Make Every Letter Count: Building Dialect Variation Dictionaries from Monolingual Corpora

Robert Litschko[1,2] **Verena Blaschke**[1,2] **Diana Burkhardt**[1]
Barbara Plank[1,2] **Diego Frassinelli**[1]
[1] MaiNLP, Ludwig Maximilian University of Munich, Germany
[2] Munich Center for Machine Learning (MCML), Munich, Germany
robert.litschko@lmu.de

Abstract

Dialects exhibit a substantial degree of variation due to the lack of a standard orthography. At the same time, the ability of Large Language Models (LLMs) to process dialects remains largely understudied. To address this gap, we use Bavarian as a case study and investigate the lexical dialect understanding capability of LLMs by examining how well they recognize and translate dialectal terms across different parts-of-speech. To this end, we introduce DIALEMMA, a novel annotation framework for creating dialect variation dictionaries from monolingual data only, and use it to compile a ground truth dataset consisting of 100K human-annotated German-Bavarian word pairs. We evaluate how well nine state-of-the-art LLMs can judge Bavarian terms as dialect translations, inflected variants, or unrelated forms of a given German lemma. Our results show that LLMs perform best on nouns and lexically similar word pairs, and struggle most in distinguishing between direct translations and inflected variants. Interestingly, providing additional context in the form of example usages improves the translation performance, but reduces their ability to recognize dialect variants. This study highlights the limitations of LLMs in dealing with orthographic dialect variation and emphasizes the need for future work on adapting LLMs to dialects.[1]

1. Judging Translation Pairs

German	Bavarian	Translation?
zweisprachig	*zwaasprochig*	yes
("bilingual")	*zwaspråchig*	yes
	zwoasprachign	infl.
	dreisprochige ("trilingual")	no
	...	...
dazwischen	*dozwischn*	yes
("in between")	*dawischn* ("to catch")	no
	daktischen ("tactical")	no
	...	...

2. Dialect-to-Standard Translation

Bavarian		German lemma
zwaasprochig	→	*zweisprachig*
zwaspråchig	→	*zweisprachig*
dozwischn	→	*dazwischen*
...	→	...

Table 1: We annotate whether Bavarian dialect words with high string similarity to Standard German lemmas are (direct or inflected) translations (1). We test whether LLMs are able to do the same, and whether they can translate dialectal words into their standard-language counterparts (2).

1 Introduction

Although most languages have standardized orthographies, this assumption does not hold for many languages and language varieties (Millour and Fort, 2020). One prominent example are non-standard dialects, which have become increasingly popular in natural language processing (NLP) research (Faisal et al., 2024; Joshi et al., 2025).

Dialects exhibit a large degree of spelling variation in written data, owing both to individual pronunciation differences and spelling preferences (Just and Widmer, 2025). Such variation is commonly reflected in dialect NLP datasets (cf. Blaschke et al., 2023a; Ramponi, 2024). However, current NLP methods – which typically are trained on large amounts of standard-language data – are badly equipped to deal with such spelling variation, even if the word forms are similar to the standard. Issues range from subword tokenization (Blaschke et al., 2023b; Srivastava and Chiang, 2025), to evaluating generative tasks (Aepli et al., 2023), to retrieving information without being able to match word spellings (Litschko et al., 2025). Therefore, it is important to build NLP systems that are either ro-

[1] https://github.com/mainlp/dialemma

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 14157–14174
November 4-9, 2025

bust to variation or good at normalization. In order to gauge how good current or future systems are at this, we need evaluation datasets. However, prior works on collecting datasets of real-life dialectal spelling variation have resulted in relatively small datasets (Section 2).

To allow large-scale analyses of the impact of dialectal spelling variation on NLP tasks, we conduct a case study on the Wikipedia edition in Bavarian, a dialect group related to German. This dialect wiki is comparatively large and has been used in prior work on dialect NLP (Artemova and Plank, 2023; Peng et al., 2024; Blaschke et al., 2024; Litschko et al., 2025), and is included in pre-training datasets for, e.g., mBERT (Devlin et al., 2019). It covers the entire Bavarian-speaking area (regardless of sub-dialect), and explicitly encourages contributors to spell words as they find appropriate.[2]

Our approach only requires access to one monolingual corpus per language variety. Taking advantage of the fact that related standard and dialect varieties have a high number of cognates with similar word forms,[3] we automatically extract German–Bavarian word pairs with high string similarities and annotate whether they are (direct or inflected) translations of each other (Table 1). We use our dataset to systematically analyze how well recent large language models (LLMs) can perform such annotations, and how good they are at translating the Bavarian terms into their German counterparts. Our study makes the following contributions:

- We introduce a **novel annotation framework** for creating dialect variation dictionaries from monolingual data, without requiring parallel corpora (Section 3.1).
- For 10k German words, we extract and manually annotate similarly spelled Bavarian words, resulting in **11k direct German–Bavarian translation pairs** and **7k pairs with inflection differences** (Section 3.2).
- We **evaluate nine state-of-the-art LLMs** on two tasks: **judging** whether a given word pair is a translation (Section 5.1), and **translating** Bavarian words into their German counterparts (Section 5.2). We show that, overall, larger models perform better than their smaller versions, but they still struggle to distinguish between translations and inflected variants (Section 5.3).
- We further show that (i) including additional context in the form of usage examples, on average, improves the model's ability to translate and hurts the performance in judging translation candidates, (ii) using German prompts leads to worse results, and (iii) the Levenshtein distance between word pairs crucially impacts the model performance (Section 5.3).

2 Related Work

Inducing dialect and low-resource language dictionaries Artemova and Plank (2023) induce dialect dictionaries from parallel articles in German and dialectal wikis, extracting ∼800 word pairs per language pair, and manually verifying them. Similarly, Haddow et al. (2013) extract German–Bavarian word lists from parallel and near-parallel sentences. Unlike these works, our approach does not rely on parallel data.

Burghardt et al. (2016) and Burghardt (2023) crowdsourced German translations for 259 Bavarian words, while Millour and Fort (2019) collected spelling variants for 145 Alsatian words, resulting in about seven variants per word. Schmidt et al. (2020) manually translated a German word list into Swiss German dialects. Compared to these efforts, we do not require large-scale manual translations.

Litschko et al. (2025) automatically induced translations for dialectal terms in seven German dialects and regional languages based on Wikipedia article titles and links, also collecting some spelling (and, rarely, lexical) variations. Unlike this approach, we are not limited to wiki article topics. Ylonen (2022) uses Wiktionary as a structured data source for extracting word-level translations (with inflection information). However, dialects are not thoroughly covered in Wiktionary.

The idea behind finding dialect/standard translation pairs is similar to work on identifying cognate pairs in more distantly related languages, for which complex statistical alignment methods have been developed (Inkpen et al., 2005; Haghighi et al., 2008; Wettig et al., 2011; Kontonatsios et al., 2014, *inter alia*). We are however interested in finding

[2] bar.wikipedia.org/wiki/Wikipedia:Wia_schreib_i_a_guads_Boarisch? "How do I write good Bavarian?".

[3] We acknowledge that there are other kinds of differences between standard and dialect varieties, e.g., lexical differences due to regional word choices. In this study, we focus on *spelling differences between cognate words* as the spelling variation is a specific challenge for NLP systems.

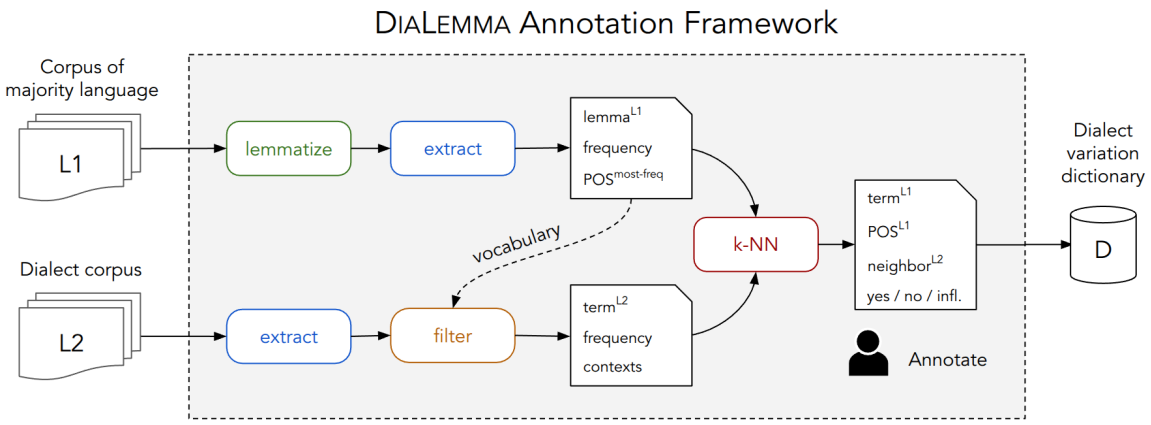

Figure 1: DIALEMMA annotation framework. Step 1: Lemmatize the corpus of the majority language. Step 2: Extract the vocabularies from both the corpora of the majority language and dialect, respectively. Step 3: Filter the lemmas of the majority language out of the dialect vocabulary. Step 4: Find for each lemma the lexically most similar dialect terms using the Levenshtein distance. Step 5: Annotate if dialect terms are translations ('yes'), dialectal inflected forms ('inflected'), or neither ('no').

word pairs in closely related varieties, and we allow many-to-one dialect-to-standard mappings to account for spelling variations. Li et al. (2023) fine-tune LLMs to predict word-level translations between pairs of languages, showing that models are sensitive to prompt choices and perform poorly on low-resource languages.

