, Yiqi Jin, and Dragan Gašević. 2024. Practical and ethical challenges of large language models in education: A systematic scoping review. *British Journal of Educational Technology*, 55(1):90–112.
- Shuo Zhang, P. Ganapathy Prasad, and Noah L. Schroeder. 2025. Learning about AI: A systematic review of reviews on AI literacy. *Journal of Educational Computing Research*. Advance online publication.

A Appendices

A.1 Search-queries used to retrieve the documents

For the education domain, the exact query was: ("large language models" OR "LLM" OR "LLMs" OR "AI" OR "GenAI" OR "artificial intelligence") AND ("skills" OR "competences" OR "upskilling" OR "up-skilling" OR "deskilling" OR "de-skilling") AND ("students" OR "pupils") AND ("education" OR "school"). For the workplace domain, the query was: ("large language models" OR "LLM" OR "LLMs" OR "AI" OR "GenAI" OR "artificial intelligence") AND ("skills" OR "competences" OR "upskilling" OR "up-skilling" OR "deskilling" OR "de-skilling") AND ("workplace" OR "job" OR "workforce" OR "labor" OR "labour" OR "market").

The conceptual query used for Swissdax corresponds to the following structure: ("LLM" OR "LLMs" OR "GPT" OR "AI" OR "KI" OR "IA" OR "GenAI" OR "large language models" OR "grosse Sprachmodelle" OR "grands modèles de langage" OR "artificial intelligence" OR "künstliche Intelligenz" OR "intelligence artificielle") AND ("skills" OR "compétences" OR "capacités" OR "Fähigkeiten" OR "Kompetenzen" OR "deskilling" OR "Entqualifizierung" OR "déqualification" OR "de-skilling" OR "De-Qualifizierung" OR "dé-qualification" OR "upskilling" OR "Weiterqualifizierung" OR "requalification" OR "upskilling" OR "Up-Skilling").

A.2 Description of the topics 4 to 10

Topics -1 to 3 cover the vast mass of the textual data, but it is insightful to explore further topics. The keywords of topics 4 to 10, followed by our interpretation and an excerpt from the most prototypical passage are:

Topic 4: ki, Or0, fähigkeiten, intelligenz, ia, menschliche, intelligence artificielle, artificielle, capacités, künstliche. Topic 4 focusses on how similar AI is to human intelligence. “Früher als gedacht könnten die Fähigkeiten von künstlicher Intelligenz denjenigen eines menschlichen Gehirns entsprechen.”

Topic 5: capacités, capacités calcul, calcul, ia, données, développement, microsoft, centres, ki fähigkeiten, plus. Topic 5 is about the computing centres and their energy consumption “ ... l’IA générative, qui nécessite des capacités de calcul colossales pour traiter les informations accu-

mulées”

Topic 6: lernen, müssen, wurden, kompetenzen, compétences, führt, denken, gpt, wichtiger, daran. Topic 6 takes up the discussion on skills needed in society, it is close to topic 3, but with concrete framing. “Diese Kompetenzen wurden schon immer gebraucht. Aber sie wurden wichtiger. Gerade jetzt, wo die künstliche Intelligenz hinzukommt, führt kein Weg daran vorbei, das Lernen an sich zu thematisieren”

Topic 7: capacités, scientifique, intelligence artificielle, artificielle, ki, Fähigkeiten, zurich, avancée, intelligence, dix. Topic 7 takes up the scientific discourse and fulfills a bridging function. “Une approche en collaboration avec de nombreuses institutions dont les Universités de Genève et Zurich et les deux Écoles polytechniques fédérales.”

Topic 8: gar, eigenen, Intelligenz, apple, co, kognitiven Fähigkeiten, kognitiven, weitere, chat-gpt, vielmehr. Topic 8 features philosophical discourses which are often absent in scientific literature. “Was uns bei ChatGPT als künstliche Intelligenz erscheint, ist demnach lediglich ein Spiegel, der unsere eigenen Wünsche, Bedürfnisse und kognitiven Fähigkeiten zurückwirft.”

Topic 9: Intelligenz, künstliche, künstliche Intelligenz, menschliche Fähigkeiten, imitieren, Fähigkeit, menschliche, 00, Fähigkeiten, Kreativität. Topic 9 seems as unspecific as Topic -1. What may distinguish it is its pedagogical focus. “Dieser Artikel stammt aus der Schweizer Familie Was genau ist künstliche Intelligenz?”

Topic 10: générale, sam, sam altman, altman, humaines, intelligence artificielle, artificielle, cognitives humaines, intelligence, artificielle générale. Topic 10 shows the personalization which is typical for news reporting. “Son patron, Sam Altman, a récemment expliqué travailler désormais vers l’intelligence artificielle dite «générale» ...”

A.3 Additional conceptual maps

The Same Email, Signed Differently: Testing Negotiation Bias and Recommendation Stability in LLMs

Jasmin Heierli and Alexandre de Spindler

Zurich University of Applied Sciences, Winterthur, Switzerland
heej@zhaw.ch, desa@zhaw.ch

Abstract

Large language models (LLMs) are increasingly mediating hiring communications, serving both as tools for applicants to draft negotiation emails and as systems for recruiters to evaluate them. Such mediation risks introducing variability and hidden dependencies into high-stakes outcomes such as salary expectations and hiring decisions. This paper investigates how the outcomes of these bidirectional interactions, specifically salary expectations and hiring recommendations, are influenced by gender signaling, model identity, and language context. We study this setting with a two-stage analysis across models and English/German contexts, using 2,880 Stage 1 observations and 1,441 paired Stage 2 evaluations. We find no strong or consistent pooled gender effects. Instead, model differences dominate, while scalar ratings are stable and categorical recommendations are less robust.

1 Introduction

Large language models are rapidly becoming embedded in hiring workflows, not only as tools for evaluating candidates¹ but also as assistants that help applicants draft negotiation messages² (Chaturvedi and Chaturvedi, 2025). Rather than supporting isolated decisions, models now shape the full interaction loop, generating negotiation content and evaluating it, raising new questions about how such systems influence outcomes in hiring contexts (Geiger et al., 2025). This raises a concern: outcomes may reflect interacting model behaviors that are difficult to anticipate or control rather than transparent human judgment.

Most prior work studies LLMs in hiring as isolated decision-makers, focusing on static tasks

¹<https://www.herohunt.ai/blog/ai-adoption-in-recruiting-2025-year-in-review>

²<https://www.interviewpal.com/free-tools/salary-negotiation-email-generator>

such as ranking, screening, or salary assignment (Chaturvedi and Chaturvedi, 2025; Rozado, 2026; Li et al., 2025). This perspective overlooks a key aspect of real-world hiring: negotiation as an interactive process, where outcomes depend on how candidates articulate their expectations (Mazei et al., 2015). Although recent work has begun to explore more personalized uses of LLMs, such as career advice (Geiger et al., 2025; Eloundou et al., 2025), it is unclear how outcomes are shaped when LLMs mediate both sides of such a high-stakes interaction.

Recent work shows that LLM behavior in hiring can vary substantially across model providers and versions (Geiger et al., 2025). At the same time, bias patterns have been shown to vary between languages, for example, between English and German prompts (Ikae and Kurpicz-Briki, 2025). We are building on this line of studies using minimal identity perturbations, such as name-based gender swaps, demonstrating that even small changes in identity signals can affect model outputs (Rozado, 2026; Li et al., 2025). In this paper, we combine these perspectives by examining how model choice, language context, and decision format interact within a fully bidirectional, LLM-mediated negotiation setting.

We introduce a reproducible two-stage experimental pipeline in which models from three providers are used to generate salary negotiation emails and to evaluate them³. In Stage 1, we extract salary range midpoints and widths from generated negotiation responses (Geiger et al., 2025). In Stage 2, the same responses are evaluated under paired identity perturbations, isolating evaluation differences while holding content constant (Rozado, 2026; Li et al., 2025). Experiments are conducted in English (UK) and German

³https://github.com/zhaw-iwi/swisstext-gender-in-job-ads_pub

(Switzerland) across two professional roles.

We examine whether LLMs exhibit gender differences in generated salary expectations, whether identical negotiation emails are evaluated differently under minimal identity changes, and how model, language, and decision format shape these patterns. We contribute a two-stage generation–evaluation pipeline, a controlled English/German comparison, and a systematic analysis of model, language, and decision-format effects in LLM-mediated hiring interactions.

2 Method

2.1 Experimental Design

We used a two-stage pipeline built on controlled synthetic stimuli. In Stage 1, a model generated a short salary-negotiation response to a manually created recruiter template. In Stage 2, that response was evaluated twice under minimal identity perturbation: only the signature name was swapped, while the email body remained unchanged. This isolates generation from evaluation effects.

The design varies the language context, role, candidate name, and model. We used English for a UK market prompt and German for a Swiss-German market prompt. We focused on two roles, *Senior Data Analyst* and *Senior Project Manager*, each requiring 5–7 years of experience.

2.2 Stimuli and Prompt Setup

All Stage 1 prompts used the same manually created recruiter-message template for each language. The full text of these templates and the specific system instructions for each provider are provided in our repository. This template served as the stimulus, where the "recruiter" asked for a response with a realistic annual salary range, and the model was instructed to sign with the candidate's name. The task structure was constant between models; language, role, market, and candidate name varied.

Gender was signaled only through names. For each language context, we used four female and four male names. In Stage 2, each name had a fixed opposite-gender partner, and only the signature name was changed. The fixed mapping reduced variance and preserved a tightly controlled paired comparison.

2.3 Models

We included one current model from three major providers at execution time: OpenAI, Anthropic,

and Google. The runs used gpt-5.4 (resolved in the metadata as gpt-5.4-2026-03-05) (OpenAI, 2026), claude-sonnet-4-6 (Anthropic, 2026), and gemini-3.1-flash-lite-preview (Google, 2026). We used identical prompt templates and retained model default generation settings unless required by the API. Model identifiers and metadata were recorded for reproducibility and are stored in our GitHub repository.

2.4 Stage 1: Negotiation Generation Stage

In Stage 1, each model was instructed to write a brief professional response to a recruiter message and include a realistic annual salary range. The primary results were salary midpoint and range width, derived from the minimum and maximum values extracted. Extraction was performed using rule-based parsing with manual validation by the authors. The validation showed around 85% extraction accuracy, and errors were fixed manually to enable further processing of the data.

2.5 Stage 2: Synthetic Evaluation Stage

Stage 2 tested whether identical negotiation content is judged differently after a name swap. For each Stage 1 email, we created a paired variant by replacing only the candidate's signature name with its mapped opposite-gender counterpart. The models assessed the professionalism, likeability/warmth, and appropriateness of the salary request (1–7), plus a categorical recommendation (*Proceed*, *Proceed with caution*, *Do not proceed*). These assessments were conducted through separate inference runs for each variant to ensure no context overlap. This isolates within-model evaluation before introducing cross-model variability.

In the main analysis, the evaluation was performed by the same model: each model judged texts generated by that same model. We took this decision to reduce the complexity of the design in this preliminary study.

2.6 Analysis

For Stage 1, we compared midpoint and width under female versus male signaling while modeling language, role, and model. For Stage 2, we computed paired deltas for the three scalar ratings and the recommendation shift rate for the categorical decision. Inferential analyses used paired tests and regression as appropriate, with bootstrap confidence intervals. We applied the Holm correction within each outcome family.

Two design choices matter for interpretation. First, the English and German conditions should be read as language-context comparisons, not as national pay estimates. Second, the study prioritizes internal control over demographic realism: names serve as minimal identity cue and the fixed prompt suppresses real-world covariates.

3 Results

We analyze 2,880 Stage 1 observations and 1,441 paired Stage 2 observations. We find no consistent gender effects in generated salary targets or mean scalar evaluation scores. Instead, the clearest patterns are model differences in Stage 1 and decision-format differences in Stage 2: scalar ratings are stable on average under name swaps, while categorical recommendations are less robust.

3.1 Stage 1: Small Gender Effects

Pooling across models, roles, and prompts, the difference between the female and the male midpoint was GBP -128.472 in English and CHF -97.222 in German. Width differences were also small and inconsistent (GBP 20.833 in English and CHF 375.000 in German). In the pooled model, the gender effect was not significant for midpoint ($p = 0.565$, Holm $p = 0.860$) or width ($p = 0.430$, Holm $p = 0.860$).

3.2 Stage 1: Model Effects Dominate

Model differences were much larger than pooled gender gaps. In both language contexts, Claude produced the lowest midpoint means, GPT the highest, and Gemini lay between them. In English, model midpoint means ranged from roughly GBP $76.5k$ (Claude) to GBP $85.0k$ (GPT); in German, they ranged from roughly CHF $127.3k$ (Claude) to CHF $136.2k$ (GPT). Width differences also varied by model, with Gemini showing especially narrow ranges in the German conditions.

Controlled models support this pattern: relative to Claude, model terms for Gemini and GPT were significant for midpoint (both $p < 0.001$). Figure 1 shows that model variation exceeds gender gaps.

3.3 Stage 2: Stable Scalar Ratings

In Stage 2, the average effects of name-swapping on the three scalar evaluation dimensions were close to zero. Across all observation pairs, mean deltas were 0.014 for professionalism, 0.008 for likeability, and 0.024 for appropriateness. None of these changes were significant: professionalism

$p = 0.792$, likeability $p = 0.829$, and appropriateness $p = 0.635$ (all Holm-corrected $p = 1.000$). Scalar evaluations are therefore largely invariant to the name swap at the aggregate level.

3.4 Stage 2: Categorical Recommendations

The categorical recommendation outcome behaved differently. The recommendation class changed in 16.3% of the paired cases, and this shift rate was significant ($p = 0.009$, Holm $p = 0.036$). Thus, even though the mean scores for professionalism, likeability, and appropriateness remained near zero, the final decision category was more sensitive to minimal identity disturbance.

Recommendation instability was also model-dependent. Gemini had a shift rate of 2.9% , compared to 21.9% for Claude and 24.1% for GPT. In the controlled provider model, again using Claude as the omitted reference category, Gemini showed a strongly negative log-odds coefficient of recommendation-shift (-2.236 , $p < 0.001$), whereas GPT did not differ significantly (0.127 , $p = 0.407$). Figure 2 shows both the quasi-zero scalar deltas and the much wider spread in recommendation-shift rates.

3.5 Extreme Cases and Variation

The quasi-zero scalar averages conceal a tail of extreme reversals. 225 paired cases showed large simultaneous movement in all three scalar ratings ($|\Delta| \geq 3$ for professionalism, likeability, and appropriateness). These cases were concentrated almost entirely in GPT evaluations ($224/225$), with more in English (128) than in German (96). Directionally, these flips were not fully symmetric: the female-signaled version scored lower in 126 cases and higher in 99 cases.

Language differences were present, but mainly through model-language combinations rather than as a single uniform cross-lingual effect. Instability varied by model-language combination: Claude was less stable in German than English, GPT showed the opposite pattern, and Gemini remained comparatively stable in both. These patterns suggest that language effects are less consistent than model effects.

4 Discussion & Conclusion

Our findings suggest that LLM behavior in hiring-related tasks is not well captured by a single notion of bias. Although we do not observe strong or consistent pooled gender effects in generated salary

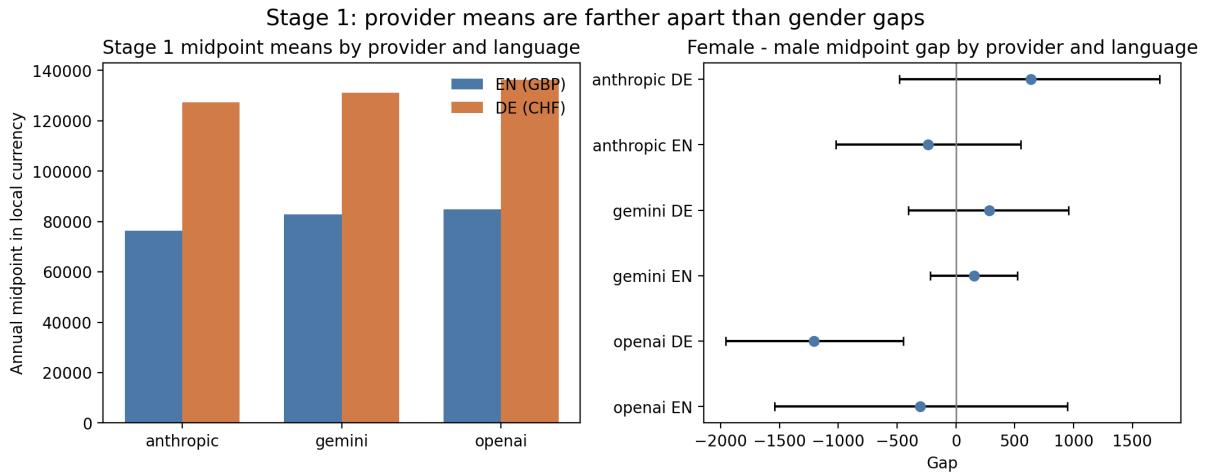


Figure 1: Stage 1 results. Left: model-level midpoint means by language context, shown in local currency (GBP for English prompts and CHF for German prompts). Right: female-minus-male midpoint gaps by model and language.

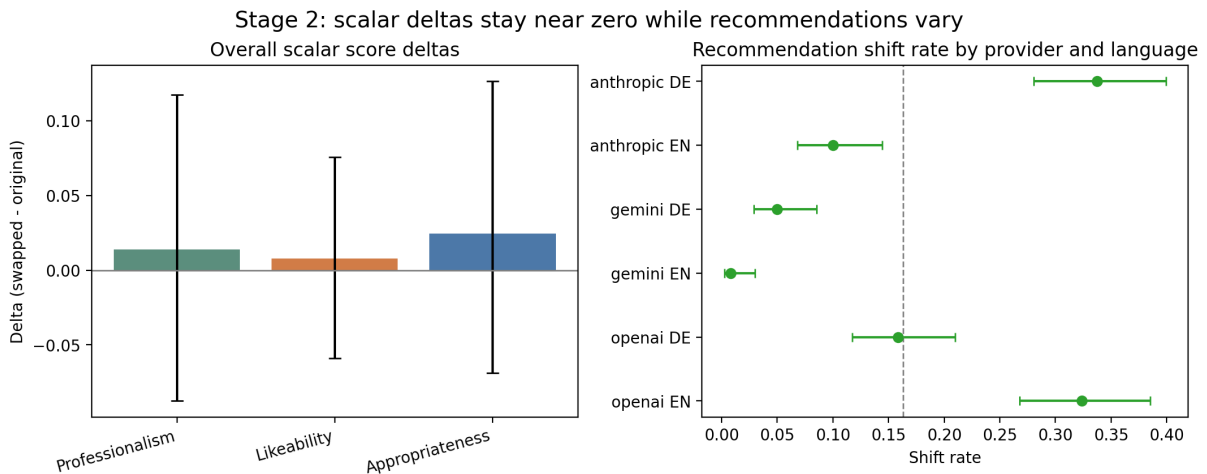


Figure 2: Stage 2 results. Left: overall score deltas for professionalism, likeability, and appropriateness. Right: recommendation-shift rates vary substantially across model-language conditions, with the dashed line marking the overall shift rate of 16.3%.

ranges or mean scalar evaluation scores, model outputs are not simply neutral or stable. Instead, they exhibit structured instability, with behavior depending on model, language context, and decision format. This implies vendor-dependent outcomes: salary expectations differ across models, reflecting system choice rather than candidate characteristics. Decision inconsistency emerges, as identical content can lead to different recommendation categories, and hidden instability arises, with scalar ratings appearing stable while final decisions remain sensitive to small perturbations.

Model differences dominate generation, making model choice a primary driver of salary expectations. In evaluation, scalar ratings remain stable under minimal identity perturbations, but this masks

a tail of extreme reversals. In contrast, categorical recommendations are substantially less stable, suggesting that output formats differ in operational reliability.

These results highlight the importance of viewing LLM use in hiring as an interaction between generation and evaluation, rather than isolated tasks. When models are used on both sides, differences in model, language, or output format may propagate, amplifying variability. Neither scalar nor categorical outputs should be treated as reliable for high-stakes decisions. Although scalar ratings appear more stable, model behavior is configuration-dependent. This highlights the need for reproducible, context-sensitive evaluation across models, languages, and decision formats.

Limitations

Gender was operationalized through names, which provides a minimal and naturalistic cue but does not isolate gender alone: names may also carry class, ethnicity, age, or other sociocultural associations. The observed effects should therefore be interpreted as responses to name-based gender signaling under controlled conditions rather than as clean estimates of gender bias in isolation.

The cross-lingual comparison is also a language-market comparison. English prompts were paired with a UK market context and German prompts with a Swiss-German market context because realistic salary negotiation depends on local compensation norms. This improves plausibility, but it means the design cannot disentangle linguistic effects from economic and cultural expectations. The cross-language results are therefore contextual rather than causal.

The design scope is narrow. We study only two relatively senior white-collar roles, so the findings may not generalize to other occupations, industries, or seniority levels. Likewise, the factorial design is better suited to detecting the main contrasts of interest than very small higher-order interactions.

Stage 2 uses same-model evaluation of model-generated text. This keeps the comparison tightly controlled but limits claims about whether the same patterns would hold under cross-model evaluation. More generally, the task remains a simplified proxy for real negotiation, relying entirely on researcher-authored stimuli and model-generated responses without human participants. Actual hiring communication is often multi-turn, includes richer identity cues, and involves strategic adaptation by both sides. Furthermore, this study focuses on the technical behavior of interacting models; however, the practical displacement of transparent, criteria-based human judgment by automated LLM-to-LLM interactions raises significant ethical questions regarding accountability and hidden dependencies in hiring.

Finally, the findings are tied to specific model snapshots. Commercial systems are updated frequently, and even small version changes may alter both generation style and evaluation behavior. The results should therefore be read as evidence about the tested systems at execution time, not as stable properties of future versions.

References

- Anthropic. 2026. [Introducing claude sonnet 4.6](#). Accessed: 2026-03-10.
- Sugat Chaturvedi and Rochana Chaturvedi. 2025. Who gets the callback? generative ai and gender bias. *arXiv preprint arXiv:2504.21400*.
- Tyna Eloundou, Alex Beutel, David Robinson, Keren Gu, Anna-Luisa Brakman, Pamela Mishkin, Meghan Shah, Johannes Heidecke, Lilian Weng, and Adam Tauman Kalai. 2025. [First-person fairness in chatbots](#). In *International Conference on Learning Representations*, pages 58234–58268.
- R. Stuart Geiger, Finn O’Sullivan, Edward Wang, and Jessica Lo. 2025. [Asking an ai for salary negotiation advice is a matter of concern: Controlled experimental perturbation of chatgpt for protected and non-protected group discrimination on a contextual task with no clear ground truth answers](#). *PLOS ONE*, 20(2).
- Google. 2026. [Gemini 3.1 flash-lite \(vorabversion\)](#). Google Cloud Documentation. Accessed: 2026-03-10.
- Chidubem Ikae and Marlena Kurpicz-Briki. 2025. [Measuring bias in german prompts to gpt models using contact hypothesis](#). In *Proceedings of the 2nd Workshop on AI Bias: Measurements, Mitigation, Explanation Strategies (AIMMES 2025)*.
- Miaomiao Li, Hao Chen, Yang Wang, Tingyuan Zhu, Weijia Zhang, Kaijie Zhu, Kam-Fai Wong, and Jindong Wang. 2025. [Understanding and mitigating the bias inheritance in llm-based data augmentation on downstream tasks](#). *arXiv preprint arXiv:2502.04419*.
- Jens Mazei, Joachim Hüffmeier, Philipp A. Freund, Alice F. Stuhlmacher, Lisa Bilke, and Guido Hertel. 2015. [A meta-analysis on gender differences in negotiation outcomes and their moderators](#). *Psychological Bulletin*, 141(1):85–104.
- OpenAI. 2026. [Entdecke gpt-5.4](#). Accessed: 2026-03-10.
- David Rozado. 2026. [Gender and positional biases in llm-based hiring decisions: evidence from comparative cv/résumé evaluations](#). *PeerJ Computer Science*, 12:e3628.