Dialect-to-standard translation and normalization Translating dialectal words and texts into the corresponding standard language is a well-established task within dialect NLP (Zampieri et al., 2020). Similar efforts have focused on normalizing historical spellings (Bollmann, 2019), and research on normalizing social media data has also included normalizing phonetic spellings or slang terms into the corresponding standard language forms (van der Goot et al., 2018). Popular strategies for normalizing dialect spellings on a word or sentence level use character-based machine translation (Honnet et al., 2018; Kuparinen et al., 2023; Scherrer, 2023). More recent works have also treated sentence-level dialect-to-standard translation like a common translation task, using encoder-decoder transformer architectures with subword tokens (Kuparinen et al., 2023; Kresic and Abbas, 2024; Her and Kruschwitz, 2024).

Our focus lies on instruction-tuned LLMs, as they might normalize data in an unsupervised way. Alam and Anastasopoulos (2025) prompt instruction-tuned LLMs for sentence-level dialect normalization, observing poor zero-shot performances. We study LLMs on the word level.

3 Method

In this section, we first introduce DIALEMMA, our annotation framework for obtaining dialect variation dictionaries (Section 3.1). We then introduce our dataset (Section 3.2), which we obtained by applying DIALEMMA to the Bavarian Wikipedia. Based on this, we introduce two word-level dialect tasks (Section 3.3).

3.1 DIALEMMA Annotation Framework

Figure 1 shows our annotation framework for obtaining dialect variation dictionaries from monolingual corpora. We require two monolingual corpora, one in a standard language L_1 (e.g., German), and one in a related non-standardized dialect L_2 (e.g., Bavarian). The goal is to match spelling variations in L_2 (*zwaasprochig, zwaspråchig, zwoasprochig, zwasprochig*; "bilingual") to their normalized counterpart in L_1 (*zweisprachig*). We additionally match inflected L_2 word forms (like *zwoasprochign*)[4] to the corresponding L_1 lemmas. DIALEMMA frames the extraction of dialect variation dictionaries as a one-to-many matching problem between the two vocabularies, extracted from monolingual corpora, where judgments about word matches are done by native speakers. In Section 5, we investigate to what extent LLMs can be used in place of human judges.

In principle, each dialect term could be a match for any term in the majority language. However, manually comparing all possible pairs is intractable

[4]The suffix *-n* marks DEF.PL, DAT, or MASC.ACC.

for large vocabularies. To reduce the search space, DIALEMMA first applies part-of-speech (POS) tagging and lemmatization on the L_1 corpus (we use spaCy, Honnibal et al., 2020, for this).[5] We then extract the vocabulary of L_1 consisting of unique lemmas in $\mathcal{V}^{L_1}$. For each lemma, we extract also its most frequent POS tag (POS^{max}) as additional information for annotators. In practice, we limit the vocabulary to the most frequent n standard-language terms. Since there are no morphosyntactic analyzers for most dialects, we directly extract the list of *all* unique terms found in L_2. Importantly, as a result of dialectal spelling variations, many dialect terms have a low term frequency, which is why we do not limit $\mathcal{V}^{L_2}$ by frequency. To further reduce the search space, we filter out tokens that are shared between $\mathcal{V}^{L_1}$ and $\mathcal{V}^{L_2}$.

In the next step of DIALEMMA, the k nearest neighbors for each lemma are extracted based on their Levenshtein (1966) distance (LD). If more than k dialect terms have the same LD, we select k of them at random. As a result, we obtain for each lemma its POS^{max}, a set of k possible dialect variations (candidates) and term frequencies. To provide annotators with further context, we include up to c examples in which the dialect term is used.

The final step involves native speakers, who compare lemmas in the majority language to matched dialect terms, and annotate whether the dialect term corresponds to a dialect translation ('yes'), an inflected dialect translation ('inflected'), or neither of the two ('no') (see Table 1). For the 'inflected' class, we refer to forms inflected in ways distinct from the lemma form (e.g., infinitives are considered exact matches rather than inflected variants). We distinguish between 'yes' and 'inflected' so that our work can be used to induce bilingual dictionaries for lemmas and inflected translations (since forms with different inflectional morphology do not correspond to direct translations). We lemmatize the German words since dictionaries usually are on the lemma level, and since this reduces the search space for finding matches between German and Bavarian words, allowing us to annotate a greater number of (entirely) different words.

All instances annotated as 'yes' or 'inflected' are included in the final dialect variation dictionary. This allows us to analyze the performance of LLMs with respect to different POS classes, and between dialectal translations and inflections.

[5] https://spacy.io/, v3, *de_core_news_lg* model

	Yes	Inflected	No
Noun	6,720	2,670	28,480
Adjective	1,358	3,066	5,496
Adverb	1,157	—	2,783
Verb	934	1,182	6,214
Proper Noun	574	86	34,430
Total	11,044	7,070	81,586
Noun	6,564	—	—
Adjective	1,325	—	—
Adverb	1,126	—	—
Verb	916	—	—
Proper Noun	556	—	—
Total	10,775	—	—

Table 2: Label distributions for the test set of the judgment task (top half) and translation task (bottom half). We show the numbers for the entire test split as well as for the five most frequent parts of speech in the test set. Statistics for all POS classes are shown in Table 7.

3.2 Dataset

We use DIALEMMA on monolingual corpora we extract from the Bavarian and German Wikipedias (CC BY-SA 4.0). In total, we collect for each of 10k German lemmas the $k = 10$ nearest neighbors, each with $c = 3$ local contexts, consisting of fifty characters before and after a dialect term within a sentence. Note that local contexts are not used in the extraction process. We extract it to provide additional information for human annotators and to investigate whether it improves the performance of LLMs in recognizing and translating dialect variants. This constitutes a total number of 100k Bavarian–German word pairs, out of which we hold out 300 instances as a development set for prompt selection. All instances are manually annotated by two in-house undergraduate and one postgraduate student, who are native speakers of Bavarian and German. The annotators show a high consistency in their judgments, with a Fleiss' kappa (Fleiss, 1971) score of 0.86.

As shown in Table 2, our annotated dataset contains Bavarian translations for $11,044$ German lemmas, and inflected Bavarian translations for $7,070$ lemmas (test set). German lemmas for which we found direct Bavarian translations are mapped to 2.61 ± 1.88 spelling variations on average. Lemmas for which we found inflected counterparts are mapped to 2.57 ± 1.90 inflected Bavarian forms on average. Bavarian translations have an average

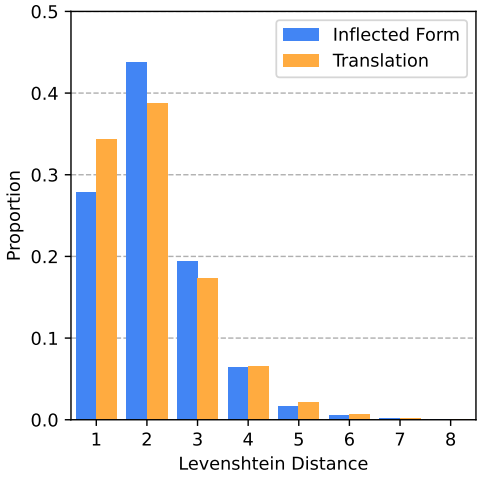

Figure 2: Distribution of word pairs by Levenshtein distance. The bars indicate the proportion of extracted translations (i.e., exact matches; in orange) versus inflected forms (in blue) at each distance value.

Levenshtein distance of 2.07 ± 1.07 from their German counterparts. Inflected Bavarian terms have a Levenshtein distance of 2.13 ± 1.01. Word pairs that are *not* translations of each other ('inflected' or 'no') have an average Levenshtein distance of 2.72 ± 1.18. Figure 2 presents the percentage of word pairs for each Levenshtein distance; the bars distinguish between Bavarian translations (in orange) and inflected forms (in blue).

3.3 NLP Tasks: Can LLMs Understand Spelling Variation?

We now describe the two benchmarks that we build to systematically evaluate the orthographic dialect understanding capabilities of LLMs. The focus of our work lies on Bavarian and German.