A Bounded Coordination-Support Capability for Multi-Party Settings: Task-State Monitoring in Firefighter Incident Command

David Grünert and Barbara Morgenstern and Florian Peczinka
and Dario Holenstein and Roland Brunner and Alexandre de Spindler

Zurich University of Applied Sciences, Winterthur, Switzerland
{grund, desa}@zhaw.ch, {morgebar, pecziflo, holendar}@students.zhaw.ch
Freiwillige Feuerwehr Stadt Zug, Zug, Switzerland
roland.brunner@stadtzug.ch

Abstract

Many collaboration settings require digital support systems for several humans who coordinate through ongoing communication. We study one such application in firefighter incident command: a dashboard that monitors, from radio transcripts, the state of predefined tasks derived from standard operating procedures (SOPs) and related procedures. Building such a dashboard raises a practical design question: how much transcript structure is actually needed for LLM-based task-state monitoring? More specifically, we examine whether additional transcript structure materially improves monitoring performance, even though it is difficult to obtain reliably from radio communication and increases complexity and latency. We evaluate this question on source-grounded synthetic firefighter scenarios under transcript conditions that vary speaker identity and utterance boundaries, with incremental inference as the deployment-facing condition and full-transcript inference as an offline reference. Across repeated runs, incremental monitoring remains strong across all transcript conditions. Differences between transcript structures are small, continuous transcripts remain competitive, and the main weaknesses are unit-related assignment timing and capturing the completion result, which remain broadly similar across conditions. These results suggest that for this bounded dashboard-support capability, neither speaker identities nor semantically precise utterance boundaries are a primary requirement in the controlled setting studied here.

1 Introduction

Large language models are now widely used in bilateral settings such as chat assistants and individual copilots (Maedche et al., 2019; Nah et al., 2023). Yet many real work settings require digital support for several humans who coordinate through ongoing communication and maintain a shared operational picture rather than seek isolated

answers. Recent work on human-AI collaboration therefore argues for moving beyond assistant-centric interaction toward AI systems that support team processes and shared coordination (Seeber et al., 2020; Anthony et al., 2023; Banks et al., 2024). In mission-critical domains, one concrete support need is to maintain an explicit view of what has been assigned, what remains pending, and what has been completed.

Firefighter incident command provides a concrete instance of this broader problem. Commanders coordinate multiple units through short, interleaved radio transmissions while tracking operational tasks. We study this setting through a command-support dashboard whose task list is already derived from standard operating procedures (SOPs), related procedures, and incident context. The model does not discover new tasks or make operational decisions. Instead, it supports blind-spot checking by proposing conservative state updates for predefined tasks under human oversight.

Once such a dashboard is the target system, a more specific technical question becomes central: how much transcript structure is actually needed for LLM-based task monitoring from multi-party radio traffic? Rich transcript structure may be costly or difficult to obtain robustly in this setting. Speaker identity may require diarization or prior speaker models, both of which remain difficult under noisy multi-speaker conditions and heterogeneous channels (Mehri et al., 2023). Utterance boundaries may be supplied by voice activity detection, but pause-based segmentation can split incomplete thoughts, while more refined speaker-change or semantic segmentation adds models, engineering effort, and delay (Mehri et al., 2023). For a command-support system, these are not cosmetic transcript properties but potential deployment bottlenecks because real-time speech systems must trade off model complexity against latency and computational load (Michelsanti et al., 2021).

This motivates the research question of the paper: *how much transcript structure is actually needed to track predefined task state reliably from ongoing multi-party communication?* Specifically, we ask whether task-state monitoring depends on speaker identities and explicit utterance boundaries, or whether a simpler transcript representation suffices for this monitoring task.

We address this question through a controlled evaluation of closed-world task-state monitoring across transcript conditions that vary speaker identity and utterance boundaries while keeping the underlying scenario content fixed. The paper contributes in three ways. First, we formulate a bounded coordination-support problem centered on monitoring the state of predefined SOP- and procedure-related tasks in firefighter incident command. Second, we introduce a source-grounded synthetic dataset construction and validation pipeline that enables controlled comparison of transcript-structure effects while holding operational content constant. Third, we report results for incremental monitoring and a secondary offline full-transcript reference in a reproducible evaluation pipeline with auditable artifacts and local tests for scenario handling, prefix-gold derivation, parsing, and metric computation¹.

The remainder of the paper is structured as follows. Section 2 situates the work in the literature on multi-human AI support and facilitation. Sects. 3–5 then cover the application framing, dataset, and experimental design, before Sect. 6 reports the results and Sect. 7 concludes.

2 Background

This section situates the paper at the intersection of multi-human AI support and team information-processing support. It motivates why a dashboard-centered command-support system can be understood as one instance of a broader class of digital support systems for shared coordination, and why task-state monitoring is a meaningful bounded capability within that class.

A large share of current generative AI applications remains focused on the individual user: question answering, drafting, summarisation, or recommendation for a single human operator (Maedche et al., 2019; Nah et al., 2023). By contrast, many work settings require models to interpret contributions distributed across several people, maintain

state over time, and support shared rather than individual understanding. Reviews of human-AI teaming show that much of the literature still focuses on bilateral assistants, dyads, or otherwise narrowly scoped team configurations (Lyons et al., 2021; O’Neill et al., 2022; Anthony et al., 2023; Bankins et al., 2024). This leaves open how AI systems should support shared state and coordination across ongoing multi-party interaction through a shared artefact such as a dashboard.

Research on Group Support Systems has long shown that collaboration quality depends not only on who contributes, but also on how information is processed and coordinated within the group (Nunamaker et al., 1991; Dennis and Valacich, 1993; Dennis et al., 2001; Briggs et al., 2003). Facilitation has been central in this tradition because groups systematically suffer from process losses such as production blocking, dominance effects, incomplete information pooling, and premature convergence (Stasser and Titus, 1985; Dennis and Valacich, 1993; Briggs et al., 2003). Recent work on AI in teams suggests that digital artefacts may take over selected support capabilities, especially those involving consistency, timing, monitoring, or large-scale information processing, while humans retain judgement, legitimacy, and contextual authority (Dellermann et al., 2019; Seeber et al., 2020; Dennis et al., 2023).

Our paper focuses on one such capability: maintaining an explicit representation of task state from ongoing multi-party communication. In facilitation terms, this is primarily an information-processing support function. The command-support system does not replace operational decision making. Rather, it externalises on the dashboard what the team has already established: which tasks have been handed out, which unit appears responsible, which tasks have been completed, and what completion result has been reported.

Technically, this formulation is related to dialogue state tracking and information extraction. Dialogue state tracking maintains structured state representations from dialogue history in task-oriented systems (Balaraman et al., 2021). Information and event extraction similarly recover structured records, events, or arguments from text, often against predefined schemas (Chambers and Jurafsky, 2011; Xiang and Wang, 2019). Our setting differs in that the state variables are predefined SOP- and procedure-related operational tasks, the evidence is distributed across multi-party radio com-

¹https://github.com/zhaw-iwi/swisstext26_pub

munication, and the output is used for incremental dashboard support rather than autonomous dialogue management or open-ended event discovery.

We study this capability in a deliberately bounded operational setting rather than in the most open-ended form of multi-human coordination. If a digital support system cannot reliably maintain assignment and completion state for predefined tasks, then more advanced functions such as summarisation, blind-spot detection, or process coaching are unlikely to be trustworthy. Conversely, if task-state monitoring proves robust under reduced transcript structure, this suggests that at least some multi-human information-processing functions may not require expensive preprocessing pipelines.

Taken together, these strands motivate AI support for shared coordination, but they do not yet answer the more specific deployment question studied in this paper: how much transcript structure is required for closed-world task-state monitoring from ongoing multi-party communication? Firefighter incident command is a useful case because the dashboard-centered support function is operationally meaningful, the communication is genuinely multi-party, and transcript enrichment can be costly. For a bounded function such as digital task-state tracking, richer transcript preprocessing may help, but it is not obvious that speaker labels and clean utterance boundaries are necessary.

3 Application Context and Requirements

Firefighter incident command provides a concrete example of the broader class of multi-human digital support settings outlined in the previous section. In this setting, the command-support system is not meant to converse with a single user in isolation. Instead, it supports shared coordination by helping incident command maintain an explicit, inspectable view of operational task state as evidence arrives through multi-party radio traffic. We use this case not as the whole claim of the paper, but as a demanding setting from which to derive requirements for one bounded coordination-support capability.

3.1 Application Setting

The user-facing surface is a dashboard that presents the task list together with current task status and recent communication context, as shown in Figure 1. Its role is to reduce memory burden and preserve an explicit overview of open and completed tasks. The dashboard thus serves as the interface of a

bounded information-processing support system for the command team. It helps keep tasks from getting lost in noisy, interleaved, time-critical radio communication.

Figure 2 situates this dashboard within our implemented command-support system. Predefined tasks originate from SOPs, related procedures, and incident context, for example from baseline checklists associated with the incident type, from object-specific information, and from a building register. This information is available and selected at dispatch time. Radio communication then provides the dynamic evidence for whether those tasks have been assigned to firefighter units, acknowledged, and completed. This paper therefore treats task-state monitoring as a closed-world problem: the task set is supplied in advance, and transcript evidence is used to update task state.

This application framing is also consistent with preliminary feedback from a prototype demonstration of our command-support system with active firefighters. Beyond transcript capture itself, they emphasized the value of temporal tracking features, such as seeing how long a task has remained open or how long a crew has been silent. We treat this feedback as informal practitioner grounding rather than as a formal user-study result.

3.2 Monitoring Problem

Emergency radio communication differs substantially from the dialogue structure often assumed in conversational NLP. Messages are short, domain-specific, and often elliptical. Operational meaning is distributed across acknowledgements, updates, and follow-up reports rather than contained in a single self-sufficient utterance. For the command-support system, useful support therefore depends on reconstructing task state from distributed conversational evidence rather than isolated turns.

This creates the concrete design pressure behind our research question. If the dashboard requires speaker identities, the implementation may depend on diarization or prior speaker models despite radio-channel noise and changing crews (Mehrish et al., 2023). If it requires semantically well-formed utterance boundaries, pause-based segmentation may be insufficient and additional speaker-change or semantic segmentation steps may be needed before task monitoring can run (Mehrish et al., 2023). These components increase engineering effort and add latency, which is a relevant concern in real-time speech systems that must trade off model complex-

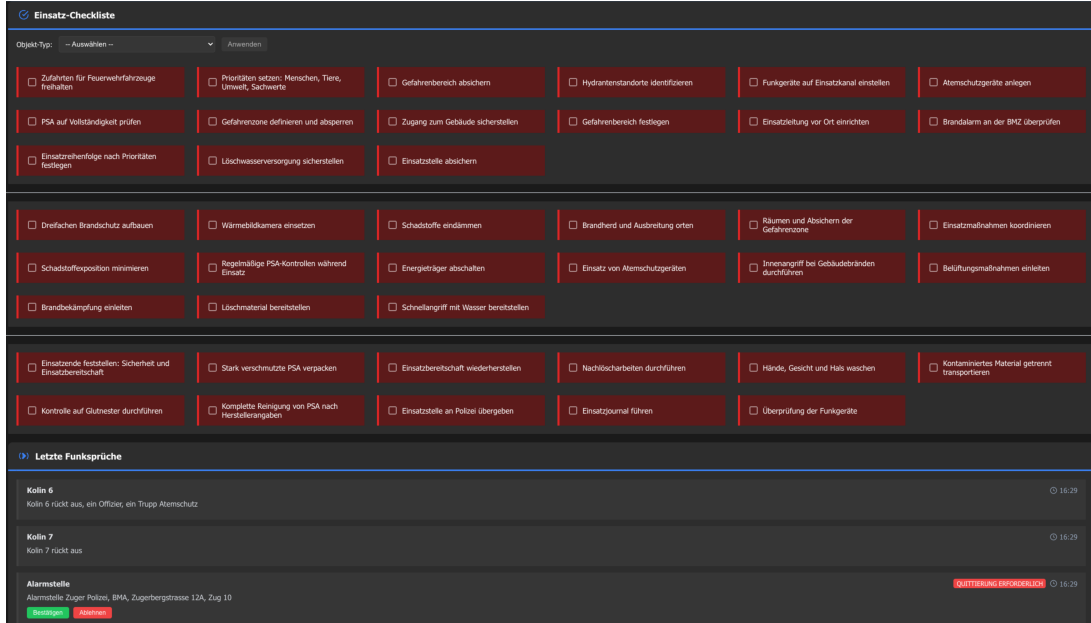


Figure 1: Dashboard interface of our command-support system.

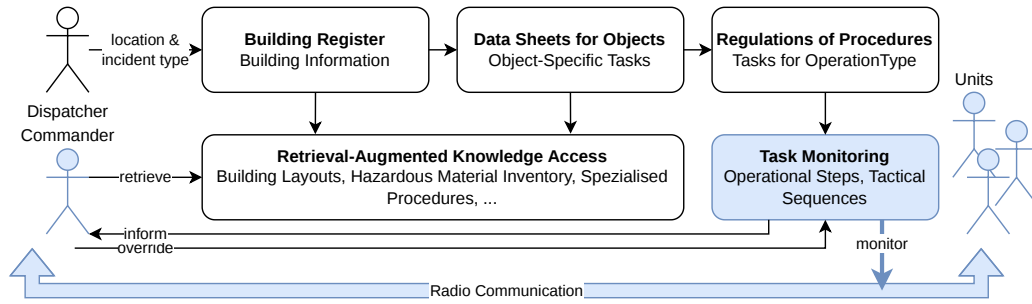


Figure 2: Operational flow of our command-support system, with dashboard components shown in blue.

ity against computational load (Michelsanti et al., 2021). In our setting, such components may fail precisely in the noisy conditions where operational support is most valuable. We therefore investigate how much structure the transcript representation actually needs for this monitoring task. More broadly, the firefighter case can thus be understood as one example of a digital support system that performs a bounded, process-relevant information-processing function inside a human team.

3.3 System Requirements

The command-support system must support commanders in checking that SOP-relevant tasks have been addressed. In our setting, these predefined SOP- and procedure-related tasks are already available from SOPs, related procedures, and scenario context before transcript analysis. The scope is therefore closed-world monitoring of predefined tasks rather than open-ended task discovery.

The command-support system must operate under human oversight. Proposed task-state updates must remain transparent to the command team and be subject to human confirmation or override.

The command-support system must support near-real-time dashboard updates during ongoing incidents. Task-state information must be revised as new communication evidence becomes available so that the dashboard remains a current overview.

3.4 Task Formulation

These application requirements lead to a closed-world task formulation. For each predefined task τ and each discrete prefix index k , the model outputs a structured task-state prediction

$$y(\tau, k) = (a, u, c, o).$$

Here, k denotes the current transcript prefix, that is, the communication evidence available up to that point. The tuple components correspond

to assigned, assigned_unit, completed, and completion_outcome, respectively. assigned (boolean) indicates whether the transcript provides evidence that responsibility for the task has been assigned to a unit and acknowledged. assigned_unit (nullable string) indicates which unit is identified as responsible for the task. completed (boolean) indicates whether the transcript provides evidence that the task has been completed. completion_outcome (nullable string) captures the reported completion result or completion evidence in short textual form.

4 Dataset

We evaluate the monitoring formulation on a controlled synthetic dataset. The synthetic design is deliberate: the research question requires the same underlying operational scenario to be rendered under different transcript-structure conditions while keeping task content fixed. A source-grounded synthetic dataset makes this possible, allowing differences between transcript conditions to be attributed to representation rather than scenario content.

4.1 Source-Grounded Construction

The dataset was constructed with a prompt-based generation and validation pipeline grounded in three source types: Swiss radio-procedure material², a firefighter communication transcript used for surface realism, and canonical procedural regulations³. These materials were distilled into source notes used for scenario generation and validation.

Each scenario was generated as a fixed-schema JSON object containing an ordered message sequence, a closed-world list of predefined tasks, and gold task states. In addition to the final task state, each gold entry stores the message id at which assignment first becomes valid and, when applicable, the message id at which completion first becomes valid. Prefix-level gold states are then derived deterministically from these transition annotations. Cases in which transition timing would remain ambiguous are resolved through explicit metadata rather than heuristic inference.

Validation was performed in two independent rubric-guided passes: a structural pass for radio style and protocol coherence, and a content pass for operational plausibility, role-task alignment, task

sequencing, and traceability of gold states to explicit message evidence. We implemented both dataset generation and validation with gpt-5.4 as a bounded LLM judgment process under source-grounded constraints (Zheng et al., 2023; Röttger et al., 2024). This source-grounded generation-and-judge setup was used to ground the synthetic data in the intended application setting. The generation-validation-revision loop ran for 12 rounds until the dataset passed both validation streams.

As an additional validation layer before evaluation, all scenario files pass deterministic local checks for schema conformance, sequential message ids, exact alignment between predefined tasks and gold task states, and consistency constraints on assignment/completion labels.

As an additional plausibility check, the extracted structure and resulting scenarios were reviewed within the team. They were judged to align with observed communication structures and contents from the intended application setting. While not a formal annotation study, this review provides additional support for the dataset’s plausibility as a set of realistically structured synthetic scenarios.

4.2 Dataset Contents

The final dataset contains five German-language firefighter scenarios with 102 ordered radio messages and 15 predefined tasks in total. Every scenario contains exactly three monitored tasks, yielding 15 task-state traces overall. All 15 tasks are explicitly assigned in the transcript evidence, 12 are completed within the scenario, and 3 remain assigned but incomplete at scenario end.

The five scenarios cover a kitchen fire in an apartment building, an underground-garage fire, a school basement fire, a workshop fire involving gas cylinders, and an attic fire in a row house. Across these scenarios, the monitored task families include water supply, search, ventilation, fire suppression, access control, gas-cylinder cooling, CO measurement, and final control. The assigned units include command-relevant firefighter roles such as Angriffstrupp, Wassertrupp, Messgruppe, Kontrolltrupp, and Verkehrstrupp.

4.3 Limitations

The limitations of the study follow directly from this dataset design and should be read upfront. The dataset is synthetic, small, and intentionally controlled. The scenarios are cleaner and more linear than real firefighter radio traffic, with limited pro-

²Bundesamt für Bevölkerungsschutz, Funkmaterial Sprechregeln, February 2004

³Feuerwehr Koordination Schweiz FKS, Reglement Einsatzführung, November 2022

to noise, dispatch and handover tails, and some bundled tasks that compress multi-step work into a single monitored item. The evaluation also begins from text transcripts rather than raw audio, so it does not include upstream speech-recognition or radio-channel degradation effects. Finally, the task formulation is deliberately narrow: it tests closed-world monitoring of predefined tasks, not open-ended task discovery or broad operational performance. We therefore use the dataset as controlled evidence for transcript-structure sensitivity in dashboard-oriented task monitoring, not as a realism corpus or a direct deployment claim.

5 Experimental Setup

We evaluate the dataset under a 3×2 design that varies transcript structure and processing mode while keeping the underlying scenario content fixed.

5.1 Conditions

We compare three transcript structures: **structured_dialogue**, which includes speaker identities and utterance boundaries, **no_speaker**, which retains utterance boundaries but omits speakers, and **continuous_transcript**, which contains neither speakers nor boundaries. This last condition approximates the minimum structure available when speakers cannot be identified and reliable utterance segmentation is not available.

We evaluate two processing modes: **incremental**, which predicts on each growing message prefix from the first message to the full transcript, and **full_transcript**, which predicts once on the complete transcript.

Incremental processing is the primary application condition. The `full_transcript` condition serves as a secondary offline reference once all evidence is available. For incremental `continuous_transcript` mode, each prefix is rendered as one continuous text string without speaker labels or message boundaries.

5.2 Execution and Reproducibility

All reported values are batch-level means across 12 repeated runs. The notebook runs the evaluation in live mode with `gpt-5.2` at temperature 0.0. Repeated runs are reported because live API inference can still show small residual variation despite deterministic settings. The evaluation prompt instructed the model to behave conservatively: when

the transcript did not provide sufficient evidence for assignment or completion, the corresponding fields were to remain false/null. This reflects the operational preference to avoid overstating task progress under incomplete evidence. The evaluation pipeline persists each request and response artifact using deterministic identifiers over run, scenario, structure condition, processing mode, and prefix index, supporting resume-safe execution and auditability. The repository contains the dataset-generation prompts, validation prompts, source notes, evaluation prompt payloads, and persisted request/response artifacts. The tables report 95% confidence intervals from the batch summaries exported by the notebook and evaluation code.

5.3 Metrics

We report four state-monitoring metrics in both processing modes. In the incremental setting, each metric is computed at every evaluated prefix and then averaged over prefixes. In the `full_transcript` setting, the same metric is computed once on the final scenario transcript.

Assignment accuracy. This metric measures, for each predefined task, whether the predicted assigned value matches gold.

Unit assignment accuracy. This metric measures whether the predicted `assigned_unit` matches gold. It is evaluated on tasks for which either gold or prediction marks the task as assigned, so that unit assignment is scored only where a responsible unit is relevant. In the incremental setting, scores are first computed per prefix and then averaged. Prefixes without any unit-assignment support contribute 0.0 by construction.

Completion accuracy. This metric measures, for each predefined task, whether the predicted completed value matches gold.

State accuracy. This metric collapses each task state to one of `NOT_ASSIGNED`, `ASSIGNED`, or `COMPLETED`, and then measures whether the predicted three-way state matches gold.

For incremental evaluation, we additionally report assignment and completion detection latency together with assignment and completion miss rate. The gold transition points come from explicit scenario annotations for first assignment and, when applicable, first completion evidence. Assignment is anchored to the first operative command that hands responsibility for the predefined task to the assigned unit. Later refinements to the same unit do not reset the assignment point.

Assignment detection latency. This metric measures the number of prefix steps between the gold assignment point and the first prefix at which the model predicts the task as assigned with the correct unit. It is reported only for successfully detected assignment events.

Completion detection latency. This metric measures the number of prefix steps between the gold completion point and the first prefix at which the model predicts the task as completed while preserving assigned state and the correct unit. It is reported only for successfully detected completion events.

Assignment detection miss rate. This metric is the proportion of gold assignment events that are never correctly detected before the scenario ends.

Completion detection miss rate. This metric is the proportion of gold completion events that are never correctly detected before the scenario ends.

Terminal gaps. As a reference comparison, we report terminal assignment, unit-assignment, completion, and state gaps. These are the absolute differences between the last incremental-prefix metrics and the corresponding `full_transcript` metrics for the same run, scenario, and transcript structure.

Completion outcome. For `completion_outcome`, exact match proved too strict because the model often paraphrases the completion evidence rather than reproducing the gold message string verbatim. We therefore report a retrospective similarity-based reanalysis on gold-completed tasks using ROUGE-L F1 as a secondary reference metric.

6 Results

The main empirical result is straightforward: in this controlled setup, simplifying transcript structure has little effect on aggregate task-state monitoring performance. Across all conditions, assignment, completion, and current-state accuracy remain high. The lowest aggregate metric is unit assignment, but this should be interpreted as a stricter prefix-level score rather than as frequent confusion between responsible units.

6.1 Incremental Accuracy

Incremental inference is the more deployment-relevant setting because the command-support system must update task state while communication unfolds. Table 1 shows that performance remains strong and tightly clustered across transcript struc-

Structure	Assign.	Unit	Complete	Current
Structured dialogue	0.981 ± 0.006	0.864 ± 0.016	0.977 ± 0.008	0.958 ± 0.011
No speaker	0.980 ± 0.007	0.865 ± 0.016	0.978 ± 0.007	0.958 ± 0.009
Continuous transcript	0.981 ± 0.007	0.868 ± 0.014	0.978 ± 0.007	0.959 ± 0.010

Table 1: Incremental monitoring accuracy by transcript structure.

tures. The unit-assignment column is consistently lower than the other state-tracking metrics, but inspection of the persisted artifacts suggests that this does not primarily reflect wrong responsible-unit names. Most unit mismatches arise when an incremental prediction has not yet marked a gold-assigned task as assigned, especially for conditional or preparatory tasking, with a smaller number of premature assignments. Thus, the harder part is aligning assignment timing and unit grounding as evidence unfolds, rather than choosing among units once a task is accepted as assigned.