Judging translation candidates We test whether LLMs can step in the role of annotators and judge whether Bavarian candidate words correspond to a given German lemma (Figure 1). LLMs are presented with a word pair and need to classify their relationship into one of the three possible classes 'yes', 'inflected', and 'no'. Since the class distribution is heavily skewed towards the 'no' class, we choose the macro-averaged F1 score as our evaluation metric, treating all classes equally. In Table 2 (top), we show the label distribution for the test split of the annotated dataset. The word pairs included in this task have an average Levenshtein distance of 2.60 ± 1.18. We reserve a development set for LLM prompt selection (Section 4.2).

Dialect-to-standard translation We also test the ability of LLMs translate Bavarian terms to German. For this, we use all instances with the label 'yes'. Instances belonging to the other two classes cannot be used due to the lack of reference translations. We report the word-level translation accuracy and limit the output of LLMs to at most 20 tokens. Instances where LLMs output more than one word are considered instruction-following errors.

4 Experimental Setup

4.1 Language Models

We conduct our experiments using nine state-of-the-art open-source large language models. We experiment with different model families and model sizes, including both smaller and larger variants for each family. Specifically, we use `Aya-expanse-8b` and `Aya-expanse-32b` (Dang et al., 2024); `Gemma3-12b` and `Gemma3-27b` (Team et al., 2025); `Llama3.1-8b`, `Llama3.3-70b`, and `Llama4-17b` (Grattafiori et al., 2024); as well as `Mistral-7b` and `Mistral-123b` (Jiang et al., 2023). For all LLMs, we use the instruction-tuned version and greedy decoding (temperature = 0).

4.2 Prompt Selection

LLMs are known to generate different outputs under minor prompt modifications. For this reason, we develop a pool of, respectively, 21 and 13 English-language prompts for the judgment and translation task (Appendix C). We follow Li et al. (2023) and use a development set to select the best-performing prompt for each LLM. For the judgment and translation task, the development set contains 300 instances, respectively. Our main results are based on these prompts, however we carry out two additional experiments to determine the effect of the prompt selection:

Prompts with context. We extend the best prompts to also contain a usage example of the dialect word (see Tables 22 and 23 in Appendix C). Here, the contexts are extracted from those used in the annotation process and correspond to example usages found in the Wikipedia dump.

German-language prompts. Our original list of prompts is in English, since this is the primary language of the models we use. To determine whether prompts in the standard language of the dataset work better, we also evaluate the performance of the best-performing prompts, after translating them into German (see Tables 24 and 25 in Appendix C).

Model	Noun	Adjective	Adverb	Verb	Proper Noun	Overall	IF Error
Random	0.268	0.309	0.257	0.280	0.174	0.250	–
Levenshtein	0.355	0.307	0.331	0.307	0.202	0.284	–
Majority Label ('no')	0.286	0.238	0.271	0.285	0.330	0.300	–
Logistic Regression	0.351	0.310	0.270	0.344	0.312	0.364	–
Mistral-7b	0.292	0.270	0.278	0.335	0.330	0.332	0.001
Mistral-123b	**0.561**	**0.488**	**0.484**	**0.517**	**0.473**	**0.567**	0.026
Llama3.1-8b	0.398	0.305	0.300	0.351	0.383	0.399	0.003
Llama3.3-70b	0.444	0.369	0.379	0.340	0.359	0.428	0.000
Llama4-17b	0.354	0.394	0.237	0.338	0.309	0.368	0.438 (!)
Aya-expanse-8b	0.447	0.332	0.392	0.390	0.436	0.436	0.000
Aya-expanse-32b	0.500	0.411	0.442	0.469	0.414	0.499	0.005
Gemma3-12b	0.492	0.444	0.441	0.430	0.390	0.496	0.000
Gemma3-27b	0.453	0.431	0.288	0.323	0.261	0.418	0.001
average	0.438	0.384	0.382	0.388	0.373	-	-

Table 3: **Translation candidate judgments** – Macro F1 scores for each model across the five most frequent POS tags. The final two columns report the average Macro F1 across 15 POS (See Table 16) and the overall percentage of outputs where LLMs failed-to-follow prompt instructions (IF Error). Boldface font indicates the **highest scores**.

4.3 Baselines

To assess the ability of LLMs to judge translation candidates, we compare their performance against several baselines. We include a random baseline (Random), a heuristic based on the Levenshtein distance (LD), where instances with a distance of two or less are classified as 'yes' and otherwise as 'no', and a majority label baseline (Majority Label) that always predicts the majority label 'no'. Additionally, we train a logistic regression model on the development set, and use the Levenshtein distance and Jaccard similarity between character bi- and trigrams of the German lemma and Bavarian candidate as input features.

5 Results & Discussion

We report the task performance for each LLM and part of speech (POS), and across all parts of speeches. We additionally report the percentage of outputs where LLMs fail to follow prompt instructions (instruction following error rate; **IF Error**). In the Appendix, we show a list of all POS categories and number of instances (Table 7), and results for each POS category (Tables 16 and 17).

5.1 Judging Translation Candidates

Table 3 reports macro-F1 scores for the top five POS categories, together with each model's overall macro-F1 (averaged over all 15 POS tags).

Overall Performance The results show clear differences across models' performance. In general, for each model family the larger version of the model outperforms the smaller version. Mistral-123b obtains the highest overall macro-F1 (0.567), indicating high accuracy across POS tags. Other strong models are Aya-expanse-32b (0.499) and Gemma3-12b (0.496), both of which also have very low IF Error rates.

Conversely, Llama4-17b, despite being a newer model, shows lower overall performance (0.368) and the highest IF Error (0.438) among all models. While the model can occasionally produce correct outputs, it often fails to follow the requested format and it returns overly long responses rather than the expected single-word answers. In contrast, models such as Mistral-7b and Gemma3-27b also achieve lower overall scores despite relatively low IF Error rates, suggesting a gap between instruction-following and actual performance at judging translation candidates. Importantly, several models including Llama3.3-70b, Aya-expanse-8b, and Gemma3-12b show an IF Error of zero – all their outputs followed the prompt requirements, even if the predictions were not always accurate.

POS-Specific Trends Mistral-123b has the best results in all five major POS categories, with especially high results on nouns (0.561) and verbs (0.517). Gemma3-12b and Aya-expanse-32b also

Model	Noun	Adjective	Adverb	Verb	Proper Noun	Overall	IF Error
Mistral-7b	0.085	0.003	0.000	0.000	0.101	0.058	0.764 (!)
Mistral-123b	0.552	0.381	0.374	0.425	0.511	0.493	0.018
Llama3.1-8b	0.371	0.189	0.154	0.105	0.390	0.298	0.028
Llama3.3-70b	0.596	0.502	0.498	0.505	0.590	0.563	0.019
Llama4-17b	**0.610**	**0.560**	0.514	0.527	0.586	0.582	0.025
Aya-expanse-8b	0.403	0.334	0.340	0.377	0.335	0.379	0.050
Aya-expanse-32b	0.524	0.452	0.428	0.440	0.487	0.494	0.039
Gemma3-12b	0.516	0.476	0.429	0.397	0.433	0.483	0.013
Gemma3-27b	0.609	0.534	**0.543**	**0.563**	**0.594**	**0.586**	0.016
average	0.474	0.381	0.364	0.371	0.447	-	-

Table 4: **Dialect-to-standard translation** – Accuracy scores for each model across the five most frequent POS tags. The final two columns report the average accuracy across 15 POS (See Table 17) and the overall percentage of outputs that failed to follow prompt instructions (IF Error rate). Bold font shows the **highest scores**.

obtain high macro-F1 scores across most categories, with the former showing strong performance on nouns (0.492), and the latter scoring well on nouns (0.500) and verbs (0.469).

POS difficulty varies across models: nouns (0.438) tend to be the easiest to be correctly classified, while proper nouns (0.373) and adverbs (0.382) often are more challenging. Even models with high overall performance, like Mistral-123b, show a drop in correctly judging proper noun pairs.

Qualitative Analysis We analyzed the classification predictions of the best model (Mistral-123b). Overall, we did not find many striking patterns as to which terms get classified correctly or not. Many inflected forms whose inflection status should be obvious due to being identical to the German lemma plus a suffix are misclassified as translations without inflection differences (e.g., the Bavarian inflected form *billign*[6] is predicted to be an exact translations of the German lemma *billig* "cheap"). Such misclassifications might be influenced by tokenizers that do not usually split such inputs along morpheme lines. Bavarian adjective forms ending with *-schn* (stem ending with *-sch* followed by the inflectional suffix *-n*) almost always get misclassified (most often as 'yes'), but we do not know why these cases should be especially difficult.

Many of the cases where this model fails to follow the instruction of simply outputting a label are words from the Bavarian wiki that are written in non-Latin characters (gold: 'no').

[6] *billig-n* "cheap"-DAT / MASC.ACC / DEF.PL.

5.2 Dialect-to-Standard Translation

Table 4 presents the results for the translation task. Once more, the results indicate high variation in translation quality across model families, sizes, and POS categories.

Overall Performance Larger and more recent instruction-tuned models consistently outperform smaller ones, showing both substantially higher accuracy scores and lower IF Errors – indicating stronger adherence to the task instructions. It is noteworthy that the best-performing model in the translation task (Gemma3-27b) is different from the best model in judgment task (Mistral-123b). Gemma3-27b achieves the highest overall accuracy score (0.586), closely followed by Llama4-17b (0.582) and Llama3.3-70b (0.563). In contrast, Mistral-7b performs extremely poorly, with an almost zero overall accuracy. Its extremely high IF Error (0.764) further indicates its inability to follow instructions and produce valid outputs.

POS-Specific Trends Across individual POS categories, Gemma3-27b shows the best performance for adverbs (0.543), verbs (0.563), and proper nouns (0.594), while Llama4-17b achieves the highest scores for nouns (0.610) and adjectives (0.560). Consistent with results on the judgment task, we find that nouns are the easiest dialect words to translate. However, LLMs show generally higher IF Error rates on the translation task.

Qualitative Analysis In a few cases, models output a term that the automatic evaluation fails to recognize as correct because of spelling differences

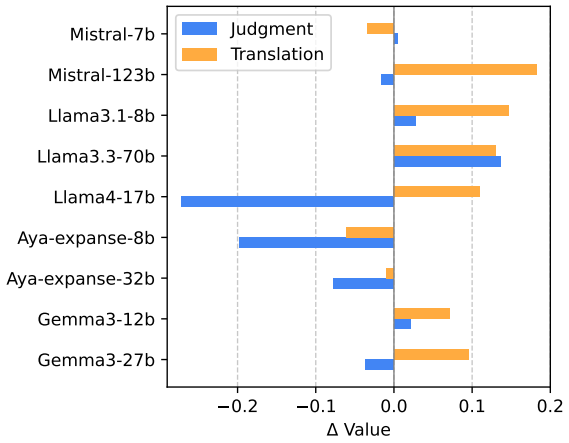

Figure 3: **Context effects** – Overall changes (Δ) in macro F1 for judging translation candidates (blue bars) and in accuracy for translation (orange bars), measured as the difference between contextualized and non-contextualized prompts.

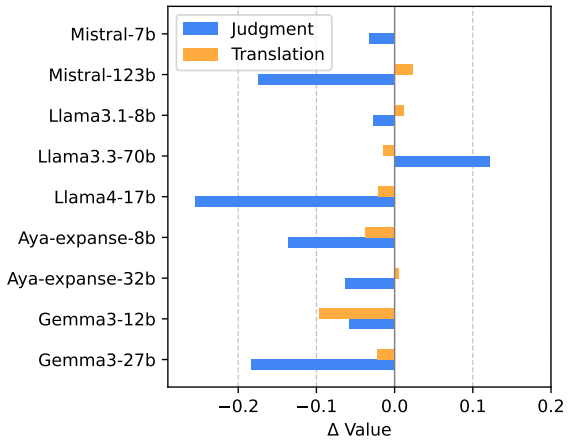

Figure 4: **Language effects** - Overall changes (Δ) in macro F1 for judging translation candidates (blue bars) and in accuracy for translation (orange bars), as the difference between prompts in German and English.

sanctioned by German orthography (e.g., *geografisch* vs. *geographisch*; "geographic"). When analyzing the outputs of the best-performing translation model (`Gemma3-27b`), we observe a range of different error types: Sometimes, the model is just slightly wrong. It might produce a related word with different derivative morphology: *Literaturwissnschoftla* ("literary scholar"; German: *Literaturwissenschaftler*) gets translated as *Literaturwissenschaft* ("literary studies"), or it produces a nonce word that is just one letter away from the real word: *Dreifoitichkeit* ("trinity"; German: *Dreifaltigkeit*) becomes *Dreifältigkeit*. Many words are compound words or they contain prefixes. Frequently, only part of the word is translated correctly: *Iabaprifung* ("audit", lit. "over+test"; German: *Überprüfung*) becomes *Jahresprüfung* ("yearly test", lit. "year+test"). However, there are also many translations that are entirely dissimilar: *Vameahrung* ("proliferation"; German: *Vermehrung*) becomes *Wanderung* ("hike"). More generally, there are many cases where a model correctly recognizes a translation pair ('yes') but fails the translation task, or where a model wrongly predicts 'no' or 'inflected' instead of 'yes' and still produces the correct translation.

5.3 Additional Analyses

Effect of Context Figure 3 shows that adding a short local context (see Section 3.2) to the prompts affects model performance to different degrees when judging potential translation candidates (blue bars). Among larger models, context generally causes a slight drop in macro-F1 scores. An exception is `Llama3.3-70b`, which shows the largest improvement, outperforming `Mistral-123b` and becoming the best model in this condition (macro F1: 0.564 vs. 0.551). In contrast, smaller models show small improvements when adding context but still remain behind their larger counterparts. `Llama4-17b` shows a substantial drop in its performance, indicating its limited ability to incorporate contextual information effectively. When inspecting the IF Error (see left-side Table 26 in the Appendix), we observe that adding context strongly decreases the probability of following instructions for `Mistral-7b`, `Mistral-123b`, and `Aya-expanse-32b`. In contrast, `Llama4-17b` benefits from contextualization, showing improvements in its ability to follow instructions.

In the translation task (orange bars), context has a more systematic positive effect: most models show small improvements when additional context is provided. The only exceptions are `Mistral-7b`, `Aya-expanse-8b`, and `Aya-expanse-32b`, where context slightly reduces performance. Adding context in the translation task slightly worsens instruction-following ability of LLMs (higher IF Error rates) in exchange for an overall improved translation performance. This observation is consistent with the pattern we see when comparing smaller to larger models (see discussion below).

Effect of Prompt Language For judging the Bavarian–German word pairs, using prompts in German has a negative impact on model performance for all models except `Llama3.3-70b`. The

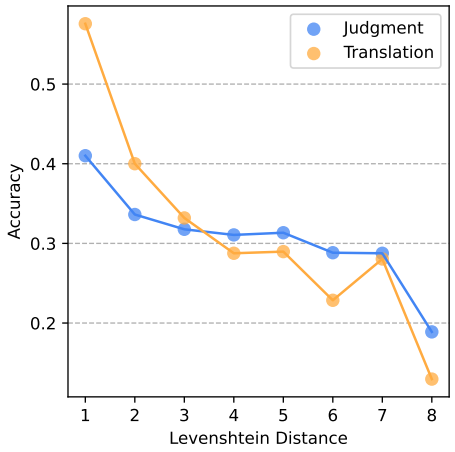

Figure 5: Accuracies for judging translation candidates (excluding the majority class 'no') and translations with respect to Levenshtein distances. Results aggregated over models and POS tags.

Predicted	Actual		
	yes	inflected	no
yes	0.76%	0.41%	0.52%
inflected	5.50%	7.09%	3.16%
no	93.46%	92.29%	96.27%
IF Error	0.28%	0.21%	0.05%

Table 5: Recognition task: Confusion matrix for `Mistral-7b` showing the relative distribution of predictions for each class. F1 scores per class: 0.015 ('yes'), 0.093 ('infl.') and 0.888 ('no').

Predicted	Actual		
	yes	inflected	no
yes	72.05%	47.93%	8.01%
inflected	1.02%	15.97%	1.62%
no	25.86%	34.89%	87.45%
IF Error	1.07%	1.20%	2.91%

Table 6: Recognition task: Confusion matrix for `Mistral-123b`. Each column shows the relative distribution of predictions. F1 scores per class: 0.550 ('yes'), 0.234 ('infl.') and 0.902 ('no').

change in language also has a strong negative effect on the ability of following instructions, especially for smaller models (see right-side Table 26 in the Appendix). The only strong exception is `Llama4-17b`, which shows a big improvement in following instructions that written in German. For the translation task, using German prompts only minimally affects model's performance and IF Error, with some models showing slight improvements and others slight drops, but without any consistent pattern across model size or family.

Impact of Levenshtein Distance Figure 5 shows the accuracy across the first eight Levenshtein distance (LD) values for judging Bavarian–German translation candidates and translating Bavarian words into German, respectively. It is important to mention that for this analysis we excluded the majority class 'no' for the judgment task, since models are biased towards predicting 'no' (see discussion below). Our results reveal a similar trend in how LD affects LLM performance on both tasks. We find that the model performance deteriorates as word pairs become more lexically distant. Bavarian words like *Basketboispuia* ("basketball player", German: *Basketballspieler*; LD $= 8$) and *Dochdauntanehmen* ("Subsidiaries", German: *Tochterunternehmen*; LD $= 6$) have a higher LD to their German translation and are more difficult for LLMs. Both translations were incorrectly classified as being unrelated ('no') by `Mistral-123b`.