In the offline `full_transcript` reference, assignment and unit-assignment accuracy are 1.000 in all three transcript conditions. Completion and final-state accuracy are 0.933 in all conditions.

6.2 Incremental Latency and Misses

Latency is the second key deployment-oriented result because the dashboard benefits not only from eventually correct updates but from updates that arrive close to the true transition point. We therefore report two complementary quantities in Table 2: conditional latency in prefix steps for events that were correctly detected, and miss rate for gold transitions never correctly detected before the scenario ended. Assignment latency therefore requires the model to mark a task as assigned with the correct unit, while completion latency requires the model to mark the task as completed while preserving coherent assigned state and the correct unit.

The latency results are straightforward. Across all three transcript conditions, correct assignment updates remain well below one prefix step on average and no assignment misses occur. Completion detection is even stronger, with effectively immediate updates and zero completion misses throughout. In practical terms, once the transcript supports a state change, the model usually updates at once or with only a very small delay.

6.3 Terminal Convergence

Table 3 reports absolute terminal gaps between the last incremental prefix metrics and the corresponding full-transcript reference for the same run,

Structure	Assign. lat.	Assign. miss	Comp. lat.	Comp. miss
Structured dialogue	0.256 ± 0.127	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
No speaker	0.256 ± 0.126	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
Continuous transcript	0.278 ± 0.133	0.000 ± 0.000	0.006 ± 0.011	0.000 ± 0.000

Table 2: Incremental correct-detection latency and miss rate by transcript structure.

Structure	Assign. gap	Unit gap	Comp. gap	State gap
Structured dialogue	0.000 ± 0.000	0.000 ± 0.000	0.006 ± 0.011	0.006 ± 0.011
No speaker	0.000 ± 0.000	0.000 ± 0.000	0.006 ± 0.011	0.006 ± 0.011
Continuous transcript	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

Table 3: Terminal convergence of the last incremental prefix to the offline full-transcript reference.

scenario, and transcript structure.

The convergence pattern is very tight. By the end of the scenario, the incremental setup almost exactly recovers the performance profile of the offline `full_transcript` reference. This makes the reference useful as a sanity check for interpreting the incremental results.

6.4 Completion-Outcome Reanalysis

The original exact-match `completion_outcome` metric remains 0.000 in all three transcript conditions and is therefore not useful on its own. Inspection of the outputs, however, suggests that this mostly reflects paraphrastic variation: the model often paraphrases the completion evidence instead of reproducing the gold completion string verbatim. We therefore ran a retrospective similarity-based comparison on the already collected outputs for gold-completed tasks only.

Table 4 reports ROUGE-L F1 for this reanalysis. The resulting scores are clearly above zero but remain broadly similar across transcript conditions. This makes the `completion_outcome` field somewhat more informative than exact match suggested, while leaving the main conclusion unchanged: transcript structure still has only small effects in this bounded monitoring setup. At the same time, the scores remain well below the stronger assignment and completion-state results, so we interpret them conservatively as evidence of partial string-level overlap rather than robust semantic understanding.

7 Conclusion

We studied task-state monitoring in firefighter incident command as a bounded coordination-support capability for an incident-command dashboard.

The central system-design question was whether this capability depends on speaker identities and explicit utterance boundaries, both of which can be

Structure	ROUGE-L F1
Structured dialogue	0.625 ± 0.007
No speaker	0.657 ± 0.015
Continuous transcript	0.638 ± 0.017

Table 4: Retrospective `completion_outcome` reanalysis on gold-completed tasks only.

expensive or difficult to obtain robustly from radio communication. For the controlled setting studied here, the answer is largely no: reduced transcript structure does not materially change the overall task-monitoring performance profile. The main remaining weaknesses are unit-related assignment timing in incremental prefixes and capturing the reported completion result rather than detecting assignment or completion itself, and both remain broadly similar across transcript conditions.

More broadly, this suggests that some coordination-support functions may be feasible without heavy transcript preprocessing. Methodologically, the paper also shows how a source-grounded synthetic dataset construction and validation pipeline can support controlled comparison of transcript-structure conditions while holding operational content fixed. At the same time, the present evidence remains deliberately narrow: it comes from a controlled synthetic dataset, from text rather than raw audio, and from a closed-world monitoring task. Future work should therefore test whether the same pattern holds under noisier audio pipelines, less linear communication, and stronger ambiguity, while improving unit-assignment grounding, evidence presentation, and human-in-the-loop integration into command workflows. A further direction is to move from closed-world monitoring toward mixed closed- and open-world coordination support, where systems track predefined tasks while also identifying additional operational tasks that emerge from the incident record. Another direction is to connect evolving coordination state to relevant supporting knowledge resources, so that a shared dashboard can surface applicable procedures, plans, or reference documents without turning the system into an autonomous decision maker.

References

- Chris Anthony, Beth A. Bechky, and Anne-Laure Fayard. 2023. “collaborating” with AI: Taking a system view to explore the future of work. *Organization Science*, 34(5):1672–1694.
- Vevake Balaraman, Seyedmostafa Sheikhalishahi, and Bernardo Magnini. 2021. Recent neural methods on dialogue state tracking for task-oriented dialogue systems: A survey. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 239–251.
- Sarah Bankins, Alannah C. Ocampo, Maurizio Marrone, Simon L. D. Restubog, and Sang Eun Woo. 2024. A multilevel review of artificial intelligence in organizations: Implications for organizational behavior research and practice. *Journal of Organizational Behavior*, 45(2):159–182.
- Robert O. Briggs, Gert-Jan de Vreede, and Jay F. Nunamaker. 2003. Collaboration engineering with thinklets to pursue sustained success with group support systems. *Journal of Management Information Systems*, 19(4):31–64.
- Nathanael Chambers and Dan Jurafsky. 2011. Template-based information extraction without the templates. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 976–986.
- Dominik Dellermann, Adrian Calma, Nicola Lipusch, Tim Weber, Sascha Weigel, and Patrick Ebel. 2019. The future of human-AI collaboration: A taxonomy of design knowledge for hybrid intelligence systems. In *Proceedings of the Hawaii International Conference on System Sciences (HICSS)*.
- Alan R. Dennis, Anjali Lakhiwal, and Akanksha Sachdeva. 2023. AI agents as team members: Effects on satisfaction, conflict, trustworthiness, and willingness to work with. *Journal of Management Information Systems*, 40(2):49–54.
- Alan R. Dennis and Joseph S. Valacich. 1993. Computer brainstorming: More heads are better than one. *Journal of Applied Psychology*, 78(4):531–537.
- Alan R. Dennis, Barbara H. Wixom, and Robert J. Vandenberg. 2001. Understanding fit and appropriation effects in group support systems via meta-analysis. *MIS Quarterly*, 25(2):167–197.
- Joseph B. Lyons, Katia Sycara, Michael Lewis, and Alexandra Capiola. 2021. Human-autonomy teaming: Definitions, debates, and directions. *Frontiers in Psychology*, 12:589585.
- Alexander Maedche, Christine Legner, Alexander Benlian, Benedikt Berger, Henner Gimpel, Thomas Hess, Oliver Hinz, Stefan Morana, and Matthias Söllner. 2019. Ai-based digital assistants. *Business & Information Systems Engineering*, 61(4):535–544.
- Ambuj Mehrish, Navonil Majumder, Rishabh Bhardwaj, and Soujanya Poria. 2023. A review of deep learning techniques for speech processing. *Information Fusion*, 99:101869.
- Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. 2021. An overview of deep-learning-based audio-visual speech enhancement and separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1368–1396.
- Fiona F.-H. Nah, Ruilin Zheng, Jian Cai, Keng Siau, and Langtao Chen. 2023. Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3):277–304.
- Jay F. Nunamaker, Alan R. Dennis, Joseph S. Valacich, Douglas R. Vogel, and Joey F. George. 1991. Electronic meeting systems to support group work: Theory and practice at arizona. *Communications of the ACM*, 34(7):40–61.
- Thomas O’Neill, Nathan McNeese, Anthony Barron, and Brandon Schelble. 2022. Human-autonomy teaming: A review and analysis of the empirical literature. *Human Factors*, 64(5):904–938.
- Paul Röttger, Fabio Pernisi, Bertie Vidgen, and Dirk Hovy. 2024. SafetyPrompts: a systematic review of open datasets for evaluating and improving large language model safety. *arXiv preprint arXiv:2404.05399*.
- Isabella Seeber, Eva Bittner, Robert O. Briggs, Triparna de Vreede, Gert-Jan de Vreede, Aaron Elkins, Ronald Maier, Alexander B. Merz, Sarah Oeste-Reiß, Niels Randrup, Gerhard Schwabe, and Matthias Söllner. 2020. Machines as teammates: A research agenda on ai in team collaboration. *Information & Management*, 57(2):103174.
- Garold Stasser and William Titus. 1985. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of Personality and Social Psychology*, 48(6):1467–1478.
- Wei Xiang and Bang Wang. 2019. A survey of event extraction from text. *IEEE Access*, 7:173111–173137.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Datasets and Benchmarks Track*.

Can Large Language Models Replace Statistical Software?

Prof. Dr. Yves Staudt

Institute of Data Analysis, Artificial Intelligence, Visualization and Simulation
University of Applied Sciences of the Grisons
yves.staudt@fhgr.ch

Abstract

Statistical hypothesis testing is a cornerstone of evidence-based medicine and clinical research. Despite its central importance, previous research has consistently shown substantial deficits in statistical literacy among healthcare professionals. At the same time, large language models (LLMs) have demonstrated remarkable capabilities in scientific reasoning and data analysis. This study examines whether LLMs can serve as viable substitutes for conventional statistical software in guiding users through the selection, execution, and interpretation of hypothesis tests. Using a standardized prompt based on real survey data on the association between kick-scooter riding and knee pain in children, we evaluated seven LLMs and compared their outputs with statistical software results. Our findings indicate that none of the evaluated models can currently be considered a viable substitute. Although all models selected the appropriate test, substantial variation was observed in the quality of their explanations and in test execution. Gemini 3.1 Pro Preview, Claude Opus 4.6, and ChatGPT 5.4 Thinking performed strongly in test selection and result interpretation, with Gemini producing the most structured responses. However, none matched statistical software's result in test execution.

1 Introduction

Statistical hypothesis testing is one of the most widely used methodological tools in empirical research. It addresses a fundamental question in scientific inquiry: whether an observed effect or association in a sample reflects a true phenomenon in the underlying population or can instead be attributed to chance (Turner et al., 2020; Zhang, 2025). By quantifying uncertainty and enabling structured decision-making, hypothesis testing has become indispensable to evidence-based medicine, underpinning trial design, therapeutic comparisons, and diagnostic evaluation (Turner et al., 2020; Zhang, 2025).

At the same time, several studies indicate that statistical knowledge among medical researchers and healthcare professionals is often limited (Jenny et al., 2018). Selecting suitable methods, analysing data correctly, and drawing valid conclusions all involve considerable potential for error (Lakhlifi et al., 2023). Insufficient statistical competence may therefore substantially compromise the scientific quality of medical research.

Recent advances in large language models (LLMs) have introduced a useful interface for statistical reasoning and data analysis (Angelis et al., 2023; Thirunavukarasu et al., 2023). Unlike traditional statistical software such as SPSS or R, LLMs can be queried in natural language, which may lower the barrier to entry for users with limited statistical training. This raises the question of whether LLMs can act as intelligent assistants in statistical hypothesis testing.

In this paper, we investigate whether contemporary LLMs-ChatGPT 5.4 Thinking (OpenAI, 2026), Gemini 3.1 Pro Preview (Google DeepMind, 2026), Claude Opus 4.6 (Anthropic, 2026), Llama 4 Scout (Meta AI, 2025), Apertus 8b (Swiss AI Initiative, 2025), Qwen 3.5 9b (Swiss AI Initiative, 2025), and Phi 4 (Abdin et al., 2024)-can serve as substitutes for traditional statistical software in hypothesis testing. We evaluate their ability to select, execute, and interpret hypothesis tests by comparing their outputs with results obtained from conventional statistical software. Our findings indicate that LLMs cannot currently replace standard statistical tools, although they may still serve as useful supportive systems. Among the evaluated models, Gemini, Claude Opus, and ChatGPT showed the strongest overall documentation and performance.

2 Literature Review

2.1 Hypothesis Testing and Statistical Literacy

Hypothesis testing remains a central component of statistical inference in empirical research (O’Dushlaine, 2019). At the same time, methodologists have warned against an overly narrow focus on statistical significance alone, arguing for complementary reporting of effect sizes and confidence intervals (Mark et al., 2016; Patel and Green, 2025). In healthcare, several studies show that professionals often lack the statistical literacy needed to interpret p -values, confidence intervals, and diagnostic statistics correctly (Jenny et al., 2018; Lakhlifi et al., 2023). This is particularly concerning because weaknesses in statistical reasoning may compromise the planning, analysis, and interpretation of scientific studies.

2.2 LLMs as Statistical Assistants

LLMs have emerged as potentially useful tools for statistical assistance because of their capabilities in instruction following, reasoning over textual input, and code generation. In principle, an LLM should be able to identify an appropriate statistical test based on data structure, variable types, and assumptions (Fay and Brittain, 2022; Nikolić and Popovic, 2024). However, whether these models can reliably perform statistical calculations and provide valid interpretations remains an open empirical question, particularly in medically relevant contexts (Liu, 2025). This question motivates the present study.

3 Methodology

The aim of this study is to evaluate whether LLMs can serve as viable substitutes for conventional statistical software in the context of hypothesis testing. To this end, we benchmarked the models’ outputs for test selection, execution, and interpretation against reference results obtained from statistical software. For this purpose, we defined a standardized prompting procedure, a fixed output structure, and an explicit evaluation framework. This design was intended to ensure comparability, transparency, and reproducibility across all tested models.

3.1 Prompt

All models were evaluated with the same prompt and input data, without model-specific adaptations,

to minimize variation unrelated to model capability. Responses were analysed according to the predefined evaluation framework. The benchmark prompt was designed using a chain-of-thought-inspired reasoning framework (Sahoo et al., 2025). In the first part, the research problem and relevant tabular input data were provided. To ensure reproducibility and reuse, the complete prompt was documented in Markdown, with the input data embedded in the same file.

In the second part, the LLM was instructed to analyse the data and perform the hypothesis test. The prompt specified the required output format, the expected data structure, and the handling of missing values. In addition, the models were instructed to solve the task using a zero-shot chain-of-thought strategy. This ensured that all models received the same information under identical conditions.

3.2 Output Structure

To compare model responses systematically, a fixed output structure was imposed on all generated answers. This structure was chosen to improve transparency, traceability, and comparability across models. It was inspired by the classical statistical procedure for hypothesis testing described by (Gonick and Smith, 2005), who conceptualize hypothesis testing as a four-step process: formulation of hypotheses, calculation of the test statistic, calculation of the p -value, and evaluation of the result against a predefined significance level (α).

To assess whether LLMs can perform this process in practice, we adapted the classical procedure into a task-oriented format suitable for prompting and software-based execution. Accordingly, the required output structure consisted of the following four steps:

1. Selection of the appropriate statistical test and formulation of the hypotheses
2. Execution of the statistical test
3. Summary of the statistical results
4. Interpretation of the results

3.3 Model Selection

Seven models were selected to represent different segments of the contemporary LLM landscape:

- **Proprietary high-end models:** ChatGPT 5.4 Thinking (OpenAI, 2026), Claude Opus 4.6

(Anthropic, 2026), and Gemini 3.1 Pro Preview (Google DeepMind, 2026)

- **Open-weight benchmark models:** Llama 4 Scout (Meta AI, 2025) and Qwen 3.5 9b (Qwen Team, 2026)
- **Compact models:** Phi 4 (Abdin et al., 2024) and Apertus 8b (Swiss AI Initiative, 2025)

This selection captures variation in model size, accessibility, and expected performance. By including proprietary, open-weight, and compact models, the benchmark enables a broader comparison of how closely current LLMs approximate conventional statistical software. To reflect realistic usage, all models were accessed through web interfaces. The open-weight and compact models were hosted locally using Ollama, whereas the proprietary models were accessed through the providers' web interfaces.

3.4 Output Evaluation

The benchmark task focused on a defined use case (see Section 4) involving the association between two binary variables. For this type of problem, the appropriate statistical procedure is the χ^2 -test of independence. The corresponding hypotheses are:

- Null hypothesis (H_0): There is no relationship between the two categorical variables.
- Alternative hypothesis (H_1): There is a statistically significant relationship between the two categorical variables.

To solve this task correctly, a model had to identify the appropriate test, determine the observed joint frequencies in a contingency table, derive the expected frequencies under the assumption of independence, calculate the test statistic and the corresponding p -value, and evaluate the result against the specified significance level. Because the calculation of frequencies depends on the sample size, correct determination of the sample size was included as an additional evaluation criterion. Finally, the model was expected to provide an interpretation of the result in the context of the use case.

Based on this setup, model outputs were evaluated according to the following criteria:

1. Selection of the statistical test
2. Formulation of the null and alternative hypotheses
3. Determination of the sample size

4. Presentation of the observed joint frequencies
5. Presentation of the expected joint frequencies
6. Provision of calculation details
7. Verification of test assumptions
8. Reporting of the p -value and other relevant statistical values
9. Identification or discussion of the significance level
10. Evaluation of the test result
11. Interpretation of the result

Each criterion was assessed for every evaluated LLM and later summarized in tabular form in the Results section (Section 5). All responses were additionally required to follow a structured Markdown format, reproduced in the Appendix, in order to support consistent documentation and systematic comparison.

To assess output quality across these criteria, we defined an ordinal grading scheme. This scheme captures graded differences in performance rather than relying on a binary correct/incorrect judgment. This was necessary because LLM responses frequently contained partial reasoning, conceptually correct intermediate steps, or errors that only became apparent during the application of the statistical procedure. The grading framework was therefore designed to distinguish between theoretical understanding, practical execution, and complete omission of a criterion.

Because the central research question of this study is whether LLMs can function as an alternative to statistical software, the grading scheme places particular emphasis on methodological correctness and successful application to the concrete use case. In this sense, the benchmark does not only assess whether a model mentions a relevant concept, but whether it applies the statistical procedure correctly and produces results consistent with the software-based reference solution.

All outputs were evaluated manually by the author with formal training in statistics, using the predefined criteria and grading rubric. To strengthen the consistency of the assessment, each response was systematically compared with the corresponding reference results produced by conventional statistical software.

The following grades were used:

- **A:** Criterion addressed correctly and applied correctly
- **B:** Criterion addressed correctly with minor application errors

- **C:** Criterion addressed correctly in principle, but affected by follow-up errors in application
- **D:** Criterion addressed correctly in principle, but with substantial application errors
- **E:** Theoretical understanding present, but no meaningful application to the use case
- **F:** Criterion treated incorrectly at the theoretical level
- **G:** Criterion not addressed

Grade A represents the highest level of performance, whereas Grade G represents the lowest. Grade G indicates that a criterion was omitted entirely. Grade F indicates that the model failed at the conceptual or methodological level. Grade E indicates that the model identified the relevant theoretical concept but did not successfully apply it to the given task. Grades A to D reflect increasing levels of correctness in execution. The distinction between Grades C and D is that Grade C was assigned when an error primarily resulted from an earlier mistake, whereas Grade D was assigned when the criterion itself was applied incorrectly.

4 Use Case

This study uses data from (Crottogini et al., 2026), who surveyed primary school students in grades 3 to 6 (aged 8 to 12 years) in Chur, Switzerland, about their kick-scooter use and knee health. The dataset comprises 134 complete responses, with a gender distribution of 51.5% female and 48.5% male. Most participants were 11 years old. Overall, 85% of respondents reported using a kick scooter either regularly or occasionally, and 41% reported regular use.

For the present benchmark, the analysis was restricted to the two variables *knee pain* and *kick-scooter riding*. Both variables were encoded as binary variables (yes/no). The observed joint frequencies are shown in Table 1.

Table 1: Observed contingency table for knee pain and kick-scooter use

		Kick-scooter riding		
		No	Yes	Total
Knee pain	No	14	55	69
	Yes	6	59	65
Total		20	114	134

This use case was selected because it represents a simple but realistic hypothesis-testing scenario in which correct test selection, computation, and inter-

pretation can be assessed against a clear software-based reference. This use case assesses how well LLMs can identify and analyse the relationship between these two variables by means of hypothesis testing. As a reference standard, the corresponding χ^2 -test of independence was performed in both Python and R. The tested hypotheses were:

- H_0 : There is no association between knee pain and kick-scooter riding.
- H_a : There is an association between knee pain and kick-scooter riding.

As described in Section 3.2, a key step is the correct derivation of the expected joint frequencies and test statistics. The expected frequencies for the contingency table are reported in Table 2, and the resulting statistical values are summarized in Table 3. Both Python and R applied Yates' continuity correction, commonly used for χ^2 -tests with one degree of freedom.

Table 2: Expected contingency table for knee pain and kick-scooter use under the null hypothesis of independence

		Kick-scooter riding	
		No	Yes
Knee pain	No	10.3	58.7
	Yes	9.7	55.3

Table 3: Summary of the statistical values of the χ^2 -test with Yates' continuity correction

Statistic	Value
χ^2 value	2.41
p -value	0.12
Degrees of freedom	1

Using a significance level of $\alpha = 0.05$, the null hypothesis cannot be rejected. Accordingly, the reference analysis does not provide sufficient evidence for an association between knee pain and kick-scooter riding in this sample.

For comparison, the χ^2 -test can also be calculated without Yates' continuity correction. In that case, the test statistic increases to $\chi^2 = 3.22$, and the p -value decreases to 0.07 (see Table 4). Although the null hypothesis still cannot be rejected at the 5% significance level, the result lies closer to the significance threshold. This illustrates the impact of Yates' correction in small 2×2 contingency

tables and highlights why correct handling of this issue is relevant when evaluating LLM-generated analyses.

Table 4: Summary of the statistical values of the χ^2 -test without Yates' continuity correction

Statistic	Value
χ^2 value	3.22
p -value	0.07
Degrees of freedom	1

5 Results

We assessed the seven models' structured outputs according to the criteria defined in Section 3.2. Table 5 summarizes the resulting grades. Across criteria, Gemini, Claude Opus, and ChatGPT documented and performed best, whereas Llama 4 and Qwen 3.5 performed worst.

Overall, all models were able to identify the task as a hypothesis-testing problem involving two binary variables and selected the χ^2 -test as the appropriate procedure. All models also formulated the null and alternative hypotheses correctly and recognized the significance level of $\alpha = 5\%$. Substantial differences emerged, however, in the execution of the analysis, especially in determining the correct population size, constructing the observed and expected frequency tables, checking assumptions, computing the relevant statistical values, and drawing the conclusion.

Only one model showed limitations already at the stage of data recognition (Figure 1). Llama 4 identified the variables as binary, but its description of the data structure was less complete than that of the other systems.

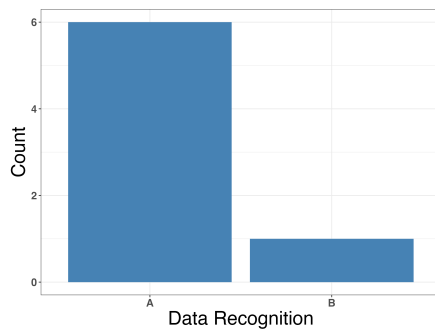


Figure 1: Recognition of the data format by the evaluated LLMs.

A clearer separation between stronger and

weaker models emerged in the determination of the population size. As shown in Figure 2, only three models-Gemini, ChatGPT, and Claude Opus-correctly identified the sample size. The remaining four models produced incorrect values, indicating hallucinated or inconsistent reconstructions of the input data.

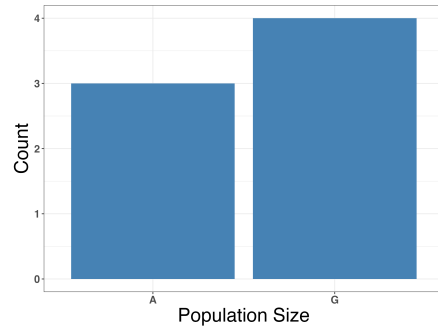


Figure 2: Performance of the LLMs in determining the population size.

These errors in population size were reflected directly in the observed contingency tables (Figure 3). All models that failed to determine the correct sample size also failed to reconstruct the observed joint frequencies correctly. In contrast, the observed frequency tables generated by ChatGPT, Gemini, and Claude Opus matched the reference results shown in Table 1.

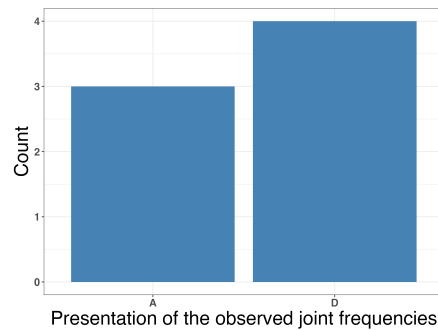


Figure 3: Performance of the LLMs in determining the observed joint frequencies.

Performance deteriorated further when the models were required to derive the expected joint frequencies (Figure 4). Llama 4 and Apertus did not provide expected frequencies at all. Qwen 3.5 produced expected frequencies that were not grounded in its own preceding contingency table, indicating a substantial inconsistency in the analytical chain. Phi 4 derived expected frequencies from an already incorrect observed table and therefore produced a follow-up error. Only ChatGPT, Gemini, and

Table 5: Summary of the evaluation of LLM performance across the predefined criteria.

Model	ChatGPT 5.4 Thinking	Gemini 3.1 Pro-Preview	Claude Opus 4.6	Llama 4 Scout 17b	Apertus 8b	Phi 4	Qwen 3.5 9b
Data Recognition	A	A	A	B	A	A	A
Selection of statistical test	A	A	A	A	A	A	A
Formulation of the null and alternative hypotheses	A	A	A	A	A	A	A
Determination of the sample size	A	A	A	G	G	G	G
Presentation of the observed joint frequencies	A	A	A	D	D	D	D
Presentation of the expected frequencies	A	A	A	G	G	C	E
Provision of calculation details	G	A	A	F	G	C	E
Verification of test assumption	A	A	A	G	G	G	G
Report of the p -value and other relevant statistical values	B	B	B	G	D	C	E
Identification or discussion of the significance level	A	A	A	A	A	A	A
Evaluation of the results	A	A	A	F	C	A	C
Interpretation of the results	B	A	A	G	A	A	B

Claude Opus generated expected frequencies consistent with the software-based reference reported in Table 2.

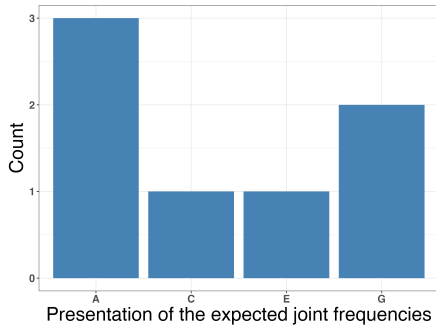


Figure 4: Performance of the LLMs in determining the expected joint frequencies.

The provision of calculation details also varied markedly across models (Figure 5). This criterion is particularly important because it allows expert users to verify whether the reported statistical values follow coherently from the data and formulas. Gemini and Claude Opus provided transparent and traceable intermediate steps. ChatGPT reported the key results but did not disclose intermediate calculations, which reduced transparency but did not prevent interpretation. Llama 4 failed to connect

the underlying statistical procedure to the concrete use case, while Apertus provided insufficient detail to reconstruct the analysis. Qwen 3.5 presented correct formulas in principle, but the numerical values did not consistently follow from the earlier steps, which limited the trustworthiness of the output.

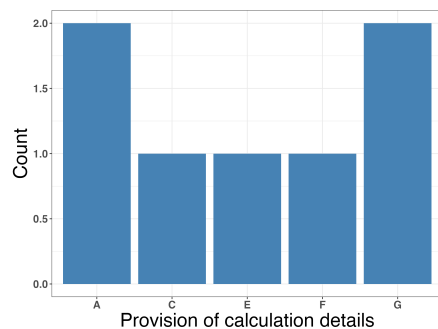


Figure 5: Performance of the LLMs in providing calculation details for the selected test.

Before interpreting the test outcome, it is necessary to verify whether the assumptions of the χ^2 -test are satisfied. As shown in Figure 6, only Gemini, ChatGPT, and Claude Opus explicitly checked the minimum assumptions of the test. The remaining models did not assess whether the minimum expected cell frequency requirement was met.

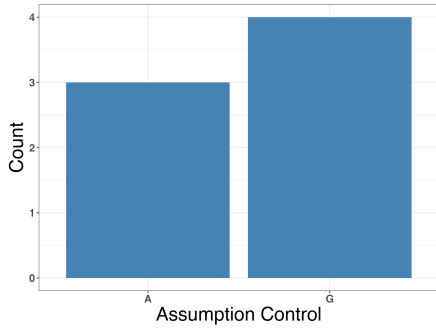


Figure 6: Performance of the LLMs in checking the assumptions of the χ^2 -test.

The most critical criterion concerned the reporting of the final statistical values, namely the χ^2 statistic, the p -value, and the degrees of freedom (Table 3). These values are essential because they allow an independent reader to verify the correctness of the analysis. Among all evaluated models, Gemini came closest to reproducing the reference calculations, as it was the only model to explicitly mention Yates’ correction. However, neither Gemini nor any other model reproduced the software output exactly. Instead, Gemini, Claude Opus, and ChatGPT reported values corresponding to the χ^2 -test without Yates’ continuity correction, that is, the values shown in Table 4 rather than those from Table 3.

This distinction is important because the omission of Yates’ correction shifted the result closer to the conventional significance threshold. In the present use case, the corrected analysis yields $\chi^2 = 2.41$ and $p = 0.12$, whereas the uncorrected analysis yields $\chi^2 = 3.22$ and $p = 0.07$. Thus, the stronger models were able to reconstruct an analytically plausible result, but not the exact reference result produced by conventional software. Phi 4 correctly propagated its own internal calculations, but these were based on an incorrect population size and therefore did not correspond to the actual dataset. Llama 4 did not provide usable final statistics, while the outputs of Apertus and Qwen 3.5 lacked sufficient consistency for reliable verification.

These differences also affected the evaluation and interpretation of the test results (Figures 8 and 9). Gemini and Claude Opus provided the strongest overall interpretations, including discussion of the small number of observations in the “no kick-scooter riding” category. ChatGPT also produced a largely coherent evaluation, although with

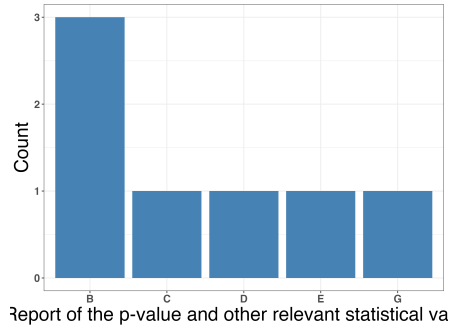


Figure 7: Performance of the LLMs in reporting the χ^2 statistic, p -value, and degrees of freedom.

less detail. In contrast, weaker models were unable to derive a reliable conclusion because earlier errors in data reconstruction or statistical execution propagated into the final interpretation.

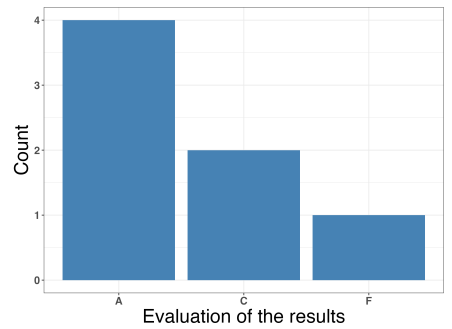


Figure 8: Performance of the LLMs in evaluating the result of the χ^2 -test.

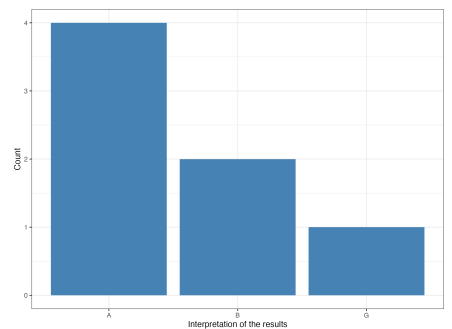


Figure 9: Performance of the LLMs in interpreting the χ^2 -test in relation to the use case.

6 Conclusion, Limitations, and Outlook

Statistical hypothesis testing remains one of the central methodological tools in medical research and evidence-based medicine (Turner et al., 2020). At the same time, previous work has shown that statistical literacy among medical researchers and healthcare professionals is often limited (Jenny

et al., 2018), which may negatively affect the selection of appropriate methods, the correct analysis of data, and the derivation of valid conclusions (Lakhlifi et al., 2023). In this context, LLMs have emerged as a potentially useful new interface for statistical reasoning and hypothesis testing (Fay and Brittain, 2022; Nikolić and Popovic, 2024). If such models were able to serve as reliable substitutes for conventional statistical software, statistical analyses could become more accessible through natural-language interaction and thereby support broader and more efficient use of quantitative methods in medical research.

The present study evaluated whether contemporary LLMs can serve as a viable substitute for conventional statistical tools in a standardized hypothesis-testing task. Our results show that, despite remarkable capabilities, none of the evaluated models—ChatGPT 5.4 Thinking, Gemini 3.1 Pro Preview, Claude Opus 4.6, Llama 4 Scout, Apertus 8b, Qwen 3.5, and Phi 4—can currently be considered a viable substitute for statistical software. The main reason is that none of the models reproduced the full reference analysis generated by conventional software, most notably because none applied Yates’ continuity correction, which is relevant for small 2×2 contingency tables with one degree of freedom.

At the same time, the evaluated models differed substantially in quality. Gemini achieved the strongest overall documentation and performance, followed closely by Claude Opus and ChatGPT. These models were generally able to identify the correct test, formulate the hypotheses appropriately, and provide plausible analyses and interpretations. However, they still deviated from the software-based reference in critical aspects of statistical execution. ChatGPT was less transparent than Gemini and Claude Opus because it provided fewer calculation details and a less nuanced discussion of the data distribution. Gemini and Claude Opus therefore appear more useful in practice when expert users require both methodological guidance and interpretability.

The open-weight and compact models performed less reliably. Llama 4 Scout and Qwen 3.5 showed substantial weaknesses in reconstructing the dataset and reporting the necessary statistical values, which makes them unsuitable for reliable use in the present task. Apertus 8b and Phi 4 showed partial methodological understanding and were able to provide some correct theoretical el-

ements, but their analyses were undermined by incorrect assumptions about the underlying data. These findings suggest that smaller or less capable models may still offer limited educational or explanatory value, but they cannot currently support statistical analysis with sufficient reliability.

Taken together, the findings indicate that current LLMs are better understood as supportive assistants than as replacements for conventional statistical software. They may help users identify suitable tests, structure an analysis, and interpret outputs, but the calculations should still be performed by statistical tools. Statistical literacy therefore remains necessary to verify analyses and ensure high-quality medical research.

This study has several limitations. First, the benchmark was restricted to a single use case involving a χ^2 -test of independence. The generalizability of the findings to other forms of hypothesis testing, more complex datasets, or other analytical settings therefore remains uncertain. Second, the study used a single standardized prompt based on a zero-shot chain-of-thought strategy. Different prompting strategies may lead to different levels of performance and should therefore be investigated systematically. Third, the evaluation was conducted by the author, although model outputs were cross-checked against software-based reference results to increase consistency.

Future work should extend this benchmark in three directions. First, a broader range of statistical tasks should be evaluated, including t -tests, non-parametric tests, regression models, and multivariable analyses. Second, future studies should compare different prompting strategies, including iterative prompting, tool-augmented prompting, and workflows in which LLMs interact directly with statistical software. Third, it would be valuable to examine not only whether LLMs can reproduce statistical results, but also whether they improve the analytical performance of human users in realistic research settings. Such work would help clarify whether the most promising role of LLMs lies not in replacing statistical software, but in augmenting statistical reasoning and decision-making in practice.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero

- Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. *Phi-4 technical report*. *arXiv preprint arXiv:2412.08905*.
- Luigi De Angelis, Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi, and Caterina Rizzo. 2023. *Chatgpt and the rise of large language models: the new ai-driven infodemic threat in public health*. *Frontiers in Public Health*, 11.
- Anthropic. 2026. Introducing Claude Opus 4.6. <https://www.anthropic.com/news/claude-opus-4-6>. Accessed: 2026-05-04.
- Armando Crottogini, Bianca Schenk, and Yves Staudt. 2026. Rolling towards resilience: The impact of scooter riding on pediatric knee health.
- M Fay and E Brittain. 2022. *Statistical hypothesis testing in context*. pages –.
- Larry Gonick and Woolcott Smith. 2005. *The Cartoon Guide to Statistics*. HarperCollins.
- Google DeepMind. 2026. Gemini 3.1 Pro model card. <https://deepmind.google/models/model-cards/gemini-3-1-pro/>. Accessed: 2026-05-04.
- Mirjam Annina Jenny, Niklas Keller, and Gerd Gigerenzer. 2018. Assessing minimal medical statistical literacy using the quick risk test: a prospective observational study in germany. *BMJ Open*, 8:e020847.
- Camille Lakhli, François-Xavier Lejeune, Marion Rouault, Mehdi Khamassi, and Benjamin Rohaut. 2023. Illusion of knowledge in statistics among clinicians: evaluating the alignment between objective accuracy and subjective confidence, an online survey. *Cognitive Research: Principles and Implications*, 8:23.
- Siyu Liu. 2025. *Modern developments in hypothesis testing with emphasis on computational techniques*. *Theoretical and Natural Science*, pages –.
- D Mark, Kerry Lee, and F Harrell. 2016. Understanding the role of p values and hypothesis tests in clinical research. *JAMA cardiology*, 1 9:1048–1054.
- Meta AI. 2025. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation. <https://ai.meta.com/blog/llama-4-multimodal-intelligence/>. Model card: <https://huggingface.co/meta-llama/Llama-4-Scout-17B-16E-Instruct>.
- Božana Nikolić and Tamara Popovic. 2024. *Hypothesis testing and statistical test selection: Fundamentals of statistics in clinical studies - part ii*. *Medical review*, pages –.
- OpenAI. 2026. Introducing GPT-5.4. <https://openai.com/index/introducing-gpt-5-4/>. Accessed: 2026-05-04.
- C. O’Dushlaine. 2019. *Hypothesis testing and confidence intervals*, pages 523–626.
- Sharad Patel and Adam Green. 2025. *Death by p-value: the overreliance on p-values in critical care research*. *Critical Care*, 29:73.
- Qwen Team. 2026. Qwen3.5: Towards native multimodal agents. <https://qwen.ai/blog?id=qwen3.5>. Accessed: 2026-05-04.
- Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2025. *A systematic survey of prompt engineering in large language models: Techniques and applications*.
- Swiss AI Initiative. 2025. *Apertus: Democratizing open and compliant LLMs for global language environments*. *arXiv preprint arXiv:2509.14233*. Developed by EPFL, ETH Zurich, and CSCS.
- Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. *Large language models in medicine*. *Nature Medicine*, 29:1930–1940.
- D Turner, H Deng, and T Houle. 2020. *Statistical hypothesis testing: Overview and application*. *Headache: The Journal of Head and Face Pain*, 60:–.
- Sirui Zhang. 2025. *Analysis and optimization of the applicability of hypothesis testing methods*. *Advances in Operation Research and Production Management*, pages –.

7 Appendix

The complete benchmark prompt, the input data, and all model outputs are available in a closed GitHub https://github.com/staudtyves/llm_my_statistical_software repository for the purpose of reproducibility and further inspection. Access to the repository can be granted by the authors upon request.

Automated German Alt Text Generation for News Charts

Alessia Vannini^{1,+}, Julia Locher^{1,+}, Marianne Santaholma^{1,*},
Claudia Amsler¹, Jonas Oesch², Arzu Coltekin^{1,*}

¹University of Applied Sciences and Arts Northwestern Switzerland FHNW

²Neue Zürcher Zeitung, Zurich, Switzerland

⁺Shared first authorship with equal contributions ^{*} Shared senior authorship

Correspondence: marianne.santaholma@fhnw.ch and arzu.coltekin@fhnw.ch

Abstract

We investigate whether a Multimodal Large Language Model (MLLM) can automatically generate well-formed German chart alt texts that meet the requirements of visually impaired persons while following the accessibility guidelines for chart alt texts, and match the quality of manually authored gold-standard alt texts. Focusing on bar, line, and stacked bar charts from a German-language newspaper (the *Neue Zürcher Zeitung*), we define an alt text structure, construct a gold-standard corpus, and evaluate MLLM-generated chart alt texts both quantitatively on semantic similarity and qualitatively with visually impaired persons as participants in terms of clarity, conciseness, meaningfulness, and output consistency.

1 Introduction

Data visualizations such as line and bar charts, and other graphical representations play a central role in communicating complex information efficiently (Çöltekin et al., 2021). Yet these visual elements are frequently inaccessible to people with visual impairments (PVI), primarily due to the absence of alternative text descriptions, commonly referred to as alt texts. Alt texts are plain text descriptions embedded in digital documents and web pages that are rendered by screen readers, the assistive technology most widely used by PVI (AbilityNet, 2025). Alt texts can be further rendered as large print, braille, speech, symbols, or simpler language (Accessibility Guidelines Working Group (AG WG), 2023).

The provision of alt texts for all non-text digital content has recently been formalized into a legal requirement in the EU: the European Accessibility Act (EAA), in effect since June 2025, mandates that digital media providers make non-text content such as images, charts, and graphics accessible through appropriate text alternatives (European Union, 2019). WebAIM reports that 55.5% of the one million most-visited web pages were

still missing alternative text for images in 2025 (WebAIM, 2026). This gap is not simply a matter of negligence or lack of awareness. Manual alt text authoring is time-consuming and impractical, compounded by a lack of established guidelines for what constitutes a good chart description (Gleason et al., 2019). This challenge is especially acute in high-volume publishing environments such as online newspapers, where fast-paced editorial workflows leave little room for annotating every visual element.

Recent advances in natural language generation (NLG), particularly through Large Language Models (LLMs) and Multimodal Large Language Models (MLLMs), offer a promising avenue for automating chart alt text generation (Hoque and Islam, 2025; Nylund et al., 2025). These models can process visual input alongside textual prompts to produce descriptive natural language output, potentially satisfying accessibility requirements while reducing editorial burden. However, generating high-quality alt texts for charts demands more than visual understanding: it requires accurate interpretation of quantitative data and graphical conventions, as well as clear criteria for what constitutes a good alt text per chart type (Yan et al., 2025; Kantharaj et al., 2022).

We address the above-mentioned challenges through a case study in this paper, i.e., by investigating whether an MLLM can automatically generate well-formed German chart alt texts that meet the needs of PVI, conform to the set accessibility guidelines (Accessibility Guidelines Working Group (AG WG), 2023), and match the quality of manually authored gold-standard alt texts. The study focuses on bar, line, and stacked bar charts published in the *Neue Zürcher Zeitung* (NZZ), a national German-language newspaper in Switzerland. We define a structured alt text schema, construct a gold-standard corpus, and evaluate MLLM-generated alt texts against it in terms of clarity,

conciseness, meaningfulness, and output consistency.

The following sections describe the related work, methodology, evaluation setup and results. We conclude with discussion and conclusions.

2 Background and Related Work

This section reviews what constitutes a well-formed chart alt text and the state of the art in automated chart alt text generation.

2.1 Chart Alt Text Structure

The alt text serves to convey the information encoded in a chart to PVIs, rendered via screen readers. However, no standard definition of alt text for different chart types exists. Researchers and accessibility organizations have developed guidelines and best practices to support digital content creators in writing effective chart descriptions (e.g., [Accessibility Guidelines Working Group \(AG WG\), 2023](#); [DIAGRAM Center, 2020](#); [Pennsylvania State University, 2024](#); [Consumer Financial Protection Bureau](#)). Accessibility research emphasizes that effective chart alt texts rely on a clear and consistent structure that enables users to reconstruct a mental model of the chart (e.g., [Belle et al., 2022](#); [Jung et al., 2022](#)). With respect to content ordering, prior work suggests beginning with declaring the chart type as an orienting element, followed by contextual information such as axis labels and value ranges, and concluding with the main message or trend conveyed by the data ([Jung et al., 2022](#); [Yan et al., 2025](#)). A further structural consideration concerns the separation of short and long descriptions: since longer descriptions cannot be paused or skipped during screen reader playback, clearly distinguishing between the two allows users to decide whether to engage with the more detailed account. Beyond descriptions, providing a machine-readable data table containing all values displayed in the chart has been identified as a high-priority requirement among PVIs ([Jung et al., 2022](#)). This body of work informed our development of a structural model for chart alt texts (see Figure 1).

The model specifies the components a well-formed alt text should contain. It does not, however, prescribe how individual elements should be described across different chart types, how numerical values and comparisons should be verbalized, or how alt texts should scale with varying levels of chart complexity. The present work refines these

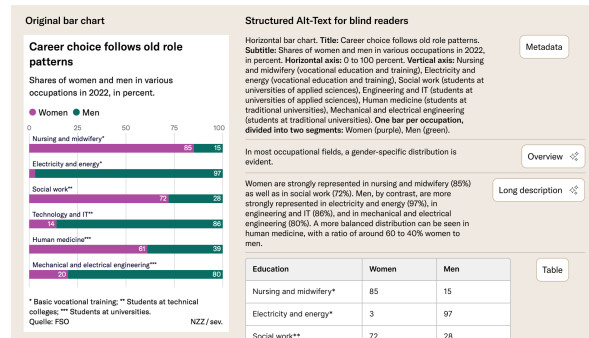


Figure 1: Chart alt text structure, illustrated with an example bar chart on the left and its structured alt text on the right (metadata, overview, long description, and data table).

structural components and their contents for specific chart types and their subcategories. We validated and iteratively refined the model in collaboration with PVIs.

2.2 Automated Chart Alt Text Generation

Automated chart alt text generation draws on a broader cluster of related research areas, including chart-to-text generation, chart question answering, and chart understanding ([Huang et al., 2025](#)). Chart-to-text generation focuses on producing natural language summaries of chart content, while chart question answering evaluates whether models can correctly respond to queries about specific data points or trends ([Obeid and Hoque, 2020](#); [Masry et al., 2022](#)). Chart understanding encompasses both tasks (with data extraction and chart type classification), addressing how well models can parse the visual and semantic structure of a chart.

Early work relied on rule-based or template-driven methods that extracted data from structured chart sources and filled predefined description templates ([Farahani et al., 2023](#)). While reliable for simple chart types, these approaches lack flexibility and cannot generalize to novel or complex visualizations ([Obeid and Hoque, 2020](#)).

The emergence of LLMs and MLLMs has opened new possibilities for chart alt text generation. These models can interpret chart images, identify salient trends, and produce fluent natural language descriptions, even without having access to the underlying data ([Wang et al., 2024](#); [Yin et al., 2024](#); [Kantharaj et al., 2022](#); [Moured et al., 2024](#); [Huang et al., 2025](#)). Recent studies have demonstrated the potential of (M)LLM-based pipelines for generating alt texts that approach human-authored quality from a language-structure

and fluency point of view (Obeid and Hoque, 2020; Kantharaj et al., 2022). Nevertheless, challenges persist on multiple fronts: on the operational side, effective prompt design and robust evaluation criteria remain open problems; on the output side, issues of numerical inaccuracy, hallucination, and appropriate descriptive granularity continue to limit practical deployment (Obeid and Hoque, 2020).

Existing literature and tools we identified on automated chart alt text generation focus on English-language (such as VisText (Tang et al., 2023) and Alt4Blind (Moured et al., 2024)), presenting a gap for other languages. We thus extend this line of research to German in this study, for which, to the best of our knowledge, no comparable work currently exists, including a gold-standard corpus. Moreover, existing approaches primarily focus on training new models and benchmarking them against established datasets such as Statista (Gong et al., 2019; Statista), without taking accessibility guidelines into account (Obeid and Hoque, 2020; Belle et al., 2022; Balaji et al., 2018; Gong et al., 2019).