Impact of Model Size Our analysis of the ability of LLMs to recognize translation pairs showed that smaller models generally perform worse than their larger variants (Table 3). To gain a deeper understanding of the results, we break down the macro-averaged F1 scores into per-class distributions. Tables 5 and 6 show the confusion matrices (%) for `Mistral-7b` and `Mistral-123b`. We can see that `Mistral-7b` has a strong bias for predicting 'no'. When moving to the larger `Mistral-123b` model, however, we find that the predictions shift from 'no' (before: 93.46% and 92.29% of 'yes' and 'inflected' instances) towards the other two classes (after: 25.86% and 34.8% of 'yes' and 'inflected' instances). We also observe a slightly higher IF Error rate for larger models. Please refer to Appendix B for further analyses.

6 Conclusion

We introduce DIALEMMA, a novel dialect annotation framework and build a dataset consisting of German lemmas and Bavarian variants. We conduct a systematic analysis of the ability of LLMs on two tasks: judging whether superficially similar dialect–standard word pairs are translations of each other, and translating dialect variants into their standard-language counterparts. Our results show that their performance varies by model size and the part-of-speech of the input word. In future work, we plan to 1) fine-tune LLMs for identifying translation pairs and as translators and 2) extend our evaluation protocol to extrinsic evaluation on downstream tasks. We will release our code and dataset for future uptake (CC BY-SA 4.0).

Limitations

The main limitation of our study is its focus on Bavarian as a single case study. While this allowed for a more precise and controlled investigation, we see a lot of potential to extend this approach to other low-resource language scenarios.

Even though we added contextual information (e.g., short Wikipedia extracts) to better align the model setup with human annotation conditions, we acknowledge that context can influence model performance in more complex ways than those explored here. Future work should systematically analyze the role of contextual information in identifying dialect–standard translation pairs and generating dialect-to-standard translations. Finally, the focus of this work lies on spelling variation of cognates with the standard language. By definition of the Levenshtein distance, this excludes other differences, such as regional word choices.

We used SpaCy for lemmatization and POS tagging. Although the tool works well (the model documentation reports an accuracy of 98 % for both tasks),[7] it is not perfect, occasionally producing an incorrect lemma or mis-tagging words. Inspecting our data, we found that this might especially be the case for adjectives that wrongly were tagged as adverbs.

Ethical considerations

We see no ethical issues related to this work. All experiments were conducted with publicly available data and open-source software, and we have made all of our code and linguistic resources openly available for reproducibility. All annotators were fairly compensated for their work. One of the three annotators was hired as a research assistant and paid according to standard national wages. The other two annotators contributed their annotations as part of their thesis work. We transparently communicated the use of their annotations and obtained explicit consent prior to the start of the project.

Use of AI Assistants

The authors acknowledge the use of ChatGPT for correcting grammatical errors, enhancing the coherence of the final manuscripts, and providing assistance with prompt engineering.

[7] https://spacy.io/models/de#de_core_news_lg-accuracy

Acknowledgments

We thank Lisa Miller and Miriam Winkler for their help with annotating the dataset. This research is in parts supported by European Research Council (ERC) Consolidator Grant DIALECT 101043235.

References

Noëmi Aepli, Chantal Amrhein, Florian Schottmann, and Rico Sennrich. 2023. A benchmark for evaluating machine translation metrics on dialects without standard orthography. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1045–1065, Singapore. Association for Computational Linguistics.

Md Mahfuz Ibn Alam and Antonios Anastasopoulos. 2025. Large language models as a normalizer for transliteration and dialectal translation. In *Proceedings of the 12th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 39–67, Abu Dhabi, UAE. Association for Computational Linguistics.

Ekaterina Artemova and Barbara Plank. 2023. Low-resource bilingual dialect lexicon induction with large language models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 371–385, Tórshavn, Faroe Islands. University of Tartu Library.

Verena Blaschke, Barbara Kovačić, Siyao Peng, Hinrich Schütze, and Barbara Plank. 2024. MaiBaam: A multi-dialectal Bavarian Universal Dependency treebank. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10921–10938, Torino, Italia. ELRA and ICCL.

Verena Blaschke, Hinrich Schuetze, and Barbara Plank. 2023a. A survey of corpora for Germanic low-resource languages and dialects. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 392–414, Tórshavn, Faroe Islands. University of Tartu Library.

Verena Blaschke, Hinrich Schütze, and Barbara Plank. 2023b. Does manipulating tokenization aid cross-lingual transfer? a study on POS tagging for non-standardized languages. In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 40–54, Dubrovnik, Croatia. Association for Computational Linguistics.

Marcel Bollmann. 2019. A large-scale comparison of historical text normalization systems. 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 3885–3898, Minneapolis, Minnesota. Association for Computational Linguistics.

Manuel Burghardt. 2023. Bairisch 2.0 – Erstellung eines Social Media-Dialektlexikons mithilfe von Crowdsourcing. In Elena Arestau, Julia Burkhardt, Nastasia Herold, Rebecca Sierig, and Marie Amnisius, editors, *Vielfalt und Integration-diversitá ed integrazione-diversité et intégration: Sprache (n) in sozialen und digitalen Räumen – Eine Festschrift für Elisabeth Burr*. Universitätsbibliothek Leipzig.

Manuel Burghardt, Daniel Granvogl, and Christian Wolff. 2016. Creating a lexicon of Bavarian dialect by means of Facebook language data and crowdsourcing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2029–2033, Portorož, Slovenia. European Language Resources Association (ELRA).

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, and 1 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

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.

Fahim Faisal, Orevaoghene Ahia, Aarohi Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. DIALECTBENCH: An NLP benchmark for dialects, varieties, and closely-related languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14412–14454, Bangkok, Thailand. Association for Computational Linguistics.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The Llama 3 herd of models.

Barry Haddow, Adolfo Hernández, Friedrich Neubarth, and Harald Trost. 2013. Corpus development for machine translation between standard and dialectal varieties. In *Proceedings of the Workshop on Adaptation of Language Resources and Tools for Closely Related Languages and Language Variants*, pages 7–14, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: HLT*, pages 771–779, Columbus, Ohio. Association for Computational Linguistics.

Wan-hua Her and Udo Kruschwitz. 2024. Investigating neural machine translation for low-resource languages: Using Bavarian as a case study. In *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 155–167, Torino, Italia. ELRA and ICCL.

Pierre-Edouard Honnet, Andrei Popescu-Belis, Claudiu Musat, and Michael Baeriswyl. 2018. Machine translation of low-resource spoken dialects: Strategies for normalizing Swiss German. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in Python.

Diana Inkpen, Oana Frunza, and Grzegorz Kondrak. 2005. Automatic identification of cognates and false friends in French and English. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 20005*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Aditya Joshi, Raj Dabre, Diptesh Kanojia, Zhuang Li, Haolan Zhan, Gholamreza Haffari, and Doris Dippold. 2025. Natural language processing for dialects of a language: A survey. *ACM Comput. Surv.*, 57(6).

Erika Just and Paul Widmer. 2025. A corpus approach to orthographic chunking: near-naive word separation in Swiss German text messages. *Corpus Linguistics and Linguistic Theory*.

Georgios Kontonatsios, Ioannis Korkontzelos, Jun'ichi Tsujii, and Sophia Ananiadou. 2014. Combining string and context similarity for bilingual term alignment from comparable corpora. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1701–1712, Doha, Qatar. Association for Computational Linguistics.

Mihael Kresic and Noorhan Abbas. 2024. Normalizing Swiss German dialects with the power of large language models. *Procedia Computer Science*, 244:287–295. 6th International Conference on AI in Computational Linguistics.

Olli Kuparinen, Aleksandra Miletić, and Yves Scherrer. 2023. Dialect-to-standard normalization: A large-scale multilingual evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13814–13828, Singapore. Association for Computational Linguistics.

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710. [Russian original (1965) in Doklady Akademii Nauk SSSR, 163(4):845—848.].

Yaoyiran Li, Anna Korhonen, and Ivan Vulić. 2023. On bilingual lexicon induction with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9577–9599, Singapore. Association for Computational Linguistics.

Robert Litschko, Oliver Kraus, Verena Blaschke, and Barbara Plank. 2025. Cross-dialect information retrieval: Information access in low-resource and high-variance languages. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10158–10171, Abu Dhabi, UAE. Association for Computational Linguistics.

Alice Millour and Karën Fort. 2019. Unsupervised data augmentation for less-resourced languages with no standardized spelling. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 776–784, Varna, Bulgaria. INCOMA Ltd.

Alice Millour and Karën Fort. 2020. Text corpora and the challenge of newly written languages. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 111–120, Marseille, France. European Language Resources association.