3 Methodology

The overall methodological framework and workflow are illustrated in Figure 2. The workflow begins with data preprocessing, followed by a manual construction of a gold-standard alt text corpus. The alt texts are grounded in a structured alt text template as shown in Figure 1. Alt texts are generated using an MLLM and few-shot prompting. The evaluation strategy combines qualitative feedback and Likert-scale ratings collected from PVI and quantitative metrics including semantic similarity comparisons and an LLM-as-a-judge approach.

3.1 NZZ News Charts

The dataset consists of 168 charts provided by the NZZ. Each chart is provided as a PNG image accompanied by a JSON file containing metadata and underlying data values. Of the 168 charts, 150 fall within the scope of this study, covering line, bar, and stacked bar charts; the remaining 18 belong to other chart types. We categorized the charts as either *simple* or *complex* based on the number of columns in the underlying dataset: simple charts contain a single data column, while complex charts contain two or more, resulting in additional categorical dimensions such as multiple lines or grouped bars. Examples of simple and complex line, bar

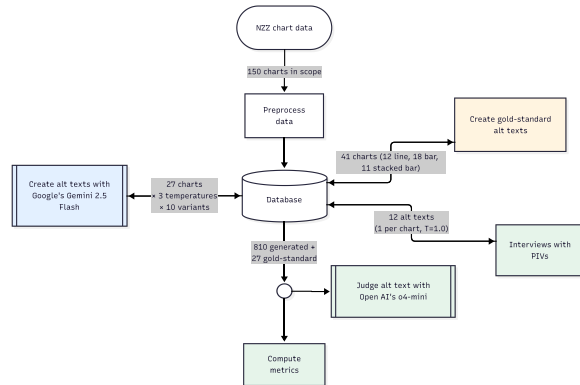


Figure 2: Overview of the alt-text generation and evaluation workflow: gold-standard creation (orange), alt-text generation with Gemini 2.5 Flash (blue), and evaluation via computed metrics, LLM-as-a-judge with o4-mini, and PVI interviews (green).

and stacked bar charts are shown in Figure 3.

Most bar charts fall into the *simple* category (46 simple, 24 complex) and can be oriented horizontally or vertically, with either time-based or categorical axes. Line charts, by contrast, have more of the *complex* category (23 simple, 39 complex) and always display a time series on the x-axis. Stacked bar charts are the smallest group and are mostly complex (5 simple, 13 complex). They are further subdivided by whether bars sum to 100% or display absolute values, and by the number of segments. In addition to these primary structures, several charts employ supplementary visual encodings such as highlighted regions, prognosis markers, and annotated events, which add further descriptive requirements for alt text generation.

A preprocessing pipeline parses the JSON files, extracts metadata and data values, and converts them into CSV files stored in a relational database. This pipeline can be extended to other chart types.

3.2 gold-standard Data

At the preliminary stages of this project, we used Statista’s open-source dataset (Statista) as a proxy gold-standard. However, this dataset has several shortcomings that limit its suitability for systematic benchmarking. The texts do not distinguish between short and long descriptions, chart metadata (such as chart type, axis labels, and value ranges) is frequently incomplete or absent, the text includes information that is not included in the chart, and the descriptions are assumed to function as monolithic long-form accounts rather than structured, modular alt texts. The discrepancies with accessibility

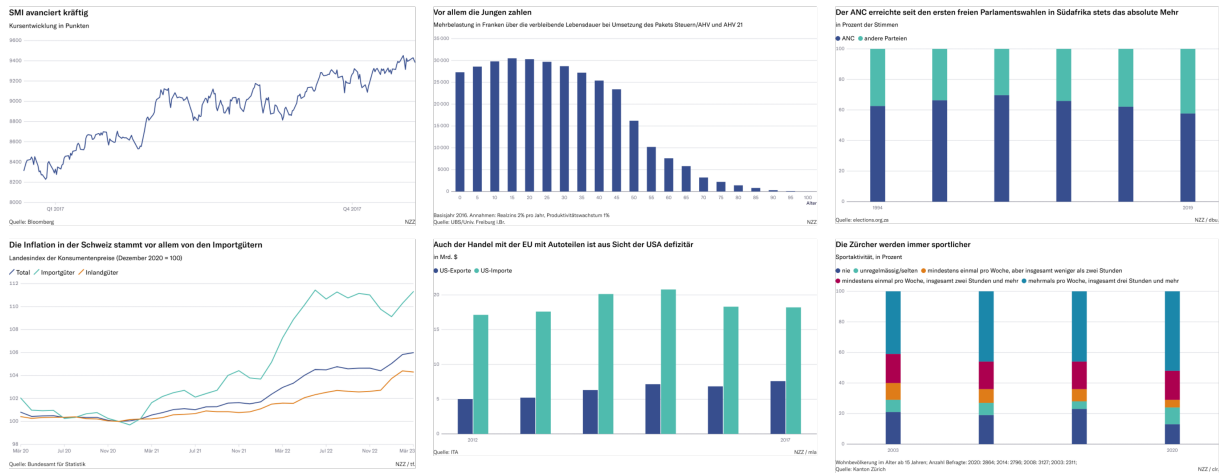


Figure 3: Representative examples from the NZZ chart dataset, organized by chart type and complexity. Columns correspond to line, bar, and stacked bar charts. The top row shows simple charts (single data column), and the bottom row shows complex charts (two or more data columns).

guidelines make the dataset ill-suited as a reference for structured chart alt text generation. Therefore, we manually created a German gold-standard corpus. The corpus covers 41 NZZ charts selected to represent a wide range of subcategories and visual encodings: 12 line charts (5 simple, 7 complex), 18 bar charts (10 simple, 8 complex) and 11 stacked bar charts (5 simple, 6 complex).

We developed the corpus through three iterative revision stages. In the first stage, we drafted initial alt texts with the assistance of Open AI’s GPT 5.1 and internally validated them through a sketching exercise, in which one author produced structural reconstructions of the charts based solely on the written descriptions as a check for completeness and clarity. In the second stage, we refined the texts through a linguistic consultation with an expert in German language and simple-language writing, focusing on neutral phrasing and readability, in line with prior work (Jung et al., 2022). In the third and final stage, we presented the alt texts to five PVIs in structured interviews. They assessed how well the descriptions supported the mental reconstruction of charts across types and complexity levels.

The feedback from PVIs directly shaped the final revision. Key changes included: (a) rendering the short description as metadata in a technical rather than fluent prose style; (b) repositioning the overview summary at the end of the short description rather than at the beginning of the long description; (c) long descriptions follow a chronological progression designed to support mental reconstruction without overloading the listener with detail; and (d) reducing numerical precision where

a data table was available, while retaining exact figures otherwise. More broadly, PVIs emphasized a strong preference for having access to the same information as sighted users, and for a consistent basic structure applied uniformly across all charts.

This final version constitutes the gold-standard reference data that we used for all subsequent evaluation in this study.

3.3 Generation of the Chart Alt Texts

We generated the alt texts using Google’s Gemini 2.5 Flash Preview 09-2025 (Google DeepMind, 2025), an MLLM capable of processing both textual and visual input. The model is accessed via the OpenRouter API (OpenRouter), which supports a temperature range of 0.0 to 2.0. Temperature governs the degree of randomness in model output: a value of 0.0 yields deterministic outputs for identical inputs, while higher values increase variability. We set the temperature parameter to the model default of 1.0. Future work should evaluate the effect of temperature on alt text quality, consistency, and factual accuracy, for instance by comparing outputs at $T=0.0$ and $T=1.0$ through human evaluation. We selected Gemini 2.5 Flash as it could be integrated into the NZZ data pipeline, facilitating the possible adoption in an operational publishing context.

As inference strategy we employed **multimodal few-shot prompting**: each request supplies the model with a chart image, a CSV excerpt of the underlying data, natural language instructions, and a set of illustrative gold-standard examples, from which the model is expected to infer the desired output structure and style without any additional

fine-tuning.

The prompt design is based on the systematic trials that we conducted in our preliminary work (Locher and Vannini, 2025). We demonstrated that structured prompts with explicit formatting constraints consistently outperform free-form instructions for chart alt text generation. In particular, prompts that clearly separate short and long descriptions, impose length constraints, and adapt to chart type and complexity through type-specific instructions and examples yield the highest overall output quality.

We designed six distinct prompt variants for this study: one for each combination of chart type (line, bar, stacked bar chart) and complexity level (simple, complex). Each variant encodes chart-type-specific rules, content ordering constraints, and wording guidelines. They all follow a shared template structure comprising the following components:

Task definition Explicit instructions for generating the short description (metadata-based), the overview, and the long description, including word limits.

Examples Illustrative alt texts drawn from the gold-standard corpus. Of the 41 gold-standard alt texts available, we selected 14 as in-prompt examples. These cover the full range of chart types (12) and their subcategories. For complex bar charts, we distinguished three subcategories: 1) category within a time series, 2) time series within a category, and 3) category within a category. For simple stacked bar charts, we defined the following three variants: 1) 100% charts with multiple bars, 2) 100% charts with a single bar, and 3) charts with absolute values. Each subcategory required a dedicated example, resulting in three in-prompt examples for these two chart types rather than default two. All remaining chart types retained two in-prompt examples each.

Data CSV excerpt of the underlying chart data extracted during preprocessing.

Stylistic rules Guidelines governing tone, phrasing, and language use. The stylistic rules reflect both the accessibility requirements and the feedback that we gathered from PVIs. Descriptions are required to be concise and clear, and causal interpretations are explicitly prohibited in favor of a neutral, observational tone. Trend descriptions are restricted to the verbs *steigen* and *sinken*, avoiding

evaluative or dramatic alternatives such as *erreichen* or *einbrechen*. Symbols are written out, e.g. negative values are expressed using the word *minus* rather than the minus symbol, as screen readers may misinterpret them.

Fixed output format A fixed output structure separates the short description, overview, and long description using defined delimiters, ensuring consistent formatting and reliable downstream parsing and evaluation.

4 Evaluation Setup

The evaluation strategy combines automatic reproducible metrics with qualitative human feedback. It addresses the central question of how MLLM-generated German chart alt texts compare to manually created gold-standard alt texts in terms of clarity, conciseness, meaningfulness, and output consistency. The evaluation is organized along three dimensions:

- **Semantic similarity:** the degree of semantic overlap between generated and gold-standard alt texts, assessed using sentence embeddings.
- **Text length as a proxy for conciseness:** comparison of character counts across generated and gold-standard alt texts. Text length serves as an objective and reproducible approximation of conciseness, as longer texts are more likely to contain redundant phrasing or unnecessary detail.
- **Quality judgements:** a comparison of LLM-based and human rating across shared evaluation criteria.

The evaluation is based on 27 gold-standard chart alt texts. A total of 282 MLLM-generated alt texts were produced for evaluation: 270 from ten generation runs per chart under identical conditions (27×10 , temperature = 1.0), plus 12 additional texts generated specifically for human evaluation. We generated these 12 additional texts in an initial pipeline validation run, using a balanced subset of the 27 benchmarking charts (two per chart type and complexity level). These texts served as stimuli in the PVI interviews (See 3.2), in which participants evaluated MLLM-generated alt texts rather than the gold-standard texts. Presenting gold-standard texts to PVIs would have introduced a methodological circularity, as the gold-standards texts were

iteratively refined based on prior PVI feedback. Alt texts are generated in a strict output format and parsed into sections. All metrics are computed separately for the full text, the short description (metadata), the short description (overview), and the long description, when relevant.

Semantic similarity was computed as the cosine similarity of sentence embeddings extracted using SBERT (Reimers and Gurevych, 2019). Embeddings were computed separately for four textual units: the full alt text, the short description (metadata), the short description (overview), and the long description. High cosine similarity indicates substantial semantic overlap with the gold-standard, while lower values reflect divergence in content or phrasing.

Text length was assessed through descriptive statistics of character and word counts per section, comparing gold-standard and generated texts across chart types and complexity levels.

LLM-as-a-judge evaluation (Gu et al., 2025) was conducted using OpenAI’s o4-mini model accessed via OpenRouter (OpenAI, 2024) to rate each generated alt text along six criteria: clarity, conciseness, neutrality, perceived completeness, completeness, and correctness. Ratings were produced on a Likert scale and compared against human ratings on the shared subset of criteria. We treated completeness and perceived completeness as distinct criteria. Completeness assesses whether the alt text factually covers the essential elements of the chart (verifiable against the chart image and underlying CSV data), and can therefore only be evaluated by the LLM judge. Perceived completeness, by contrast, captures the subjective impression of sufficiency, i.e. whether the text appears to cover everything important without requiring visual access to the chart. We introduced this distinction to enable PVI participants to provide meaningful completeness-related judgments despite their visual impairment.

Human evaluation was conducted with five PVIs across 12 charts ($12 \times 5 = 60$ ratings). Prior to the evaluation, participants received a single verbal explanation of the four rating criteria (neutrality, clarity, conciseness, and perceived completeness) to ensure a shared understanding. We excluded completeness and correctness from the human evaluation as they require direct visual access to the chart and underlying data.

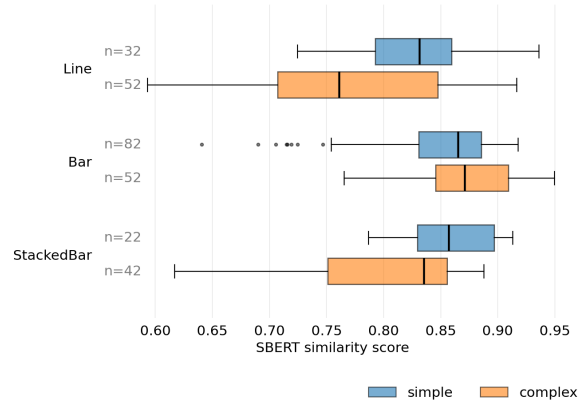


Figure 4: Overall SBERT similarity distribution across chart types and complexity levels.

5 Results

The results for quantitative and qualitative evaluations are detailed in the following sections.

5.1 Semantic Similarity

Figure 4 shows the overall similarity distribution aggregated across all alt text sections.

Across all three chart types, simple charts generally yield higher and more concentrated similarity scores than complex ones, with the exception of bar charts, where simple and complex conditions show comparable medians and interquartile ranges. Complex line and stacked bar charts show the lowest medians and widest interquartile ranges, reflecting the greater challenge of summarizing multi-layered visualizations. Simple bar charts exhibit several low-scoring outliers, though the bulk of scores remains above 0.80. When broken down by alt text section (not shown), the overview and long description sections diverge more strongly from the gold-standard and exhibit higher variance, which is expected given that these sections require abstraction and trend prioritization rather than extraction of fixed metadata.

5.2 Text Length and Conciseness

MLLM-generated texts are consistently longer than gold-standard alt texts and show a heavier tail of very long outputs, particularly for complex charts.

5.3 LLM-Judge and Human Evaluation

Overall, both evaluators assign high scores across shared criteria, indicating broad consensus that the generated chart alt texts meet essential quality standards. Median scores are closely aligned for clarity and perceived completeness, suggesting that

LLM judgements approximate human assessments on these dimensions. However, the LLM exhibits greater score variability and applies stricter criteria for conciseness and neutrality than PVIs.

Human scores show the greatest variation in **clarity**. PVIs awarded high clarity ratings when terminology was simple, structures were explicit, and trends were described sequentially. Lower scores were associated with special terminology, ambiguous references, screen reader incompatibilities, and descriptions that assumed the reader could refer back to the chart visually. No systematic directional bias was observed in LLM clarity scores relative to human ratings.

Conciseness is the dimension where LLM-judge and human evaluation diverge the most. The LLM applies stricter internal standards, while PVIs tend to tolerate longer descriptions if they are perceived as informative. Users lowered conciseness ratings when descriptions were numerically saturated, redundant, or structured as long enumerations. These were reported as cognitively demanding. Conversely, very short descriptions occasionally raised concerns about missing context.

Neutrality scores are high and closely aligned for simple charts. Minor discrepancies emerge for complex charts, where the LLM occasionally penalizes formulations that PVIs still consider neutral. User feedback suggests that neutrality concerns arose less from evaluative language than from structural choices, such as consistently pairing certain entities together in a way that implied an implicit framing.

Perceived completeness ratings are similarly distributed across both evaluators, though the LLM shows a slightly wider score range. For PVIs, perceived completeness depended primarily on whether the description conveyed a coherent overall picture rather than exhaustive detail. High scores were given when PVIs could mentally reconstruct the main trends and relationships; lower scores occurred when key contextual anchors, such as start or end values, time references, or explicit statements about the absence of relationships, were missing.

5.4 Inter-Generation Consistency

Figure 5 shows pairwise cosine similarity scores across ten generation runs per chart, aggregated across all alt text sections.

Complex charts show somewhat broader distributions, indicating greater output variability when

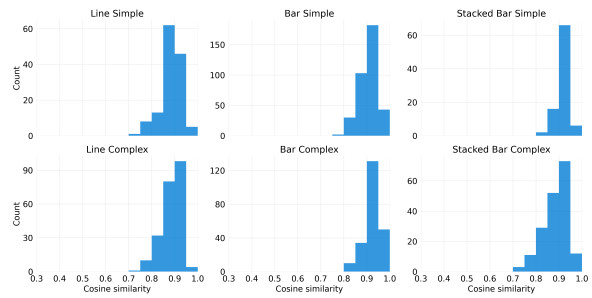


Figure 5: Pairwise cosine similarity scores across ten generation runs, broken down by chart type and complexity.

the model must describe multi-layered visualizations. The overall stability is largely driven by the metadata section, which is nearly deterministic across runs. When examined by section (not shown), the overview and long description exhibit markedly greater variability, particularly for complex charts, confirming that output instability is concentrated in the sections requiring abstraction and trend prioritization rather than in structured factual components.

6 Discussion

The results converge on a coherent picture of the strengths and limitations of MLLM-generated chart alt texts. Across all evaluation dimensions, the metadata section emerges as the most robust component: similarity to the gold standard is high and stable, length is well-controlled, and outputs are nearly deterministic across generation runs. This consistency reflects the structured, rule-governed nature of metadata content, which is directly derivable from the underlying chart data.

In contrast, the overview and long description sections are less stable. Semantic similarity to the gold standard is lower and more variable, the generated texts tend to be longer than their manually authored counterparts, and the output consistency across repeated runs is reduced, particularly for complex charts. These patterns are expected since both sections require the model to abstract, prioritize, and synthesize information rather than simply extract it. However, the variability they introduce poses a challenge for deployment in contexts where predictable and consistent output is required.

The quality evaluation broadly corroborates these findings. Both LLM and human evaluators rate the generated texts highly for clarity and completeness, suggesting that the outputs are generally

well-formed and informative. Conciseness is the dimension that most clearly differentiates the generated texts from gold-standard texts, with LLM-generated outputs consistently running longer, particularly for complex charts. The alignment between LLM-judge and PVI ratings is encouraging and suggests that LLM-based evaluation could be useful at least for clarity and perceived completeness. The LLM applies stricter standards for conciseness and neutrality.

A key implication of these findings concerns the role of MLLMs in the generation pipeline. The metadata section, whose content is fixed and deterministic, could be better handled by rule-based templates, which would eliminate the residual variability observed even in this structured section. MLLM generation is most valuable for the overview and long description, where flexible natural language generation adds the greatest benefit and no deterministic alternative exists. Taken together, the findings support a hybrid approach: deterministic template generation for metadata, and carefully prompted MLLM generation for higher-level descriptions. Even in its current form, however, automatically generated alt text represents a substantial improvement over its absence and brings chart accessibility within practical reach for high-volume publishing environments.

A limitation of the current study is the relatively small evaluation set and pool of PVI participants. The findings are based on 27 gold-standard charts and five PVIs, which constrains the statistical robustness of the human evaluation results, even though the effort provides a good starting point and qualitative insights are valuable for this line of work. Additionally, the scope is limited to three chart types from a single German-language publisher, which may limit generalizability, and the results should be seen as the outcomes of preliminary work on a valid, real-world case study.

7 Conclusions

This study investigated whether MLLMs can automatically generate well-formed German chart alt texts that meet the needs of PVIs and match the quality of manually authored gold-standards. Focusing on bar, line, and stacked bar charts from the *Neue Zürcher Zeitung* ([Neue Zürcher Zeitung](#)), we defined a structured alt text schema, constructed a gold-standard corpus in iterative collaboration with PVIs, and evaluated MLLM-generated descriptions

in terms of clarity, conciseness, meaningfulness, and output consistency.

The results demonstrate that MLLM-generated alt texts are a viable and practical solution for improving chart accessibility at scale. Generated texts were consistently rated as clear and complete, and PVIs perceived them as providing meaningful added value. Performance was strongest for simpler charts, where generation was more stable and outputs more closely aligned with the gold-standard. The evaluation also identified a clear division of labor: metadata-based short descriptions could potentially be better generated via deterministic templates, while MLLMs are most effectively employed for overviews and long descriptions where linguistic generation is required. A hybrid pipeline combining both approaches is a practical and scalable strategy.

Several directions for future work emerge from this study. First, expanding the gold-standard corpus to cover a broader range of chart types, complexity levels, and source domains would improve generalizability. Second, enlarging and diversifying the pool of PVI participants would strengthen the statistical robustness of the human evaluation. Third, integrating inter-generation consistency analysis directly into the evaluation pipeline would support ongoing quality monitoring in production settings.

Limitations

The qualitative evidence is based on a small sample of five PVIs and a limited set of charts. Thus, the findings may not represent the full diversity of screen-reader use and information needs. Furthermore, SBERT similarity measures semantic overlap, but it cannot assess structural compliance, cognitive load, or screen-reader-specific issues. Complex charts remain the main challenge: even accurate alt texts can be cognitively demanding.

Acknowledgments

We would like to thank the *Schweizerischer Blinden- und Sehbehindertenverband* (SBV) and all participants who contributed to the refinement of the gold-standard corpus and the evaluation of the generated alt texts, as well as the *Neue Zürcher Zeitung* (NZZ) for providing the chart dataset used in this study.

References

- AbilityNet. 2025. An introduction to screen readers. <https://abilitynet.org.uk/factsheets/introduction-screen-readers>.
- Accessibility Guidelines Working Group (AG WG). 2023. Web Content Accessibility Guidelines (WCAG) 2.2 Quick Reference. <https://www.w3.org/WAI/WCAG22/quickref/>.
- Abhijit Balaji, Thuvaarakkesh Ramanathan, and Venkateshwarlu Sonathi. 2018. *Chart-Text: A Fully Automated Chart Image Descriptor*. *arXiv preprint*.
- Aspen Belle, Vanessa Goh, Akshay Kumar, Richard Pranjatno, Pui Yip, Umayangani Wickramaratne, and Humphrey Obie. 2022. *Alt-Textify: A Pipeline to Generate Alt-text from SVG Visualizations*. In *Proceedings of the 17th International Conference on Evaluation of Novel Approaches to Software Engineering*, pages 275–281, Online Streaming, — Select a Country —. SCITEPRESS - Science and Technology Publications.
- Arzu Çöltekin, Amy Griffin, and Anthony Robinson. 2021. *Visualizations*. Oxford University Press.
- Consumer Financial Protection Bureau. Data visualization guidelines. <https://cfpb.github.io/design-system/guidelines/data-visualization-guidelines>.
- DIAGRAM Center. 2020. Specific Guidelines - Graphs.
- European Union. 2019. Directive (EU) 2019/882 of the European Parliament and of the Council of 17 April 2019 on the accessibility requirements for products and services.
- Ali Mazraeh Farahani, Peyman Adibi, Mohammad Saeed Ehsani, Hans-Peter Hutter, and Alireza Darvishy. 2023. *Automatic Chart Understanding: A Review*. *IEEE Access*, 11:76202–76221.
- Cole Gleason, Patrick Carrington, Cameron Cassidy, Meredith Ringel Morris, Kris M. Kitani, and Jeffrey P. Bigham. 2019. “It’s almost like they’re trying to hide it”: How User-Provided Image Descriptions Have Failed to Make Twitter Accessible. In *The World Wide Web Conference*, pages 549–559, San Francisco CA USA. ACM.
- Li Gong, Josep Crego, and Jean Senellart. 2019. *Enhanced Transformer Model for Data-to-Text Generation*. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 148–156, Hong Kong. Association for Computational Linguistics.
- Google DeepMind. 2025. Gemini 2.5 Flash (Preview).
- Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. *A Survey on LLM-as-a-Judge*. *Preprint*, arXiv:2411.15594.
- E. Hoque and M. Saidul Islam. 2025. *Natural Language Generation for Visualizations: State of the Art, Challenges and Future Directions*. *Computer Graphics Forum*, 44(1):e15266.
- Kung-Hsiang Huang, Hou Pong Chan, May Fung, Haoyi Qiu, Mingyang Zhou, Shafiq Joty, Shih-Fu Chang, and Heng Ji. 2025. *From Pixels to Insights: A Survey on Automatic Chart Understanding in the Era of Large Foundation Models*. *IEEE Transactions on Knowledge and Data Engineering*, 37(5):2550–2568.
- Crescentia Jung, Shubham Mehta, Atharva Kulkarni, Yuhang Zhao, and Yea-Seul Kim. 2022. *Communicating Visualizations without Visuals: Investigation of Visualization Alternative Text for People with Visual Impairments*. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):1095–1105.
- Shankar Kantharaj, Rixie Tiffany Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. 2022. *Chart-to-Text: A Large-Scale Benchmark for Chart Summarization*. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4005–4023, Dublin, Ireland. Association for Computational Linguistics.
- Julia Locher and Alessia Vannini. 2025. *Assisting People with Visual Impairments to Understand Data Visualizations*. Technical Report 25FS_IIT47, University of Applied Sciences and Arts Northwestern Switzerland (FHNW), School of Computer Science, Windisch, Switzerland.
- Ahmed Masry, Do Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. *ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning*. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.
- Omar Moured, Shahid Ali Farooqui, Karin Müller, Sharifeh Fadaeijouybari, Thorsten Schwarz, Mohammed Javed, and Rainer Stiefelwagen. 2024. *Alt4Blind: A User Interface to Simplify Charts Alt-Text Creation*. In Klaus Miesenberger, Petr Peňáz, and Makoto Kobayashi, editors, *Computers Helping People with Special Needs*, volume 14750, pages 291–298. Springer Nature Switzerland, Cham.
- Neue Zürcher Zeitung. NZZ – Neue Zürcher Zeitung. <https://www.nzz.ch>.
- Kai Nylund, Jennifer Mankoff, and Venkatesh Potluri. 2025. *MatplotAlt: A Python Library for Adding Alt Text to Matplotlib Figures in Computational Notebooks*. *Computer Graphics Forum*, 44(3):e70119.
- Jason Obeid and Enamul Hoque. 2020. *Chart-to-Text: Generating Natural Language Descriptions for Charts by Adapting the Transformer Model*. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 138–147, Dublin, Ireland. Association for Computational Linguistics.

- OpenAI. 2024. O4-mini.
- OpenRouter. OpenRouter API Reference. <https://openrouter.ai/docs/api/reference/parameters>.
- Pennsylvania State University. 2024. Charts & Accessibility.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks](#). *Preprint*, arXiv:1908.10084.
- Statista. Statista. <https://de.statista.com/>.
- Benny Tang, Angie Boggust, and Arvind Satyanarayan. 2023. [VisText: A Benchmark for Semantically Rich Chart Captioning](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7268–7298, Toronto, Canada. Association for Computational Linguistics.
- Jiaqi Wang, Hanqi Jiang, Yiheng Liu, Chong Ma, Xu Zhang, Yi Pan, Mengyuan Liu, Peiran Gu, Sichen Xia, Wenjun Li, Yutong Zhang, Zihao Wu, Zhengliang Liu, Tianyang Zhong, Bao Ge, Tuo Zhang, Ning Qiang, Xintao Hu, Xi Jiang, and 5 others. 2024. [A Comprehensive Review of Multimodal Large Language Models: Performance and Challenges Across Different Tasks](#). *arXiv preprint*.
- WebAIM. 2026. WebAIM: The WebAIM Million. <https://webaim.org/projects/million/>.
- Chuqiao Yan, Hans-Peter Hutter, Felix M. Schmitt-Koopmann, and Alireza Darvishy. 2025. [Chart Accessibility: A Review of Current Alt Text Generation](#). *IEEE Access*, 13:94040–94056.
- Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. [A survey on multimodal large language models](#). *National Science Review*, 11(12):nwae403.

Chapter 2
Corpus Track

A Dataset of Latin Etymologies Extracted from Wiktionary

Javier de Torres
Independent Researcher
Madrid, Spain

Marco Passarotti
CIRCSE Research Centre
Università Cattolica del Sacro Cuore
Milan, Italy

Francesco Mambrini
CIRCSE Research Centre
UCSC
Milan, Italy

Matteo Pellegrini
Surrey Morphology Group
University of Surrey
Guildford, United Kingdom

Giovanni Moretti
CIRCSE Research Centre
UCSC
Milan, Italy

Abstract

We present a curated resource of Latin etymologies automatically extracted from Wiktionary, enriched with links to the LiLa Knowledge Base of Latin and modelled as RDF triples using the LemonEty ontology. We also present the Python pipeline the data was generated with, as it can be reused to extract Wiktionary’s etymologies for other languages. The etymology chains cover Latin words and their attested or reconstructed ancestors in languages such as Proto-Indo-European, Proto-Italic, Ancient Greek, Hebrew, Egyptian, and others. To address the structural noise and editorial heterogeneity of Wiktionary etymology data, we have introduced strong rule-based filters throughout the pipeline, especially in the curation stage. After validation, the resulting dataset contains etymological chains for 9,684 lemmas, which can be used to support research in Historical Linguistics, Computational Etymology and language learning, among other applications.

1 Introduction

Etymological information is valuable for a wide array of applications related to lexicography, including Historical Linguistics and the construction of knowledge graphs for lexical resources, such as large multilingual semantic networks like BabelNet, which integrates lexical and encyclopedic knowledge across languages (Navigli and Ponzetto, 2012), and linguistic knowledge graphs that model relations between lexical units and concepts for digital humanities research (Basile et al., 2022). However, high-quality structured etymological datasets remain relatively scarce. Wiktionary contains a large collection of high-quality etymological descriptions contributed by its community of editors. However, these descriptions are primarily expressed in unstructured natural language. Our initial goal was to create software that took them as input and returned them in a structured format as output.

To embody our objective in a practical application, we focus our development on enriching the collection of lemmas that constitutes the core component of Linking Latin (LiLa)¹, a knowledge base of Latin linguistic resources made interoperable through their publication as Linked Open Data. To do this, we conduct the automatic extraction, normalization and structuring of Latin etymologies from Wiktionary into RDF format, and their subsequent alignment with LiLa lemmas. The main contributions of this work are (i) the resulting dataset of Latin etymological chains extracted from Wiktionary and (ii) the rule-based pipeline to generate that data. The dataset contains curated chains for 9,684 lemmas and, with respect to LiLa, provides a complementary layer of etymological information derived from a large-scale, collaboratively curated source.

In addition, the pipeline was designed following a clear separation-of-concerns principle, resulting in a highly modular workflow. Each stage of the process—extraction, enrichment, curation, and RDF serialization—can therefore be modified independently. This design makes the pipeline easily adaptable to other use cases. For instance, extracting etymologies for a different language and linking them to another knowledge graph would only require modifying the enrichment step preceding RDF serialization.

2 Related Work

Wiktionary has increasingly been used as a large-scale lexical resource for linguistic and NLP applications. Several efforts have focused on transforming its collaboratively edited content into structured data that can be processed automatically. In particular, the Wiktextextract system (Ylonen, 2022) provides a comprehensive extraction pipeline capable of expanding Wiktionary templates and Lua mod-

¹<https://lila-erc.eu>

ules, producing machine-readable representations of lexical entries and their associated linguistic information.

The Linked Open Data (LOD) paradigm promotes the publication of structured data on the Web using standard technologies such as RDF and HTTP identifiers, enabling datasets from different sources to be interlinked and reused across applications (Berners-Lee, 2006). Within the field of linguistics, this paradigm has encouraged the development of interoperable lexical resources published as Linked Open Data, allowing lexical information to be integrated into broader knowledge graphs and shared across linguistic infrastructures.

Within this context, the representation of lexical data as linked data has been extensively explored within the OntoLex-lemon framework (McCrae et al., 2017), a W3C community ontology designed to represent lexical information in RDF and to link lexical entries to their meanings and related linguistic data. In order to model etymological relations specifically, (Khan, 2018) proposed the lemonEty ontology, an extension of OntoLex-lemon designed to represent historical relationships between lexical items and their ancestors.

A number of projects have explored the publication of lexical data extracted from Wiktionary as Linked Open Data. One of the most prominent examples is DBnary (Sérasset, 2014), a large multilingual lexical resource automatically extracted from several language editions of Wiktionary and published in RDF using the lemon model. DBnary provides structured lexical information such as senses, translations, and morphological data, and has also served as a basis for experiments involving the extraction and visualization of etymological relations from Wiktionary. Unlike DBnary, which primarily focuses on the extraction of lexical information such as senses, translations and morphology from Wiktionary, the dataset presented in this paper specifically targets the reconstruction of etymological chains and their representation using the lemonEty ontology, with explicit links to the LiLa LemmaBank.

Related efforts have investigated the construction of explicit etymology graphs from Wiktionary data. For instance, the Etytree project (Pantaleo et al., 2017) extracts etymological relationships between lexical items from Wiktionary and represents them as navigable graph structures, enabling interactive exploration of word histories and cross-linguistic derivational relations. However, the pri-

mary goal of Etytree is the visualization and exploration of etymological networks rather than the publication of structured etymological data as Linked Open Data. In contrast, the approach presented in this paper focuses on the construction of a curated dataset of etymological chains represented in RDF using the lemonEty ontology and aligned with the LiLa Knowledge Base, with the aim of enabling reuse and interoperability within the Linked Open Data ecosystem for linguistic resources.

Within the context of Latin linguistic resources, the LiLa Knowledge Base of Linguistic Resources for Latin integrates multiple datasets through Linked Open Data principles. In particular, Mambrini and Passarotti (Mambrini and Passarotti, 2020) demonstrate how etymological information can be modeled in LiLa using the OntoLex-lemon ontology together with the lemonEty extension. The dataset presented in this paper follows the same modelling approach, ensuring interoperability with the existing LiLa infrastructure.

3 Source Data and Corpus Construction

3.1 Data extraction

In Wiktionary, each entry for a word has one or several etymology sections, depending on how many etymologies have been proposed for it. Expressing etymologies in natural language poses a series of challenges when it comes to extracting the desired information. However, no parser development work was necessary on our side, as Tatu Ylonen already made Wiktextextract, “the first known extractor capable of expanding Wiktionary templates and Lua modules” (Ylonen, 2022, p. 1317). In this context, Wiktionary templates are reusable pieces of wiki markup used to encode structured linguistic information within entries, such as derivation relations, inherited forms, or borrowings between languages. For instance, the template `{{inhlenlanglniht}}` encodes that the English word *night* is inherited from Old English *niht*.

Every few days, Wiktextextract is run on the English Wiktionary and the data dump is posted on <https://kaikki.org>. We downloaded the compressed .gz version that was on the site on March 16th 2026 and used its templates as our starting point. Wiktionary data has considerable size, so the best course of action is to always extract the minimal subset we intend to operate on. Wiktionary encodes etymological relations through a family

of templates that describe inheritance, derivation, borrowing, and roots, so, in this case, from the initial Wikitext dump, we extracted the etymology templates for Latin and stored them in a JSON file, in which a key is a lemma and the value is the list of etymology templates for that lemma. See, for example, the templates for *frater*:

```
[
  {"name": "inh", "args": {"1":
    "la", "2": "itc-pro", "3":
    "*frātēr"}},
  {"name": "inh", "args": {"1":
    "la", "2": "ine-pro", "3":
    "*bhréh2tēr"}}
]
```

It is worth noting that a word can have several such lists, which makes the phrase “etymology templates” ambiguous. From this point onwards, we will use “etymology templates list” to denote a single templates list, like the one just shown, and “etymologies” to denote a list of these lists.

3.2 Data transformation

In this step, we removed all extraneous information from the etymology templates and reduced them to the minimal representation required to encode an etymological relation: a chain in which each element consists of a word–language pair.

To do so, given a word’s etymologies, we iterate over each etymology templates list. For each list, we then iterate over each individual template and then classify it based on its name field:

- **Relevant:** contains the primary ancestry templates observed in the data, including inheritance (*inh*), derivation (*der*), borrowing (*bor*), and root relations (*root*), together with their extended variants (e.g. *inh+*, *der+*, *bor+*). These templates directly encode a relation between a target form and a source form in another language. Since inheritance, derivation, and borrowing all represent a step in a lexical ancestry chain, we treat them uniformly during transformation.
- **Inheritance-like:** groups together templates that behave structurally as ancestry transitions, including additional borrowing-related templates such as *ubor* (unadapted borrowing) and *s1bor* (semi-learned borrowing). Although their linguistic interpretation differs

slightly, they all encode the same structural pattern: a source form in a source language that precedes the target lemma.

- **inheritance-like (lite):** Wiktionary occasionally uses simplified or lightweight template variants such as *inh-lite*, *der-lite*, and *bor-lite*. These templates follow the same argument structure as their standard counterparts but contain less metadata.

If a template fits into any of the three classes, we then normalize it into a minimal ancestry node of the form (form, language), in which form is the template’s word’s lemma and language is its language code. Applying this procedure to all templates of an entry yields a linear etymology chain in template order. Given *frater*’s templates, see its etymology chain:

```
[(frātēr, itc-pro), (bhréh2tēr,
  ine-pro)]
```

After processing all words’ etymologies, etymology chains were obtained for 12,421 Latin words and stored as a JSON, in which every key is a lemma and the value is its etymology chains.

In Wiktionary, the same etymology templates may be repeated for a single lemma. This can occur for two main reasons. First, entries are organized into multiple sections (e.g., by part of speech), and Wikitext may treat these sections as separate entries while preserving the same etymology. For example, *Februārius* appears both as an adjective (“of February”) and as a proper noun (“February”), with both entries containing identical etymological templates. Second, a surface form may appear both as a lemma and as an inflected form of another word. For instance, *animalis* appears both as an adjective lemma and as the genitive singular of *animal*, again yielding duplicate etymology template sequences. In addition, due to the normalization process, distinct template sequences may collapse into the same ancestry chain when non-etymological templates (e.g., suffixation or gloss information) are discarded during transformation.

To avoid redundancy, duplicate chains are removed per lemma prior to curation. This step ensures that each lemma is associated with a set of unique etymological chains while preserving all distinct ancestry information. In total, 555 lemmas were found to contain duplicate etymology chains and were subsequently deduplicated.

The chains reveal that Latin lexical items directly descend from a wide range of languages and language stages. These include proto-language reconstructions such as Proto-Italic (*itc-pro*) and Proto-Indo-European (*ine-pro*); historical Indo-European languages such as Ancient Greek (*grc*), Oscan (*osc*), and Etruscan (*ett*); and languages involved in lexical borrowing, including Phoenician (*phn*), Hebrew (*hbo*), Egyptian (*egy*), and Akkadian (*akk*). The dataset also contains language variants and dialectal designations used in Wiktionary, for instance Doric Greek (*grc-dor*), Ionic Greek (*grc-ion*), and Koine Greek (*grc-koi*), as well as historical stages such as Vulgar Latin (*VL.*), Late Latin (*LL.*), and Medieval Latin (*ML.*). Beyond these immediate ancestors, the etymology chains also reveal more distant indirect ancestry from additional languages and proto-languages that do not appear as direct sources. These include further Indo-European branches and reconstructed stages such as Proto-Germanic (*gem-pro*), Proto-Slavic (*sla-pro*), Proto-Indo-Iranian (*iir-pro*), and Proto-Semitic (*sem-pro*), as well as historical languages such as Hittite (*hit*), Old Irish (*sga*), Coptic (*cop*), and Old Church Slavonic (*cu*). Together, these chains illustrate the wide diachronic and geographic span captured by the dataset, reflecting both deep Indo-European ancestry and later borrowing and transmission across multiple languages.

3.3 Data curation

Due to the collaborative nature of Wiktionary, and the diversity and heterogeneity among contributors, there is always bound to be some noise in its data. For this reason, a rule-based curation stage is used to filter potentially unreliable chains. In designing these filters, we prioritize precision over recall: the goal is to maximize the proportion of correct etymological relations (true positives) in the final dataset, even if this means discarding some valid chains. As a consequence, the curation process may introduce a higher number of false negatives, but it significantly reduces the presence of incorrect or noisy etymological links. This trade-off is considered preferable for a corpus intended to support linguistic research and knowledge graph construction, where data reliability is particularly important. Furthermore, false negatives can always be made into true positives (e.g. by an annotator), whereas the process to undo false positives is more cumbersome, especially if the data is published and

associated with URIs.

The etymologies associated with each lemma are divided into the following classes after applying the curation filters:

- **Valid:** The chain passes all filters and is considered a valid etymological chain.
- **Empty forms:** The word form is an empty string or a hyphen.
- **Repeated language:** The same language appears twice within a single ancestry chain.
- **Annotated forms:** The word form contains parentheses, indicating the presence of gloss commentary.
- **Morpheme entries:** The word begins with a hyphen, indicating an affix rather than a standalone lexical form.
- **Enumeration forms:** The form contains a comma, suggesting that multiple lemmas were enumerated in the same string.
- **Markup contamination:** The word contains markup characters such as `<`, `>`, `[`, `]`, `{`, or `}`.

Table 1 reports the outcome of the curation procedure. Lemmas with valid etymologies constitute approximately 78% of the original dataset, while the remaining 22% are filtered out by the curation rules. Among the filtered classes, the category “repeated language” (7%) is the one most likely to contain false negatives, since lexical items may undergo internal derivation, semantic shift, or other developments while remaining within the same language. By contrast, etymologies containing empty forms (13%) are certain to be true negatives, since an etymology cannot be valid while missing etymons.

3.4 RDF serialization

Given a list of curated etymology chains, the final stage of the pipeline consists of serializing them into RDF triples. In order to represent the data in a semantically meaningful and interoperable way, an ontological vocabulary must be used to model the relations between lexical entries and their ancestors. For this purpose, we adopt the OntoLex-lemon Etymology extension (*lemonEty*), a vocabulary specifically designed for representing etymological relations in RDF (Khan, 2018). The *lemonEty* model extends the OntoLex-lemon ontology with classes

Category	Count
Valid	9,684
Empty forms	1,639
Repeated language	861
Annotated forms	192
Morpheme-only forms	34
Enumeration patterns	9
Markup artifacts	2

Table 1: Distribution of valid and filtered etymologies by lemma after rule-based curation.

and properties that capture typical elements of etymological description, such as etymons and etymological links. Using this model allows the extracted chains to be published as linked data and facilitates interoperability with other linguistic resources.

Furthermore, `lemonEty` is already used in the LiLa Knowledge Base. Adopting the same model therefore ensures compatibility between our dataset and the existing LiLa infrastructure. According to said model, an etymology has the following elements:

- **Lexical entry:** the lexical item whose etymology is being described (e.g., *lupus*). This is represented as an `ontolex:LexicalEntry` and linked to its etymology.
- **Etymology node:** an instance of `lemonEty:Etymology` that acts as the central object representing the etymological description of the lexical entry.
- **Etymon:** the historical lexical forms from which the word derives (e.g., reconstructed forms such as **luk^wos*). These are represented as instances of `lemonEty:Etymon`, which is in turn a subclass of `ontolex:Form`.
- **Etymological links:** instances of `lemonEty:EtyLink` that connect successive stages of the etymological chain.
- **Source and target relations:** each etymological link specifies an `etySource` (the ancestor form) and an `etyTarget` (the derived form).
- **Canonical forms:** lexical entries are linked to their canonical form of citation through the property `ontolex:canonicalForm`. In the `OntoLex-lemon` model, a lexical entry is associated with an `ontolex:Form`

representing its canonical form (i.e., the lemma), while the written representation of that form is expressed through the property `ontolex:writtenRep`. In LiLa, lemmas are modeled as instances of the class `lila:Lemma`, which is defined as a subclass of `ontolex:Form`. Linking lexical entries to LiLa lemmas through `ontolex:canonicalForm` therefore constitutes the standard mechanism used to connect lexical resources to the LiLa knowledge base.

In our pipeline, this model is instantiated automatically from the curated etymology chains. Given such a chain, the RDF serialization procedure represents the word whose etymology is described as an `ontolex:LexicalEntry`, linked to a central `lemonEty:Etymology` node. All preceding historical stages are modeled as `lemonEty:Etymon` instances. The diachronic progression is encoded through a sequence of `lemonEty:EtyLink` nodes connecting each stage to the next by means of `lemonEty:etySource` and `lemonEty:etyTarget`. These links are attached to the etymology node with `lemonEty:hasEtyLink`, and the first one is marked with `lemonEty:startingLink`.

Before serializing each etymology chain, we perform a preliminary enrichment step in which each node is associated with external identifiers. First, a Wiktionary URL is assigned to every form by attempting to resolve the corresponding page in the English Wiktionary. The system generates candidate URLs based on the form and its language code, accounting for special cases such as reconstructed proto-language forms, and verifies their existence through HTTP requests. The first candidate returning a valid response is retained as the Wiktionary reference for that node. During RDF serialization, this URL is then attached to the corresponding lexical entry or etymon using the property `schema:url`.

For Latin forms, an additional linking step connects the lemma to the LiLa Knowledge Base. This is done by querying the LiLa Text Linker service endpoint², which returns a linking key identifying the corresponding LiLa lemma when a match is found (Passarotti et al., 2024). In cases where multiple candidate lemmas are returned (e.g., due to homography), the pipeline selects the first candidate. Candidates are returned sorted in ascending

²<https://lila-erc.eu/LiLaTextLinker/processText>

order by lemma ID (e.g., `lilaLemma:103739`), so this corresponds to selecting the candidate with the lowest lemma ID. The linking key is then converted into a persistent LiLa URI and used during RDF serialization to link the lexical entry to the corresponding LiLa resource through `ontolex:canonicalForm`.

Given the etymology chain of *frater* shown in Section 2.2, Listing 1 shows a simplified excerpt of its RDF representation. Prefixes such as `exlex`, `exety`, `exetym`, and `exlink` are used instead of full URIs for readability and stand for the corresponding namespaces in the dataset.

Listing 1: Simplified RDF triples for the etymology of *frater*.

```
exlex:frater-la a ontolex:LexicalEntry ;
  lemonEty:etymology exety:frater-la ;
  schema:inLanguage "la"^^xsd:language ;
  schema:url <https://en.wiktionary.org/wiki/frater> ;
  rdfs:label "frater" ;
  ontolex:canonicalForm <https://lila-erc.eu/data/id/lemma/103739> .

exety:frater-la a lemonEty:Etymology ;
  lemonEty:etymon exetym:frater-la-0 , exetym:frater-la-1 ;
  lemonEty:hasEtyLink exlink:frater-la-1 , exlink:frater-la-2 ;
  lemonEty:startingLink exlink:frater-la-1 ;
  rdfs:label "Etymology of: frater" .

exetym:frater-la-1 a lemonEty:Etymon ;
  schema:inLanguage "x-ine-pro"^^xsd:language ;
  rdfs:label "*brhtr" .

exetym:frater-la-0 a lemonEty:Etymon ;
  schema:inLanguage "x-itc-pro"^^xsd:language ;
  rdfs:label "*frtr" .

exlink:frater-la-1 a lemonEty:EtyLink ;
  lemonEty:etySource exetym:frater-la-1 ;
  lemonEty:etyTarget exetym:frater-la-0 ;
  rdfs:label "Etymology Link" .

exlink:frater-la-2 a lemonEty:EtyLink ;
  lemonEty:etySource exetym:frater-la-0 ;
  lemonEty:etyTarget exlex:frater-la ;
  rdfs:label "Etymology Link" .
```

Finally, RDF generation must account for the scale of the Wiktionary-derived data. The construction of triples itself is relatively inexpensive and, if the pipeline were purely CPU-bound, it would run substantially faster. In practice, however, the enrichment stage dominates runtime, since it depends on external network requests to resolve Wiktionary URLs and query the LiLa linking service. To keep the pipeline scalable, the graph is not stored in memory as a whole; instead, each enriched etymol-

ogy chain is immediately serialized to an `.nt` file. This streaming approach prevents excessive RAM usage and is therefore essential when processing large volumes of data. N-Triples is especially appropriate for this purpose because it supports incremental, line-based serialization. Finally, for human inspection, the resulting `.nt` file is converted into Turtle (`.ttl`), a more readable RDF format.