Siyao Peng, Zihang Sun, Huangyan Shan, Marie Kolm, Verena Blaschke, Ekaterina Artemova, and Barbara Plank. 2024. Sebastian, Basti, Wastl?! recognizing named entities in Bavarian dialectal data. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14478–14493, Torino, Italia. ELRA and ICCL.

Alan Ramponi. 2024. Language varieties of Italy: Technology challenges and opportunities. *Transactions of the Association for Computational Linguistics*, 12:19–38.

Yves Scherrer. 2023. Character alignment methods for dialect-to-standard normalization. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 110–116, Toronto, Canada. Association for Computational Linguistics.

Larissa Schmidt, Lucy Linder, Sandra Djambazovska, Alexandros Lazaridis, Tanja Samardžić, and Claudiu Musat. 2020. A Swiss German dictionary: Variation in speech and writing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2720–2725, Marseille, France. European Language Resources Association.

Aarohi Srivastava and David Chiang. 2025. We're calling an intervention: Exploring fundamental hurdles in adapting language models to nonstandard text. In *Proceedings of the Tenth Workshop on Noisy and User-generated Text*, pages 45–56, Albuquerque, New Mexico, USA. Association for Computational Linguistics.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Rob van der Goot, Rik van Noord, and Gertjan van Noord. 2018. A taxonomy for in-depth evaluation of normalization for user generated content. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Hannes Wettig, Suvi Hiltunen, and Roman Yangarber. 2011. MDL-based models for alignment of etymological data. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 111–117, Hissar, Bulgaria. Association for Computational Linguistics.

Tatu Ylonen. 2022. Wiktextract: 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.

Marcos Zampieri, Preslav Nakov, and Yves Scherrer. 2020. Natural language processing for similar languages, varieties, and dialects: A survey. *Natural Language Engineering*, 26(6):595–612.

A Full Dataset Statistics

Table 7 shows number of instances for the datasets used in both word-level dialect tasks, which are introduced in Section 3.3. In addition to total number of instances, we also show number of instances for each part of speech.

B Full Results

Table 16 contains the detailed results for the task of judging whether a Bavarian word is a translation of a German lemma. Table 17 shows the detailed results for the translation task.

Part-of-speech	Judgment			Translation
	Yes	Inflected	No	
Noun	6,720	2,670	28,480	6,564
Adjective	1,358	3,066	5,496	1,325
Adverb	1,157	–	2,783	1,126
Verb	934	1,182	6,214	916
Proper Noun	574	86	34,430	556
Adposition	148	–	342	144
Numeral	45	–	175	45
Sub. Conjunction	35	–	185	34
Determiner	34	37	139	34
Auxiliary	8	17	75	8
Pronoun	8	9	153	8
Coord. Conjunction	15	–	65	7
Other (X)	5	3	3,012	5
Interjection	3	–	7	3
Particle	–	–	30	–
Total	11,044	7,070	81,586	10,775

Table 7: Number of test instances in each part-of-speech class for both tasks.

Predicted	Actual		
	yes	inflected	no
yes	6.82%	3.78%	0.80%
inflected	14.17%	15.28%	2.87%
no	78.95%	80.89%	96.00%
IF Error	0.06%	0.06%	0.33%

Table 8: Confusion matrix for `Llama3.1-8b`. F1 scores per class: 0.118 ('yes'), 0.179 ('infl.'), and 0.898 ('no').

Predicted	Actual		
	yes	inflected	no
yes	86.07%	77.57%	24.75%
inflected	0.16%	1.94%	0.51%
no	13.76%	20.50%	74.73%
IF Error	0.00%	0.00%	0.01%

Table 9: Conf. matrix for `Llama3.3-70b`. F1 scores per class: 0.411 ('yes'), 0.036 ('infl.'), and 0.838 ('no').

Predicted	Actual		
	yes	inflected	no
yes	5.94%	0.75%	0.42%
inflected	76.37%	82.89%	35.60%
no	2.36%	1.36%	13.78%
IF Error	15.33%	15.01%	50.20%

Table 10: Confusion matrix for `Llama4-17b`. F1 scores per class: 0.108 ('yes'), 0.233 ('infl.'), and 0.763 ('no').

Judgment task Confusion matrices for `Llama`, `Gemma` and `Aya` models are shown in Tables 8-14. We find that `Llama` and `Mistral` models show the same pattern: smaller LLMs are biased towards predicting 'no', while their larger versions can identify dialect variants ('yes', 'inflected') more accurately. `Llama4-17b` stands out with a much higher **IF Error** rate. `Gemma` and `Aya` models already show relatively strong results in their smaller variants. Detailed results are shown in Table 15.

Predicted	Actual		
	yes	inflected	no
yes	38.68%	37.75%	4.91%
inflected	0.32%	0.38%	0.03%
no	61.00%	61.87%	95.05%
IF Error	0.00%	0.00%	0.00%

Table 11: Conf. matrix for `Aya-expanse-8b`. F1 scores per class: 0.388 ('yes'), 0.008 ('infl.'), and 0.911 ('no').

Predicted	Actual		
	yes	inflected	no
yes	46.73%	34.10%	7.07%
inflected	7.43%	12.91%	2.14%
no	44.17%	51.60%	90.47%
IF Error	1.67%	1.39%	0.32%

Table 12: Conf. matr. for `Aya-expanse-32b`. F1 scores per class: 0.423 ('yes'), 0.173 ('infl.'), and 0.901 ('no').

Predicted	Actual		
	yes	inflected	no
yes	60.28%	48.60%	13.86%
inflected	6.90%	18.46%	3.94%
no	32.82%	32.94%	82.20%
IF Error	0.00%	0.00%	0.00%

Table 13: Confusion matrix for `Gemma3-12b`. F1 scores per class: 0.410 ('yes'), 0.211 ('infl.'), and 0.868 ('no').

Predicted	Actual		
	yes	inflected	no
yes	78.24%	49.02%	43.60%
inflected	11.05%	42.97%	8.99%
no	10.71%	8.01%	47.32%
IF Error	0.00%	0.00%	0.09%

Table 14: Confusion matrix for `Gemma3-27b`. F1 scores per class: 0.294 ('yes'), 0.326 ('infl.'), and 0.633 ('no').

C Prompt selection

We include the prompts we included in the prompt pool (Section 4.2) in Tables 18 and 19. For the test set, we only use the best prompt per model (highlighted rows). This prompt selection matters, as the LLMs we test are all sensitive to the way prompts are phrased: Table 20 illustrates this for the word pair judgments, and Table 21 for the word-level dialect translation task.

We additionally evaluate versions of the best prompts that include a usage example of the word in context (Tables 22 and 23) and German-language versions (Tables 24 and 25). The results for these modified prompts are in Table 26.

Model	'yes'			'inflected'			'no'		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Mistral-7b	0.157	0.008	0.015	0.136	0.071	0.093	0.823	0.963	0.888
Mistral-123b	0.445	0.720	0.550	0.440	0.160	0.234	0.931	0.875	0.902
Llama3.1-8b	0.450	0.068	0.118	0.217	0.153	0.179	0.844	0.96	0.898
Llama3.3-70b	0.270	0.861	0.411	0.240	0.019	0.036	0.954	0.747	0.838
Llama4-17b	0.623	0.059	0.108	0.135	0.829	0.233	0.944	0.640	0.763
Aya-expanse-8b	0.390	0.387	0.388	0.300	0.004	0.008	0.875	0.951	0.911
Aya-expanse-32b	0.387	0.467	0.423	0.263	0.129	0.173	0.894	0.908	0.901
Gemma3-12b	0.311	0.603	0.410	0.247	0.185	0.211	0.918	0.822	0.868
Gemma3-27b	0.181	0.782	0.294	0.262	0.430	0.326	0.957	0.473	0.633

Table 15: Precision, Recall and F1 scores of individual classes in our dialect judgment task. Instances where LLMs fail to follow instructions are considered the same as predicting 'no'.