3.5 Data publication

The resulting dataset has been published as a Linked Data resource within the LiLa graph and is accessible online through the LiLa infrastructure at <https://lila-erc.eu/data/lexicalResources/englishWiktionaryLatinEtymologies/Lexicon>. Entitled *English Wiktionary Latin Etymologies*, it contains 9,684 lexical entries corresponding to Latin lemmas with curated etymologies extracted from the English edition of Wiktionary. Each entry is represented as an `ontolex:LexicalEntry` linked to its etymological representation through the `lemonEty` ontology. The lexicon metadata specifies the provenance of the data, indicating that the etymologies were retrieved from the raw Wiktextextract dump made available at <https://kaikki.org/dictionary/rawdata.html> and processed into RDF by our pipeline. The dataset is released under the same license as the underlying Wiktionary data, Creative Commons Attribution-ShareAlike (CC BY-SA), and is integrated into the LiLa ecosystem³, which allows it to interoperate with existing Latin linguistic resources already linked within the knowledge base.

3.6 Reproducibility

The corpus generation pipeline is fully reproducible. It begins with the raw Wiktextextract dump of the English Wiktionary available at <https://kaikki.org/dictionary/rawdata.html>. From this source, etymology templates are extracted and converted into structured etymology chains. These chains are then curated, enriched and, finally, serialized into RDF triples. The code implementing this pipeline can be retrieved from the GitHub repository associated with this project, <https://github.com/CIRCSE/englishWiktionaryLatinEtymologies>, in which the dataset in both `.nt` and `.ttl` can be found

³<https://lila-erc.eu/data-page/>

as well. This allows the corpus generation process to be reproduced from the original Wikitext data. That is, given a Wikitext dump, the pipeline consistently returns the same RDF output. The only exception is the entity linking step, where manual disambiguation may be required in cases of lexical ambiguity, as discussed next in the Limitations section.

4 Limitations

The dataset presented in this work is derived from the English version of Wiktionary and therefore inherits some of its properties. Multiple entries may exist for different grammatical forms of the same lexical item (e.g., nominative *animal* and its genitive *animalis*) or for distinct uses across parts of speech (e.g., *Februārius* as both a proper noun and an adjective). Wiktionary provides valuable etymological information, but some degree of noise is inherent to the source data.

A further limitation concerns the alignment with external resources, such as the Lemma Bank of LiLa. In some cases, an entity linking service yields several candidates, as ambiguity cannot be resolved through surface form matching alone. As discussed in the RDF Serialization section, the current pipeline always selects the lemma with the lowest value for the lemma ID, but accurate disambiguation requires manual inspection. In our data, 381 lemmas were identified as having multiple candidate matches. While this does not affect the structural validity of the etymological chains, it impacts the precision of the alignment with LiLa and sets direction for future work.

For example, for the Latin form *os*, the LiLa Text Linker yields the candidates 115327 (“mouth, face, opening”), 115330 (“bone”) and 68537 (no meaning available). *Os* has two etymology chains:

$[(h_3\acute{e}h_1os, ine-pro), (\acute{o}s, itc-pro), (os, la)]$
 $[(h_3\acute{e}sth_1, ine-pro), (os, la)]$

In Wiktionary, we can see that the etymology chain tracing back to Proto-Indo-European $*h_3\acute{e}h_1os*$ corresponds to the sense “mouth” and aligns with LiLa lemma 115327, whereas the chain derived from $*h_3\acute{e}sth_1*$ corresponds to “bone” and matches LiLa lemma 115330. This illustrates that selecting the correct lemma requires interpreting the meaning associated with each etymology and aligning it with the corresponding entry in the external resource.

Finally, the curation stage prioritizes precision over recall. While this reduces noise and increases the overall reliability of the dataset, it may also exclude valid etymological relations. As a result, there may be false negatives that have been excluded from the dataset, which should therefore not be considered an exhaustive collection of all Latin etymologies present in Wiktionary.

5 Conclusion

In this paper, we presented a curated resource of Latin etymological chains automatically extracted from the English edition of Wiktionary and represented as RDF triples using the OntoLex-lemon model and its lemonEty extension. The resulting dataset contains validated etymology chains for 9,684 lemmas and has been integrated into the LiLa Knowledge Base of Linguistic Resources for Latin, enabling interoperability with existing linked linguistic resources.

The pipeline used to generate the corpus is fully reproducible and can be adapted to extract etymological data for other languages available in Wiktionary. We hope that the resulting resource will support future work in Historical Linguistics, computational etymology, and linked lexical data, as well as facilitate the integration of etymological information into broader linguistic knowledge graphs.

References

- Pierpaolo Basile and 1 others. 2022. A new time-sensitive model of linguistic knowledge for digital humanities. In *Proceedings of the Workshop on Language Technology for Digital Humanities*.
- Tim Berners-Lee. 2006. [Linked data](#).
- Anas Fahad Khan. 2018. Towards the representation of etymological data on the semantic web. *Information*, 9(11):288.
- Francesco Mambrini and Marco Passarotti. 2020. Representing etymology in the lila knowledge base of linguistic resources for latin. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography*, pages 20–28, Marseille, France. European Language Resources Association.
- John P. McCrae and 1 others. 2017. The ontolx-lemon model: Development and applications. *Electronic Lexicography in the 21st Century*.
- Roberto Navigli and Simone Paolo Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic

network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Ester Pantaleo, Vito Walter Anelli, Tommaso Di Noia, and Gilles Sérasset. 2017. Etytree: A graphical and interactive etymology dictionary based on wiktionary. In *Proceedings of the WWW Companion*.

Marco Passarotti, Francesco Mambrini, and 1 others. 2024. The lila text linker: Linking latin texts to the lila knowledge base. In *Proceedings of the Workshop on Linked Data in Linguistics (LDL 2024)*.

Gilles Sérasset. 2014. Dbnary: Wiktionary as a lemon-based multilingual lexical resource in rdf. *Semantic Web*.

Tatu Ylonen. 2022. Wiktextextract: Wiktionary as machine-readable structured data. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1317–1325, Marseille, France. European Language Resources Association.

How Good is AI on Swiss Voting Booklets? A Multilingual OCR and Alignment Benchmark

Elina Stüssi and Jannis Vamvas
Department of Computational Linguistics
University of Zurich
elina.stuessi@uzh.ch, vamvas@cl.uzh.ch

Abstract

Swiss federal voting booklets are an interesting resource for natural language processing due to their high editing standards and coverage of the four national languages of Switzerland (German, French, Italian, and Romansh Grischun). In this paper, we present VotingBooklets, an automatically extracted and aligned dataset, as well as VotingBooklets-Diamond, a subset that was manually corrected and verified by multiple annotators. We use the latter to benchmark a range of open and closed AI systems on two interdependent tasks: optical character recognition (OCR) and cross-lingual text alignment. Gemini 2.5 Flash Lite achieves the best OCR performance across all conditions, while a hybrid alignment approach using Sentence-SwissBERT for initial embedding-based alignment and Gemini for targeted post-hoc correction of low-confidence pairs yields the most accurate results. Applying these systems to the full collection of Swiss federal voting booklets, we release a large-scale four-language parallel corpus as a resource for low-resource NLP, multilingual representation learning, and the computational study of Swiss political discourse.

1 Introduction

Multilingual archives hold significant potential for computational research, but realizing this potential requires converting physical documents into machine-readable digital text. For historical administrative documents, this involves two core tasks: optical character recognition (OCR) to extract text from scanned pages, and cross-lingual alignment to link parallel content across languages.

In this paper, we investigate both tasks jointly using Swiss federal voting booklets (*Abstimmungsbüchlein*) as our document collection. These booklets are distributed to all eligible voters prior to each federal referendum and present identical political content in the four Swiss national languages. Their parallel structure makes them a uniquely suitable

German	Bundesrat und Parlament empfehlen den Stimmberechtigten, am 11. März 2007 wie folgt zu stimmen:
French	Le Conseil fédéral et le Parlement vous recommandent de voter, le 11 mars 2007:
Italian	Consiglio federale e Parlamento vi raccomandano di votare come segue l'11 marzo 2007:
Romansh	Il cussegl federal ed il parlament recumondan a las votantas ed als votants da votar ils 11 da mars 2007 sco suonda:

Figure 1: Example of a parallel sentence from the VotingBooklets-Diamond dataset, illustrating all four Swiss national languages (German, French, Italian, Romansh) from the federal vote of 11 March 2007.

resource for evaluating multilingual document processing pipelines. We introduce VotingBooklets-Diamond¹, a carefully curated, manually corrected test set covering three voting dates (1977, 1985, and 2007) and use it to benchmark a range of open and proprietary AI systems on both OCR and cross-lingual alignment.

Our VotingBooklets-Diamond test set presents two central challenges. First, it includes languages with markedly different levels of resource availability. German, French, and Italian are well supported in modern NLP systems, whereas Romansh Grischun, spoken by roughly 60,000 people, remains a low-resource language with limited training data and few dedicated tools. Second, the two tasks we consider are inherently interdependent. Errors introduced during OCR propagate to downstream alignment and degrade its quality. Figure 1

¹The dataset is available at <https://huggingface.co/datasets/eljuanina/VotingBooklets-Diamond-v1>

Vote	Languages	Pages	PDF Format	Notes
1977	DE, FR, IT	8	Scanned	No Romansh; two content pages per scan page
1985	DE, FR, IT, RM	4	Scanned	Romansh added; two content pages per scan page
2007	DE, FR, IT, RM	16	Born-digital	High-quality machine-readable PDFs

Table 1: Overview of the Swiss voting booklets included in VotingBooklets-Diamond. The *Pages* column refers to PDF scan pages; the 1977 and 1985 booklets contain 15 and 8 content pages respectively, with two content pages printed per scan page. Language abbreviations: DE=German, FR=French, IT=Italian, RM=Romansh.

Year	German (de) Tokens	French (fr) Tokens	Italian (it) Tokens	Romansh (rm) Tokens
1977	3,994	4,846	4,366	–
1985	1,618	2,193	1,821	2,040
2007	2,000	2,595	2,413	2,766
Total	7,612	9,634	8,600	4,806

Table 2: Token count per language and vote in the **VotingBooklets-Diamond** dataset. Romansh (rm) is only available for 1985 and 2007.

shows an example sentence from a Swiss federal voting booklet in all four national languages, illustrating the highly parallel structure that underlies our dataset.

Beyond benchmarking, we apply our findings to the full collection of Swiss federal voting booklets. Gemini 2.5 Flash Lite achieves the best OCR performance on VotingBooklets-Diamond and is therefore used to process the full collection of Swiss federal voting booklets. For alignment, we employ a two-stage approach that combines embedding-based alignment with Sentence-SwissBERT (Grosjean and Vamvas, 2024) and targeted post-hoc correction using Gemini 2.5 Flash Lite to refine low-confidence matches. The resulting dataset, VotingBooklets², forms a large-scale four-language parallel corpus that serves as a new resource for research in low-resource NLP, multilingual representation learning, and the computational study of political discourse in multilingual societies.

2 Corpus Design

2.1 Document Collection

Swiss federal voting booklets are official documents issued by the Swiss Federal Chancellery to all eligible voters before each federal referendum.³ Under Switzerland’s system of direct democracy, referendums are held several times per year, and

²The dataset is available at <https://huggingface.co/datasets/eljuanina/VotingBooklets-v1>

³Available from the Swiss Federal Chancellery: <https://www.bk.admin.ch/bk/de/home/dokumentation/abstimmungsbuechlein.html>.

each booklet presents the same content in all four national languages: German, French, Italian, and Romansh Grischun. This makes them a naturally occurring resource of highly parallel multilingual text produced under real-world institutional conditions.

Our VotingBooklets-Diamond dataset comprises eleven voting booklets drawn from three federal votes: 12 June 1977, 1 December 1985, and 11 March 2007 (Table 1). These dates were selected to span three decades while maximizing linguistic and technical variation. The earliest booklet available online, from 1977, predates the introduction of Romansh editions and is therefore limited to German, French, and Italian. The 1985 vote marks the first inclusion of Romansh Grischun, while the 2007 materials are distributed as born-digital PDFs rather than scanned documents. Taken together, these three time points capture substantial variation in document quality, scan resolution, typography, and layout.

Each booklet follows a structured format comprising vote overviews, official proposals, arguments for and against, parliamentary recommendations, legal texts, and voting instructions. At the same time, the documents exhibit considerable layout complexity, with multi-column text, tables, and footnotes that pose challenges for OCR and layout analysis systems. Despite minor translation-level phrasing differences, the content is substantively equivalent across all language versions, enabling precise cross-lingual alignment.

Language	German (de)		French (fr)		Italian (it)		Romansh (rm)	
	Tokens	Pages	Tokens	Pages	Tokens	Pages	Tokens	Pages
Collection	1,119,494	5,037	1,420,348	5,089	1,065,848	5,108	859,802	4,058

Table 3: Token and page counts per language in the **VotingBooklets** dataset (complete collection of Swiss federal voting booklets).

2.2 Gold Standard Annotation

We created a gold-standard transcription for the eleven booklets through manual correction. The scanned PDFs from 1977 and 1985 carry an existing OCR layer, but its quality is poor and served only as a rough starting point. For the 2007 born-digital PDFs, we extracted text directly from the PDF. In both cases, every passage was fully reviewed and corrected by hand to ensure fidelity to the original document. Transcription follows the physical page sequence of the PDF files rather than inferred reading order, so that OCR and layout analysis tools can be evaluated fairly regardless of whether they correctly reconstruct logical flow from non-linear layouts. For dual-page scans, left-hand pages were transcribed in full before right-hand pages, preserving physical layout structure. Non-textual elements, such as photographs, logos, and purely illustrative graphics, were not transcribed or described in the final dataset. Each transcription was independently checked by two additional annotators, who compared the transcriptions directly with the original PDF documents to ensure accuracy. Any discrepancies found were due to minor typographical errors or omitted words in the initial transcription. All such issues were carefully reviewed and corrected through discussion, achieving full agreement among the annotators.

Cross-lingual alignment was performed at the paragraph level, using the physical paragraph boundaries present in the PDF as the primary segmentation unit. Where structural divergence across languages required it, adjacent paragraphs were merged to ensure semantic equivalence across aligned pairs. Alignment decisions were verified by two annotators, both native German speakers with basic knowledge of French and Italian. They could infer the meaning of Romansh segments based on their knowledge of the other languages, and in cases of uncertainty, they consulted the original booklets. There were no disagreements between the annotators.

2.3 Corpus Statistics

Table 2 shows token counts for VotingBooklets-Diamond across all three voting dates and four languages. Each booklet is represented as a separate file, with German (de), French (fr), Italian (it), and Romansh (rm) where available.

2.4 Availability and License

VotingBooklets-Diamond and VotingBooklets are available on Hugging Face. All code and scripts used to preprocess, perform OCR, and align the documents are provided via our [GitHub repository](#), enabling full reproducibility. The dataset is released under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, redistribution, and modification, provided appropriate credit is given.

3 Baseline Systems

To demonstrate the utility of the dataset and identify the best-performing systems for corpus creation, we benchmarked a range of open and closed AI models on both OCR and cross-lingual alignment.

3.1 OCR Evaluation

We evaluated three OCR approaches against the gold-standard VotingBooklets-Diamond transcriptions using Word Error Rate (WER) and Character Error Rate (CER): direct OCR with gemini-2.5-flash-lite (Gemini2.5), Pytesseract alone (v0.3.13), a Python wrapper for Google’s Tesseract-OCR engine (Smith, 2007), and a post-correction pipeline in which Pytesseract output was corrected by gemini-2.5-flash-lite (Py+Gem2.5). To ensure that evaluation scores reflect only transcription quality and not segmentation mismatches, OCR output was manually aligned with the gold standard before computing WER and CER. Full results are shown in Table 4.

Gemini2.5 dominates OCR Direct OCR with Gemini2.5 achieves the lowest WER and CER in almost every condition, across all three decades and all four languages. The margin over Pytesseract is substantial: for the 2007 booklets, Pytesseract

act reaches WER values above 0.20 for all languages, while Gemini2.5 stays below 0.08. Even for the older scanned documents from 1977 and 1985, Gemini2.5 performs consistently well, suggesting strong robustness to varying scan quality.

Post-correction does not help Despite evidence in prior work that LLM-based post-correction can improve OCR output (Greif et al., 2025), especially in low-resource settings (Hebbalalu, 2026; Kanerva et al., 2025), the gains are limited in our case. While post-correction does improve over raw Pytesseract output, Gemini2.5 still outperforms the post-correction pipeline in nearly every condition. This suggests that for this document type, end-to-end vision-language models are a more effective approach than pipeline-based post-correction.

Romansh is not a weak point Somewhat surprisingly, Romansh does not consistently lag behind the other languages. For 1985 and 2007, Gemini2.5 achieves competitive CER for Romansh (0.0024 and 0.0095 respectively), comparable to German, French, and Italian. This may reflect the lexical similarity of Romansh to other Romance languages rather than any specific model training coverage.

3.2 Alignment Evaluation

We evaluate four alignment approaches using F1, precision, recall, and character error rate against VotingBooklets-Diamond gold-standard alignments. The evaluated systems include embedding-based alignment with paraphrase-multilingual-MiniLM-L12-v2 (Reimers and Gurevych, 2019) (MiniLM), Sentence-SwissBERT (Grosjean and Vamvas, 2024) (SSB), a LLM-based alignment approach using gemini-2.5-flash-lite (Gemini), and hybrid methods that combine Sentence-SwissBERT with targeted post-hoc correction using either gemini-3-pro-preview (SSB+Gemini3) or gemini-2.5-flash-lite (SSB+Gemini2.5).

The Gemini-based system performs alignment by directly matching German anchor segments to target-language fragments using constrained prompting, allowing segments to be merged while enforcing one-to-one coverage of the anchor sequence. In contrast to embedding-based approaches, this method operates purely at the text level without explicit similarity scoring. Implementation details are provided in Appendix A.3.

To isolate alignment quality from OCR noise, all systems are evaluated on gold-standard tran-

scriptions rather than raw OCR output. German serves as the anchor language, with paragraph-level segmentation from the gold standard used as the reference structure. French, Italian, and Romansh are provided as segmented inputs, which must be aligned to the German paragraphs. Full results are reported in Table 5.

Post-hoc correction improves alignment quality

The combined Sentence-SwissBERT and Gemini post-hoc correction approaches achieve the best F1 scores in the majority of conditions and yield the lowest CER almost universally. The improvements are particularly pronounced for the low-resource language Romansh: in 2007, SSB+Gemini3 reaches an F1 of 0.986 and a CER of 0.003, compared to 0.903 and 0.072 for MiniLM, highlighting the benefit of combining embedding-based alignment with targeted LLM-based correction.

Comparing the two post-hoc variants, we find that gemini-2.5-flash-lite performs almost on par with gemini-3-pro-preview, with only small differences in F1 and CER across all languages and years. Given this near-identical performance, we adopt the SSB+Gemini2.5 configuration in our final pipeline, combining Sentence-SwissBERT alignment with post-hoc correction using gemini-2.5-flash-lite. This choice provides a substantially more cost-efficient solution without compromising alignment quality.

4 The Full Parallel Corpus

We construct the full VotingBooklets corpus by applying the best-performing systems identified in Sections 3.1 and 3.2, namely Gemini2.5 for OCR and SSB+Gemini2.5 for alignment, to the complete collection of Swiss federal voting booklets provided by the Federal Chancellery. The dataset comprises 144 booklets in German, 144 in French, 144 in Italian, and 97 in Romansh (RM) (91 obtained from the official webpage, with an additional 6 RM booklets acquired from the State Chancellery of Grisons). Other missing Romansh files could not be found in archives so far. The collection covers votes from June 1977 to March 2026 for German, French, and Italian, and from December 1985 to March 2026 for Romansh.

In total, VotingBooklets contains 19,292 pages and 4,465,492 tokens across all languages. Table 3 provides a detailed overview of the number of tokens and pages for each language in the collection.

The corpus is distributed as JSON Lines (JSONL) files, with one file per voting booklet containing aligned segments across all available languages. Each line in a file represents a paragraph-level alignment, with German as the anchor, alongside the corresponding segments in French, Italian, and Romansh Grischun where available. Languages that are missing for a segment are represented as empty strings.

Quality was assessed through manual evaluation on a held-out sample of aligned segments drawn from booklets not included in VotingBooklets-Diamond. Overall, quality is somewhat lower than on VotingBooklets-Diamond, which is expected: unlike the diamond set, the full pipeline operates on uncorrected OCR output and must handle the greater document diversity of the full corpus. Errors in alignment, OCR, or paragraph splitting occur more frequently, though they remain largely isolated to individual rows and do not propagate across entire documents.

5 Use Cases and Future Work

5.1 Use Cases

Benchmarking OCR and alignment systems

VotingBooklets-Diamond enables fair comparison of OCR approaches on historical Swiss administrative documents across varying scan quality, typography, and document structure. The verified paragraph-level alignments allow evaluation of cross-lingual alignment methods independently of OCR quality, including both open-source embedding models and commercial APIs.

Low-resource NLP for Romansh VotingBooklets is one of the few large parallel resources available for Romansh Grischun, and can support machine translation, cross-lingual transfer, and language modeling for this language.

Multilingual representation learning The four-language parallel structure of VotingBooklets makes the corpus well-suited for training and evaluating multilingual embeddings, particularly for Swiss national languages in institutional and political domains.

Computational analysis of political discourse

Covering several decades of federal referendum material, VotingBooklets enables longitudinal studies of political language, argumentation, and framing across languages and time periods.

5.2 Future Work

Our results reveal two interesting dependencies: between OCR and alignment, and between alignments across language pairs. These dependencies suggest promising directions for future work. An agentic pipeline that iterates between OCR post-correction and alignment, passing information in both directions, could leverage these relationships to improve performance on both tasks simultaneously. This methodology could also be extended to other Swiss multilingual document collections, such as parliamentary proceedings or cantonal administrative documents, to create richer resources for Swiss NLP.

Beyond the pipeline itself, the scope of the corpus remains an open direction for expansion. The current version omits non-textual elements such as images. Future versions could incorporate these alongside structural layout information, enabling layout-aware analysis and opening the corpus to a broader range of research applications in document understanding and multimodal NLP.

6 Related Work

OCR Technology Traditional OCR systems such as Tesseract (Smith, 2007) rely on pattern recognition and handcrafted linguistic heuristics. These systems work well on clean and modern documents but are less robust when faced with complex layouts, degraded scans, or low-resource languages (Ignat et al., 2022; Greif et al., 2025).

Recent deep learning approaches improve recognition accuracy considerably. Transformer-based models such as TrOCR (Li et al., 2023) jointly model visual and textual features and achieve strong performance on both printed and handwritten text. Large multimodal language models push this further. Gemini 2.0 Flash reaches a character error rate of 1.27% on historical German documents, which can be reduced to 0.84% with multimodal post-correction (Greif et al., 2025).

Beyond direct recognition, recent work explores agentic and LLM-based pipelines that refine OCR outputs iteratively. OCR-Agents use structured reasoning loops to improve recognition quality in difficult settings (Wen et al., 2026), and LLM-based post-correction has proven effective for historical and low-resource languages, where contextual information helps resolve uncertain character sequences (Greif et al., 2025; Hebbalalu, 2026; Kanerva et al., 2025).

The Swiss Multilingual Landscape There are relatively few resources of parallel multilingual text that cover all the Swiss national languages. Notable resources are *Swiss Law Translations* (Niklaus et al., 2025), which aligns federal laws on the level of documents, articles and paragraphs, and the trilingual *Allegra* corpus (Scherrer and Cartoni, 2012), which is composed of press releases by the canton of Grisons. Other corpora that include all Swiss national languages, but which are not necessarily parallel, pertain to domains such as web text (Krasselt et al., 2020; Penedo et al., 2025), news (Graën et al., 2023), social text (Dürscheid and Stark, 2011; Ueberwasser and Stark, 2017), and alpine yearbooks (Göhring and Volk, 2011). In addition, recent work has contributed parallel test sets for machine translation evaluation that cover all four Swiss national languages (Deutsch et al., 2025; Vamvas et al., 2025; Andrews et al., 2025).

Alignment Methods Early alignment methods rely on sentence length models (Gale and Church, 1993). Bleualign (Sennrich and Volk, 2010) uses machine translation and BLEU scores for alignment and achieves strong results on noisy OCR corpora. Later work removes the dependency on pre-trained translation models using iterative bootstrapping (Sennrich and Volk, 2011). Vecalign (Thompson and Koehn, 2019) replaces translation signals with sentence embeddings from LASER (Artetxe and Schwenk, 2019) and uses approximate dynamic programming for alignment. CroCoAlign (Molfese et al., 2024) extends this with context-aware sentence embeddings computed at document level using a transformer encoder and improves alignment quality across language pairs.

Sentence representations are a crucial factor in alignment quality. For Swiss languages, SentenceSwissBERT is introduced by Grosjean and Vamvas (2024). It is based on SwissBERT (Vamvas et al., 2023) and fine-tuned using contrastive learning on Swiss news data in German, French, Italian, and Romansh. It improves performance over multilingual Sentence-BERT (Reimers and Gurevych, 2019), especially for Romansh. This supports the use of domain-specific sentence embeddings in cross-lingual alignment.

7 Conclusion

We presented VotingBooklets-Diamond, a carefully curated gold-standard test set of Swiss federal voting booklets spanning three decades and all four

Swiss national languages, and used it to benchmark open and closed AI systems on OCR and cross-lingual alignment. Gemini 2.5 Flash Lite emerged as the strongest OCR system across all conditions, Sentence-SwissBERT (SSB) alignment with Gemini-based post-hoc correction achieved the best alignment performance, particularly for Romansh Grischun.

Applying these systems to the full collection of Swiss federal voting booklets, we release Voting-Booklets, a large-scale four-language parallel corpus as a new resource for the Swiss NLP community. We hope both the benchmark and the corpus will support future work on multilingual document processing, low-resource NLP, and the computational study of Swiss political discourse.

Acknowledgments

We thank Sophia Conrad and Giuanna Caviezel for their helpful advice. We also thank the State Chancellery of Grisons, the Federal Chancellery (BK), the Federal Department of Home Affairs, and the Swiss National Library (NB) for their assistance in searching for and locating missing voting booklets.

References

- Pierre Andrews, Mikel Artetxe, Mariano Coria Meglioli, Marta R. Costa-jussà, Joe Chuang, David Dale, Mark Duppenthaler, Nathaniel Paul Ekberg, Cynthia Gao, Daniel Edward Licht, Jean Maillard, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Eduardo Sánchez, Ioannis Tsiamas, Arina Turkatenko, Albert Ventayol-Boada, and Shireen Yates. 2025. [BOUQuET : dataset, Benchmark and Open initiative for Universal Quality Evaluation in Translation](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27515–27535, Suzhou, China. Association for Computational Linguistics.
- Mikel Artetxe and Holger Schwenk. 2019. [Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond](#). *Transactions of the Association for Computational Linguistics*, 7:597–610.
- Daniel Deutsch, Eleftheria Briakou, Isaac Rayburn Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. [WMT24++: Expanding the Language Coverage of WMT24 to 55 Languages & Dialects](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 12257–12284, Vienna, Austria. Association for Computational Linguistics.

- Christa Dürscheid and Elisabeth Stark. 2011. sms4science: An International Corpus-Based Texting Project and the Specific Challenges for Multilingual Switzerland. In Crispin Thurlow and Kristine Mroczek, editors, *Digital Discourse. Language in the New Media*, Oxford Studies in Sociolinguistics, page 299–320. Oxford University Press.
- William A. Gale and Kenneth W. Church. 1993. A Program for Aligning Sentences in Bilingual Corpora. *Computational Linguistics*, 19(1):75–102.
- Anne Göhring and Martin Volk. 2011. Le corpus Text+Berg Une ressource parallèle alpin français-allemand (The Text+Berg Corpus An Alpine French-German Parallel Resource). In *Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles. Articles courts*, pages 97–102, Montpellier, France. ATALA.
- Johannes Graën, Igor Mustac, Nikolina Rajovic, Jonathan Schaber, Gerold Schneider, and Noah Bubenhofer. 2023. [Swissdox@LiRI. A large database of media articles made accessible to researchers](#). In Krister Linden, Jyrki Niemi, and Thalassia Editors Kontino, editors, *CLARIN Annual Conference Proceedings*, CLARIN Annual Conference Proceedings, page 111–115. CLARIN ERIC.
- Gavin Greif, Niclas Griesshaber, and Robin Greif. 2025. [Multimodal LLMs for OCR, OCR Post-Correction, and Named Entity Recognition in Historical Documents](#). *Preprint*, arXiv:2504.00414.
- Juri Grosjean and Jannis Vamvas. 2024. [Fine-tuning the SwissBERT Encoder Model for Embedding Sentences and Documents](#). In *Proceedings of the 9th edition of the Swiss Text Analytics Conference*, pages 41–49, Chur, Switzerland. Association for Computational Linguistics.
- Vishwambhara Hebbalalu. 2026. [Dual-Stage OCR Correction for Classical Languages using LLMs: A Comparative Evaluation](#). *Preprint*, ResearchGate.
- Oana Ignat, Jean Maillard, Vishrav Chaudhary, and Francisco Guzmán. 2022. [OCR Improves Machine Translation for Low-Resource Languages](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1164–1174, Dublin, Ireland. Association for Computational Linguistics.
- Jenna Kanerva, Cassandra Ledins, Siiri Käpyaho, and Filip Ginter. 2025. [OCR Error Post-Correction with LLMs in Historical Documents: No Free Lunches](#). In *Proceedings of the Third Workshop on Resources and Representations for Under-Resourced Languages and Domains (RESOURCEFUL-2025)*, pages 38–47, Tallinn, Estonia. University of Tartu Library, Estonia.
- Julia Krasselt, Philipp Dressen, Matthias Fluor, Cerstin Mahlow, Klaus Rothenhäusler, and Maren Runte. 2020. [Swiss-AL: A Multilingual Swiss Web Corpus for Applied Linguistics](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4145–4151, Marseille, France. European Language Resources Association.
- Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2023. [Trocr: transformer-based optical character recognition with pre-trained models](#). In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence, AAAI’23/IAAI’23/EAAI’23*. AAAI Press.
- Francesco Maria Molfese, Andrei Stefan Bejgu, Simone Tedeschi, Simone Conia, and Roberto Navigli. 2024. [CroCoAlign: A Cross-Lingual, Context-Aware and Fully-Neural Sentence Alignment System for Long Texts](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2209–2220, St. Julian’s, Malta. Association for Computational Linguistics.
- Joel Niklaus, Jakob Merane, Luka Nenadic, Sina Ahmadi, Yingqiang Gao, Cyrill A. H. Chevalley, Claude Humbel, Christophe Gösen, Lorenzo Tanzi, Thomas Lüthi, Stefan Palombo, Spencer Poff, Boling Yang, Nan Wu, Matthew Guillod, Robin Mamié, Daniel Brunner, Julio Pereyra, and Niko Grupen. 2025. [SwiLTra-Bench: The Swiss Legal Translation Benchmark](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14894–14916, Vienna, Austria. Association for Computational Linguistics.
- Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. 2025. [FineWeb2: One Pipeline to Scale Them All — Adapting Pre-Training Data Processing to Every Language](#). In *Second Conference on Language Modeling*.
- Nils Reimers and Iryna Gurevych. 2019. [SentenceBERT: Sentence Embeddings using Siamese BERT-Networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Yves Scherrer and Bruno Cartoni. 2012. [The Trilingual ALLEGRA Corpus: Presentation and Possible Use for Lexicon Induction](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2890–2896, Istanbul, Turkey. European Language Resources Association (ELRA).
- Rico Sennrich and Martin Volk. 2010. [MT-based Sentence Alignment for OCR-generated Parallel Texts](#).

In *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas: Research Papers*, Denver, Colorado, USA. Association for Machine Translation in the Americas.

Rico Sennrich and Martin Volk. 2011. [Iterative, MT-based Sentence Alignment of Parallel Texts](#). In *Proceedings of the 18th Nordic Conference of Computational Linguistics (NODALIDA 2011)*, pages 175–182, Riga, Latvia. Northern European Association for Language Technology (NEALT).

Raymond W. Smith. 2007. [An Overview of the Tesseract OCR Engine](#). *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, 2:629–633.

Brian Thompson and Philipp Koehn. 2019. [Vecalign: Improved Sentence Alignment in Linear Time and Space](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Simone Ueberwasser and Elisabeth Stark. 2017. [What’s up, Switzerland? A corpus-based research project in a multilingual country](#). *Linguistik Online*, 84(5):online.

Jannis Vamvas, Johannes Graß, and Rico Sennrich. 2023. [SwissBERT: The Multilingual Language Model for Switzerland](#). In *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 54–69, Neuchâtel, Switzerland. Association for Computational Linguistics.

Jannis Vamvas, Ignacio Pérez Prat, Not Soliva, Sandra Baltermia-Guetg, Andrina Beeli, Simona Beeli, Madlaina Capeder, Laura Decurtins, Gian Peder Gregori, Flavia Hobi, Gabriela Holderegger, Arina Lazzarini, Viviana Lazzarini, Walter Rosselli, Bettina Vital, Anna Rutkiewicz, and Rico Sennrich. 2025. [Expanding the WMT24++ Benchmark with Rumantsch Grischun, Sursilvan, Sutsilvan, Surmiran, Puter, and Vallader](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 1028–1047, Suzhou, China. Association for Computational Linguistics.

Shimin Wen, Zeyu Zhang, Xingdou Bian, Hongjie Zhu, Lulu He, Layi Shama, Daji Ergu, and Ying Cai. 2026. [OCR-Agent: Agentic OCR with Capability and Memory Reflection](#). *Preprint*, arXiv:2602.21053.

A Corpus Creation and Technical Details

A.1 Data Collection

The raw Swiss federal voting booklets were downloaded from the official Swiss Federal Chancellery website.⁴ A Python scraper was used to traverse

⁴<https://www.bk.admin.ch/bk/de/home/dokumentation/abstimmungsbaechlein.html>

language-specific navigation menus and extract all available PDF files. Filenames were sanitized to remove illegal characters and spaces were replaced with underscores.

A.2 OCR Processing

All scanned PDFs were converted to images using pdf2image, and each page was processed individually. OCR extraction was performed using Gemini 2.5 Flash Lite via a LangChain ChatOpenAI interface.

For native-born PDFs, text was also extracted using Gemini; the quality of these outputs was sufficiently high that little post-processing was required. In both cases, extraction followed the same segmentation and formatting rules to ensure consistency across booklets for alignment.

Critical extraction rules included:

- Rejoining hyphenated words split across lines.
- Joining non-hyphenated line breaks within sentences with a single space.
- Producing one line per sentence or list item.
- Omitting repeated headers, footers, page numbers, and other boilerplate.

The output of each OCR pass was stored as a text file, named according to the PDF source.

Scope and Constraints The current version of the corpus focuses exclusively on textual content. Visual elements, such as campaign photography and statistical charts, were systematically ignored during the extraction process. We did not perform any image transcription or alternative text generation; therefore, the dataset does not contain metadata or transcriptions for non-textual components.

A.3 Cross-Lingual Alignment

We implemented a two-stage alignment pipeline to align parallel text segments across languages.

Stage 1: Embedding-based alignment with Sentence-SwissBERT Each segment in the German anchor text and the target language is encoded into a vector representation using Sentence-SwissBERT (Grosjean and Vamvas, 2024) with the appropriate language adapter (de_CH, fr_CH, it_CH, rm_CH).

Alignment is performed using dynamic programming (DP) over the two segment sequences. The

Year	Lang	Method	WER	CER	Ins	Del	Sub
1977	DE	Gemini2.5	0.0462	0.0262	40	63	76
		Pytesseract	0.1074	0.0217	185	15	216
		Py+Gem2.5	0.0860	0.0136	160	3	170
1977	FR	Gemini2.5	0.0454	0.0142	45	67	110
		Pytesseract	0.1332	0.0233	167	49	436
		Py+Gem2.5	0.1287	0.0199	158	48	424
1977	IT	Gemini2.5	0.0321	0.0143	27	43	71
		Pytesseract	0.1058	0.0224	97	24	344
		Py+Gem2.5	0.0930	0.0184	59	17	333
1985	DE	Gemini2.5	0.0122	0.0015	8	1	10
		Pytesseract	0.1312	0.0584	69	75	61
		Py+Gem2.5	0.0621	0.0462	12	73	12
1985	FR	Gemini2.5	0.0181	0.0017	1	18	20
		Pytesseract	0.1107	0.0534	65	101	72
		Py+Gem2.5	0.0689	0.0456	15	107	26
1985	IT	Gemini2.5	0.0163	0.0052	3	10	16
		Pytesseract	0.1536	0.1021	103	115	55
		Py+Gem2.5	0.1294	0.1075	40	168	22
1985	RM	Gemini2.5	0.0171	0.0024	3	9	22
		Pytesseract	0.1134	0.0661	58	96	71
		Py+Gem2.5	0.0977	0.0668	37	102	55
2007	DE	Gemini2.5	0.0348	0.0192	14	25	27
		Pytesseract	0.2294	0.1518	135	173	127
		Py+Gem2.5	0.1187	0.1004	17	167	41
2007	FR	Gemini2.5	0.0720	0.0305	31	40	109
		Pytesseract	0.2277	0.1476	149	234	186
		Py+Gem2.5	0.1509	0.1033	27	223	127
2007	IT	Gemini2.5	0.0391	0.0081	16	7	68
		Pytesseract	0.2094	0.1434	118	228	141
		Py+Gem2.5	0.1410	0.1058	29	203	96
2007	RM	Gemini2.5	0.0568	0.0095	15	12	125
		Pytesseract	0.2202	0.1489	132	261	196
		Py+Gem2.5	0.1787	0.1139	37	252	189

Table 4: Evaluation of three OCR methods on VotingBooklets-Diamond, reporting Word Error Rate (WER), Character Error Rate (CER), and raw error counts (insertions, deletions, and substitutions) across all voting booklets and languages. Gemini2.5 = direct OCR using Gemini-2.5-Flash-Lite. Pytesseract = baseline OCR without post-correction. Py+Gem2.5 = Pytesseract OCR with Gemini-2.5-Flash-Lite post-correction. Languages: DE=German, FR=French, IT=Italian, RM=Romansh. Bold values indicate the best result per year, language, and metric. Lower WER/CER values indicate better performance.

Year	Lang	Method	F1	Prec	Rec	CER	
1977	FR	MiniLM	0.9187	0.9187	0.9187	0.0891	
		SSB	0.9210	0.9234	0.9187	0.0763	
		Gemini	0.9688	0.9712	0.9665	0.0147	
		SSB+Gemini2.5	0.9210	0.9234	0.9187	0.0760	
		SSB+Gemini3	0.9569	0.9569	0.9569	0.0192	
IT	IT	MiniLM	0.8995	0.8995	0.8995	0.1681	
		SSB	0.9187	0.9187	0.9187	0.0677	
		Gemini	0.9474	0.9474	0.9474	0.0337	
		SSB+Gemini2.5	0.9713	0.9713	0.9713	0.0171	
		SSB+Gemini3	0.9856	0.9856	0.9856	0.0151	
1985	FR	MiniLM	1.0000	1.0000	1.0000	0.0008	
		SSB	0.9533	0.9533	0.9533	0.0211	
		Gemini	0.9813	0.9813	0.9813	0.0105	
		SSB+Gemini2.5	0.9720	0.9720	0.9720	0.0158	
		SSB+Gemini3	0.9626	0.9626	0.9626	0.0191	
	IT	IT	MiniLM	1.0000	1.0000	1.0000	0.0000
			SSB	1.0000	1.0000	1.0000	0.0000
			Gemini	0.9671	0.9717	0.9626	0.0133
			SSB+Gemini2.5	1.0000	1.0000	1.0000	0.0000
			SSB+Gemini3	0.9907	0.9907	0.9907	0.0064
RM	RM	MiniLM	0.9626	0.9626	0.9626	0.0911	
		SSB	0.9813	0.9813	0.9813	0.0795	
		Gemini	0.9813	0.9813	0.9813	0.0130	
		SSB+Gemini2.5	1.0000	1.0000	1.0000	0.0000	
		SSB+Gemini3	1.0000	1.0000	1.0000	0.0001	
2007	FR	MiniLM	0.9694	0.9667	0.9721	0.0257	
		SSB	0.9805	0.9778	0.9832	0.0102	
		Gemini	0.9776	0.9775	0.9777	0.0086	
		SSB+Gemini2.5	0.9916	0.9889	0.9944	0.0004	
		SSB+Gemini3	0.9916	0.9889	0.9944	0.0003	
	IT	IT	MiniLM	0.9749	0.9722	0.9777	0.0103
			SSB	0.9861	0.9833	0.9888	0.0037
			Gemini	0.9636	0.9663	0.9609	0.0122
			SSB+Gemini2.5	0.9805	0.9778	0.9832	0.0075
			SSB+Gemini3	0.9916	0.9889	0.9944	0.0004
	RM	RM	MiniLM	0.9025	0.9000	0.9050	0.0716
			SSB	0.9526	0.9500	0.9553	0.0270
			Gemini	0.9494	0.9548	0.9441	0.0235
			SSB+Gemini2.5	0.9582	0.9556	0.9609	0.0250
			SSB+Gemini3	0.9861	0.9833	0.9888	0.0027

Table 5: Alignment results for five methods on VotingBooklets-Diamond across all voting booklets and languages. F1, Precision (Prec), and Recall (Rec) are computed via fuzzy matching against the gold standard. CER measures character error rate of the aligned text against gold (lower is better). Bold values indicate the best result per year, language, and metric. Languages: FR=French, IT=Italian, RM=Romansh. MiniLM: paraphrase-multilingual-MiniLM-L12-v2. SSB: Sentence-SwissBERT. Gemini: Alignment using Gemini-2.5-Flash-Lite. SSB+Gemini2.5: Sentence-SwissBERT with post-hoc correction using Gemini-2.5-Flash-Lite. SSB+Gemini3: Sentence-SwissBERT with post-hoc correction using Gemini-3-pro-preview.

algorithm searches for the globally optimal alignment by assigning scores to different operations: a correct match between a German segment and a target segment earns a score proportional to their cosine similarity; skipping an unmatched segment incurs a penalty of -0.3 ; and merging up to five consecutive segments on either side into a single unit is allowed but penalised by -0.05 per additional segment merged. When multiple segments are merged, their embeddings are averaged before computing similarity, and merges are only considered if the resulting similarity exceeds 0.30. The algorithm thus supports one-to-one, one-to-many, and many-to-one alignments. The best alignment path is recovered by backtracking through the DP grid.

Stage 2: Post-hoc correction with Gemini

Aligned pairs with a cosine similarity below 0.65 are flagged as uncertain and sent to Gemini 2.5 Flash Lite for review in batches of five. The model receives the German segment, the proposed target segment, and up to 150 candidate fragments from the target language. It is instructed to either confirm the alignment or replace the target segment using only verbatim text from the available fragments - no translation or paraphrasing is permitted. The prompt used is shown in Appendix A.4. A correction is applied only if Gemini returns a non-empty target segment and leaves the German text unchanged.

A.4 Post-hoc Correction Prompt

The following prompt was used for post-hoc alignment correction with Gemini. Placeholders in curly braces (`{lang_name}`, `{lang_key}`, etc.) are filled dynamically at runtime.

You are a multilingual text alignment expert for Swiss official documents.

You will receive a list of aligned segment pairs (German DE and `{lang_name}` `{lang_key}`). Some alignments may be incorrect - the `{lang_name}` text may be misaligned, incomplete, or merged incorrectly.

You also receive the full list of original `{lang_name}` fragments the aligner had available.

For each pair:

- If the alignment looks correct, keep it as-is.
- If the `{lang_name}` text is clearly wrong or misaligned, find the correct fragment(s) from the available fragments and replace it.
- If no good match exists, return an empty string for that pair.
- Do NOT translate or paraphrase - only use text

from the original fragments verbatim.

You must return VALID JSON ONLY.

Rules:

- No explanations
- No markdown
- No comments
- No trailing commas
- Escape all quotes
- Output must parse with `json.loads()`

Return exactly:

```
[
  {{
    "de": "...",
    "fr": "...",
    "it": "...",
    "rm": "..."
  }}
]
```

```
--- ALIGNED PAIRS TO REVIEW ---
{pairs}
```

```
--- AVAILABLE {lang_key_upper} FRAGMENTS ---
{fragments}
```

A.5 Output Format

The final aligned corpus is provided as JSON Lines files (`.jsonl`), one per vote, with each line containing a paragraph-level alignment across all available languages. All scripts and code used to create this corpus, including OCR extraction and multilingual alignment, are publicly available on our GitHub repository.⁵

⁵<https://github.com/Eljuanina/VotingBooklets>

Author Index

- Ahmadi, Sina, 102, 172
Amsler, Claudia, 215
An, Dmitriy, 53
Andrushevich, Aliaksei, 53
Assenmacher, Matthias, 2, 18, 30
Audrin, Bertrand, 112
- Blinière, Léa, 155
Boros, Emanuela, 155
Brandenberger, Laurence, 134
Brunner, Roland, 197
- Catherine, Ikae, 91
Chen, Zhengdao, 30
Cho, Steven, 147
Clematide, Simon, 155
Coltekin, Arzu, 215
Cusa, Jacopo Negro, 112
- D'Angelo, Agnese, 172
Daum, Moritz M., 172
de Spindler, Alexandre, 192, 197
Druta, Vlada, 180
- Ehrmann, Maud, 155
- Fischer, Dominic P., 126
- Galli, Samuele, 112
Glass, Michael, 64
Graber, Michael, 83
Grünert, David, 197
Guitton, Clement, 180
- Hahnloser, Richard, 147
Hampp, Noah, 64
Haraldsdóttir, Ragna Kemp, 91
Heierli, Jasmin, 192
Heumann, Christian, 30
Holenstein, Dario, 197
Hopton, Zachary, 126
- Kakooee, Reza, 83
Kaplan, Frederic, 155
Kiepora, Anna, 147
Kurpicz-Briki, Mascha, 91
- Lam, Jessica, 147
- Locher, Julia, 215
Loss, Giacomo, 18
Luo, Yanyang, 76
- Mambrini, Francesco, 226
Mayer, Simon, 180
Michel, Samuel, 112
Mirylenka, Katya, 64
Model, Charlotte, 102
Moretti, Giovanni, 226
Morgenstern, Barbara, 197
Méndez, Javier de Torres, 226
Müller-Csernetzky, Petra, 53
- Oesch, Jonas, 215
Özeren, Enes, 2
- Paice, Andrew, 53
Passarotti, Marco Carlo, 226
Peczinka, Florian, 197
Pellegrini, Matteo, 226
Perruchoud, Daniel Olivier, 83
- Rafnsdóttir, Guðbjörg Linda, 91
Rakhmatillokhon, Khoshimov, 76
Reveilhac, Maud, 180
Rogenhofer, Julneth, 112
Rudshin, Dmitry, 76
- Salamanca, Luis, 134
Santaholma, Marianne, 215
Schlosser, Sophia, 134
Schneider, Gerold, 180
Staudt, Yves, 206
Stephan, Andreas, 18
Stüssi, Elina, 234
- Tamò-Larrieux, Aurelia, 180
Timmel, Vincenzo, 83
Trembovelskyi, Bohdan, 180
- Vamvas, Jannis, 102, 126, 234
van der Plas, Lonneke, 112
Vannini, Alessia, 215
Vogel, Manfred, 83
Vásquez-Rodríguez, Laura, 112
- Wermelinger, Stephanie, 172

Yadav, Nisha, 180
Yang, Joshua C., 180

Yuan, Yifei, 134