	Baselines				Mistral		Llama			Aya-expanse		Gemma	
	Rand.	LD	Maj.	Log.Reg.	7b	123b	3.1-8b	3.3-70b	4-17b	8b	32b	12b	27b
Noun	0.268	0.355	0.286	0.351	0.292	**0.561**	0.398	0.444	0.354	0.447	0.500	0.492	0.453
Adjective	0.309	0.307	0.238	0.310	0.278	**0.488**	0.305	0.369	0.394	0.332	0.411	0.444	0.431
Adverb	0.257	0.331	0.276	0.270	0.270	**0.484**	0.300	0.379	0.237	0.392	0.442	0.441	0.288
Verb	0.280	0.307	0.285	0.344	0.335	**0.517**	0.351	0.340	0.338	0.390	0.469	0.430	0.323
Proper Noun	0.174	0.202	0.330	0.312	0.330	**0.473**	0.383	0.359	0.309	0.436	0.414	0.390	0.261
Adposition	0.262	0.264	0.274	0.257	0.277	**0.519**	0.284	0.422	0.261	0.421	0.469	0.489	0.285
Numeral	0.242	0.336	0.295	0.292	0.313	**0.487**	0.352	0.365	0.260	0.471	0.484	0.454	0.326
Sub. Conj.	0.225	0.142	0.305	0.305	0.310	**0.453**	0.301	0.249	0.234	0.417	0.445	0.409	0.159
Determiner	0.235	0.136	0.266	0.260	0.343	**0.479**	0.382	0.225	0.264	0.377	0.406	0.421	0.226
Auxiliary	0.356	0.076	0.286	0.322	0.350	**0.552**	0.473	0.175	0.398	0.463	0.506	0.475	0.226
Pronoun	0.187	0.091	0.316	0.296	0.310	0.409	0.412	0.219	0.263	0.365	0.387	**0.415**	0.170
Coord. Conj.	0.184	0.178	0.299	0.291	0.247	**0.437**	0.301	0.225	0.211	0.421	0.407	0.351	0.186
Other (X)	0.169	0.093	0.333	0.314	0.323	**0.356**	0.332	0.294	0.308	0.331	0.336	0.333	0.197
Interjection	0.300	0.154	0.275	0.083	0.061	0.250	0.275	0.061	0.083	**0.593**	0.330	0.275	0.154

Table 16: **Judging potential German–Bavarian translation candidates** – Macro F1 scores for each model across all 15 POS tags. In bold font we highlight the **highest score** for each POS.

	Mistral		Llama			Aya-expanse		Gemma	
	7b	123b	3.1-8b	3.3-70b	4-17b	8b	32b	12b	27b
Noun	0.085	0.552	0.371	0.596	**0.610**	0.403	0.524	0.516	0.609
Adjective	0.003	0.381	0.189	0.502	**0.560**	0.334	0.452	0.476	0.534
Adverb	0.000	0.374	0.154	0.498	0.514	0.340	0.428	0.429	**0.543**
Verb	0.000	0.425	0.105	0.505	0.527	0.377	0.440	0.397	**0.563**
Proper Noun	0.101	0.511	0.390	0.590	0.586	0.335	0.487	0.433	**0.594**
Adposition	0.000	0.403	0.160	0.479	0.472	0.326	0.479	0.354	**0.556**
Numeral	0.000	0.333	0.156	**0.689**	0.667	0.378	**0.689**	0.489	**0.689**
Sub. Conjunction	0.000	0.118	0.000	0.118	**0.294**	0.147	0.206	0.324	0.294
Determiner	0.000	0.206	0.088	0.265	**0.412**	0.294	0.324	0.235	0.471
Auxiliary	0.000	0.750	0.375	0.875	**1.000**	0.625	0.625	0.750	0.875
Pronoun	0.000	0.125	0.000	**0.250**	0.125	0.125	**0.250**	0.000	**0.250**
Coord. Conjunction	0.000	0.429	0.286	**0.857**	0.714	0.429	0.571	0.714	0.714
Other (X)	0.200	0.200	0.200	**0.400**	**0.400**	0.200	0.200	0.200	0.200
Interjection	0.000	0.000	0.000	0.333	0.000	0.000	0.000	0.333	**0.667**

Table 17: **Dialect-to-standard translation** – Accuracy scores for each model across all 15 POS tags. In bold font we highlight the **highest score** for each POS.

ID	Prompt template
0	`Classify the Bavarian term 'term_bar' in relation to the Standard German term 'term_de'. Return 'yes' if it is an exact dialectal translation, 'inflected' if it is an inflected dialectal translation of it, or 'no' if neither applies. Do not say any other word.`
1	`Evaluate the relationship between 'term_bar' (Bavarian) and 'term_de' (Standard German). Respond with 'yes' for a dialectal match, 'inflected' for a dialectal inflectional variant, and 'no' if neither. Do not say any other word.`
2	`Is the Bavarian term 'term_bar' an exact dialectal variant ('yes'), a dialectal morphological inflection ('inflected'), or not a dialectal variant ('no') of 'term_de' in Standard German? Return only "yes", "inflected", or "no".`
3	`Compare 'term_bar' (Bavarian) to 'term_de' (Standard German). Answer with 'yes' if it's a direct dialectal translation, 'inflected' if it's a dialectal inflected form, or 'no' if neither applies. Do not say any other word.`

4	`Categorize the Bavarian term 'term_bar' with respect to the German term 'term_de':`
	`- 'yes' = dialectal variant`
	`- 'inflected' = morphologically related`
	`- 'no' = otherwise`
	`Return only one label.`

5	`Task: Is the Bavarian term: "term_bar" a correct dialectal variant of the German term: "term_de"?`
	`Follow the given annotation guidelines.`
	`Guidelines:`
	`- yes: The candidate is an exact dialectal variation of the Standard German word.`
	`- inflected: The candidate is a morphologically inflected variant of the German word.`
	`- no: None of the two applies.`
	`Return only "yes", "inflected", or "no".`

6	`Classify the Bavarian term 'term_bar' with respect to the Standard German term 'term_de'. Return exactly one of the following:`
	`- 'yes' if it is an exact dialectal variant`
	`- 'inflected' if it is a morphologically inflected variant`
	`- 'no' otherwise`

7	`Task: Label the relationship between term_bar (Bavarian) and term_de (German) using these rules:`
	`- 'yes': direct dialectal equivalent`
	`- 'inflected': same lemma, different form`
	`- 'no': if neither applies`

8	`Is the Bavarian term 'term_bar':`
	`- a dialectal translation of the German term 'term_de' → 'yes'`
	`- an inflection of the German term 'term_de' → 'inflected'`
	`- not a variant or inflected variant of the German term 'term_de' → 'no'`
	`Answer only with 'yes', 'inflected', or 'no'.`

9	`Compare the Bavarian word 'term_bar' against the Standard German 'term_de'.`
	`Select only one label:`
	`- 'yes' for dialectal variant`
	`- 'inflected' for inflectional variant`
	`- 'no' otherwise`

10	`Task: Is the Bavarian term: "term_bar" a correct dialectal variant of the German term: "term_de"?`
	`Follow the given annotation guidelines.`
	`Guidelines:`
	`- yes: The candidate is an exact dialectal variant of the Standard German word.`
	`- inflected: The candidate is an inflected dialectal variant of the German word.`
	`- no: None of the two applies.`
	`Return only "yes", "inflected", or "no".`

11	`Is the Bavarian term 'term_bar':`
	`- a dialectal translation of the German term 'term_de' → 'yes'`
	`- an inflected dialectal translation of the German term 'term_de' → 'inflected'`
	`- not an (inflected or uninflected) dialectal translation of the German term 'term_de' → 'no'`
	`Answer only with 'yes', 'inflected', or 'no'.`

12	`Classify the Bavarian term 'term_bar' with respect to the Standard German term 'term_de'. Return exactly one of the following:`
	`- 'yes' if it is an exact dialectal variant`
	`- 'inflected' if it is a morphologically inflected dialectal variant`
	`- 'no' otherwise`

Table 18: **Translation candidate judgment** – Best prompts : `Mistral-7b` (2), `Mistral-123b` (8), `Llama3.1-8b` (5), `Llama3.3-70b` (5), `Llama4-17b` (8), `Aya-8b` (3), `Aya-32b` (5), `Gemma3-12b` (8), `Gemma3-27b` (6).

ID	Prompt template
0	`Translate 'term_bar' from Bavarian to German.`
1	`Translate 'term_bar' from Bavarian to German. Do not say any other word.`
2	`What is the German translation of the Bavarian term 'term_bar'?`
3	`What is the German translation of the Bavarian term 'term_bar'? Only output the translated word.`
4	`What is the translation of the word 'term_bar' into German?`
5	`What is the translation of the word 'term_bar' into German? Only output the translated word.`
6	`Translate the following word to German 'term_bar'.`
7	`Translate the following word to German 'term_bar'. Do not say any other word.`
8	`This is a Bavarian term: 'term_bar'. Only return the normalized Standard German word.`
9	`Translate the Bavarian term: 'term_bar' into High German form.`
10	`Translate the Bavarian term: 'term_bar' into High German form. Only return the High German form.`
11	`Translate the Bavarian term 'term_bar' into High German form.`
12	`Translate the Bavarian term 'term_bar' into High German form. Only return the High German form.`
13	`Translate the Bavarian term: 'term_bar' into Standard German form.`
14	`Translate the Bavarian term: 'term_bar' into Standard German form. Only return the Standard German form.`
15	`This is a Bavarian term: 'term_bar'. Only return the normalized Standard German equivalent. Respond with exactly one single word.`
16	`Translate the word 'term_bar' from Bavarian to German.`
17	`Translate the word 'term_bar' from Bavarian to German. Do not say any other word.`
18	`How do you say the Bavarian word 'term_bar' in German?`
19	`How do you say the Bavarian word 'term_bar' in German? Answer only with the translation.`
20	`Perform translation: convert the Bavarian form 'term_bar' into its Standard German equivalent. Return only the Standard German form, with no additional explanation or formatting.`

Table 19: **Dialect-to-standard translation** – Best prompts : `Mistral-7b` (20), `Mistral-123b` (20), `Llama3.1-8b` (20), `Llama3.3-70b` (20), `Llama4-17b` (20), `Aya-8b` (7), `Aya-32b` (20), `Gemma3-12b` (12), `Gemma3-27b` (20).

Model	Macro F1	IF Error
`Mistral-7b`	0.249 ± 0.102	0.486 ± 0.492
`Mistral-123b`	0.494 ± 0.102	0.362 ± 0.378
`Llama3.1-8b`	0.275 ± 0.104	0.233 ± 0.409
`Llama3.3-70b`	0.406 ± 0.118	0.136 ± 0.292
`Llama4-17b`	0.258 ± 0.084	0.670 ± 0.451
`Aya-expanse-8b`	0.349 ± 0.081	0.341 ± 0.472
`Aya-expanse-32b`	0.337 ± 0.151	0.354 ± 0.468
`Gemma3-12b`	0.430 ± 0.099	0.155 ± 0.317
`Gemma3-27b`	0.342 ± 0.138	0.107 ± 0.281

Table 20: **Translation candidate judgment** – Effect of prompt variation across 13 prompts on Macro F1 and IF Error ($\pm$ standard deviation) across models, evaluated on the development set (300 items).

Model	Accuracy	IF Error
`Mistral-7b`	0.005 ± 0.016	0.930 ± 0.199
`Mistral-123b`	0.109 ± 0.128	0.632 ± 0.421
`Llama3.1-8b`	0.105 ± 0.104	0.498 ± 0.489
`Llama3.3-70b`	0.210 ± 0.193	0.461 ± 0.479
`Llama4-17b`	0.217 ± 0.211	0.528 ± 0.452
`Aya-expanse-8b`	0.148 ± 0.148	0.548 ± 0.447
`Aya-expanse-32b`	0.184 ± 0.173	0.528 ± 0.449
`Gemma3-12b`	0.193 ± 0.186	0.457 ± 0.488
`Gemma3-27b`	0.190 ± 0.193	0.455 ± 0.487

Table 21: **Dialect-to-standard translation** – Effect of prompt variation across 21 prompts on Accuracy and IF Error ($\pm$ standard deviation) across models, evaluated on the development set (300 items).

ID	Prompt template

2	`Is the Bavarian term 'term_bar' an exact dialectal variant ('yes'), a dialectal morphological inflection ('inflected'), or not a dialectal variant ('no') of 'term_de' in Standard German?`
	
	`Usage example: "####"`
	
	`Return only "yes", "inflected", or "no".`

3	`Compare 'term_bar' (Bavarian) to 'term_de' (Standard German).`
	
	`Usage example: "####"`
	
	`Answer with 'yes' if it's a direct dialectal translation, 'inflected' if it's a dialectal inflected form, or 'no' if neither applies. Do not say any other word.`

5	`Task: Is the Bavarian term: "term_bar" a correct dialectal variant of the German term: "term_de"?`
	
	`Usage example: "####"`
	
	`Follow the given annotation guidelines.`
	`Guidelines:`
	`- yes: The candidate is an exact dialectal variation of the Standard German word.`
	`- inflected: The candidate is a morphologically inflected variant of the German word.`
	`- no: None of the two applies.`
	`Return only "yes", "inflected", or "no".`

6	`Classify the Bavarian term 'term_bar' with respect to the Standard German term 'term_de'.`
	
	`Usage example: "####"`
	
	`Return exactly one of the following:`
	`- 'yes' if it is an exact dialectal variant`
	`- 'inflected' if it is a morphologically inflected variant`
	`- 'no' otherwise`

8	`Is the Bavarian term 'term_bar':`
	`- a dialectal translation of the German term 'term_de' → 'yes'`
	`- an inflection of the German term 'term_de' → 'inflected'`
	`- not a variant or inflected variant of the German term 'term_de' → 'no'`
	
	`Usage example: "####"`
	
	`Answer only with 'yes', 'inflected', or 'no'.`

Table 22: **Translation candidate judgment (with context)**. We extend for each model the best-performing prompt by including a usage example.

ID	Prompt template
7	`Translate the following word to German 'term_bar'. Usage example: "####". Do not say any other word.`
12	`Translate the Bavarian term 'term_bar' into High German form. Usage example: "####". Only return the High German form.`
20	`Perform translation: convert the Bavarian form 'term_bar' into its Standard German equivalent. Usage example: "####". Return only the Standard German form, with no additional explanation or formatting.`

Table 23: **Dialect-to-standard translation (with context)**. We extend for each model the best-performing prompt by including a usage example.

ID	Prompt template
2	`Ist der bairische Begriff 'term_bar' eine exakte dialektale Variante ('yes'), eine morphologische Beugung ('inflected') oder keine Dialektvariante ('no') von 'term_de' im Hochdeutschen?`
3	`Vergleiche 'term_bar' (bairisch) mit 'term_de' (hochdeutsch). Antworten mit 'yes', wenn es sich um eine direkte dialektale Übersetzung handelt, mit 'inflected', wenn es eine gebeugte Form handelt, oder mit 'no', wenn beides nicht zutrifft.`

5	`Aufgabe: Ist der bairische Begriff "term_bar" eine korrekte dialektale Variante des hochdeutschen Begriffs "term_de"? Befolge die untenstehenden Annotationsrichtlinien.`
	`Richtlinien:`
	`- yes: Der Begriff ist eine exakte dialektale Entsprechung.`
	`- inflected: Der Begriff ist eine morphologisch gebeugte Variante.`
	`- no: Keines von beiden trifft zu.`
	`Gib nur "yes", "inflected" oder "no" zurück.`

6	`Klassifiziere den bairischen Begriff 'term_bar' im Verhältnis zum hochdeutschen Begriff 'term_de'. Gib genau eines der folgenden Labels zurück:`
	`- 'yes', wenn es eine exakte dialektale Variante ist`
	`- 'inflected', wenn es eine morphologisch gebeugte Form ist`
	`- 'no', andernfalls`

8	`Ist der bairische Begriff 'term_bar':`
	`- eine Dialektübersetzung des deutschen Begriffs 'term_de' → 'yes'`
	`- eine Beugungsform des Deutschen Begriffs 'term_de' → 'inflected'`
	`- keine Variante oder flektierte Variante des deutschen Begriffs 'term_de' → 'no'`
	`Antworte nur mit 'yes', 'inflected', oder 'no'.`

Table 24: **Translation candidate judgment (German prompts)**. We extend for each model the best-performing prompt by including a usage example.

ID	Prompt template
7	`Übersetze das folgende Wort ins Deutsche 'term_bar'. Gib nur das übersetzte Wort aus.`
12	`Übersetze den bairischen Begriff 'term_bar' ins Hochdeutsche. Gib nur die hochdeutsche Form zurück.`
20	`Übersetzung durchführen: konvertiere die bayerische Form 'term_bar' in das hochdeutsche Äquivalent. Gib nur die hochdeutsche Form zurück, ohne weitere Erklärungen oder Formatierungen.`

Table 25: **Dialect-to-standard translation (German prompts)**. We extend for each model the best-performing prompt by including a usage example.

	Context				**Language**			
	Judgment		**Translation**		**Judgment**		**Translation**	
Model	**Macro-F1**	**IF Error**	**Accuracy**	**IF Error**	**Macro-F1**	**IF Error**	**Accuracy**	**IF Error**
`Mistral-7b`	0.004	0.817	-0.034	0.128	-0.032	0.999	0.000	-0.103
`Mistral-123b`	-0.016	0.366	0.183	-0.005	-0.174	0.351	0.023	0.011
`Llama3.1-8b`	0.028	-0.003	0.147	0.047	-0.028	0.022	0.012	0.000
`Llama3.3-70b`	0.136	0.000	0.130	0.032	0.122	0.000	-0.015	0.002
`Llama4-17b`	-0.272	-0.433	0.110	0.033	-0.255	-0.437	-0.021	0.001
`Aya-expanse-8b`	-0.198	0.000	-0.061	0.323	-0.136	1.000	-0.038	0.001
`Aya-expanse-32b`	-0.077	0.564	-0.009	0.208	-0.063	-0.004	0.006	0.001
`Gemma3-12b`	0.021	0.000	0.071	0.064	-0.058	0.000	-0.097	0.011
`Gemma3-27b`	-0.036	-0.001	0.095	0.046	-0.184	-0.001	-0.023	-0.001

Table 26: Overall changes (Δ) in model performance and instruction-following abilities (IF Error) for both tasks. On the left, we report the effect of **context** (context – no context), and on the right, the effect of **prompt language** (German – English).