Post-correction of Historical Text Transcripts with Large Language Models: An Exploratory Study

Emanuela Boros[1], **Maud Ehrmann**[1],
Matteo Romanello[2], **Sven Najem-Meyer**[1], **Frédéric Kaplan**[1]

[1]Digital Humanities Laboratory, EPFL, Lausanne, Switzerland
[2]Institute of Archeology and Classical Studies, University of Lausanne, Switzerland
{emanuela.boros,maud.ehrmann,sven.najem-meyer,frederic.kaplan}@epfl.ch
matteo.romanello@unil.ch

Abstract

The quality of automatic transcription of heritage documents, whether from printed, manuscripts or audio sources, has a decisive impact on the ability to search and process historical texts. Although significant progress has been made in text recognition (OCR, HTR, ASR), textual materials derived from library and archive collections remain largely erroneous and noisy. Effective post-transcription correction methods are therefore necessary and have been intensively researched for many years. As large language models (LLMs) have recently shown exceptional performances in a variety of text-related tasks, we investigate their ability to amend poor historical transcriptions. We evaluate fourteen foundation language models against various post-correction benchmarks comprising different languages, time periods and document types, as well as different transcription quality and origins. We compare the performance of different model sizes and different prompts of increasing complexity in zero and few-shot settings. Our evaluation shows that LLMs are anything but efficient at this task. Quantitative and qualitative analyses of results allow us to share valuable insights for future work on post-correcting historical texts with LLMs.

1 Introduction

Over the last two decades, heritage institutions have digitised their collections on a massive scale, yielding millions of historical documents in digital format along with their machine-readable text (Terras, 2011; Padilla, 2019). Whether obtained through optical character recognition (OCR, for printed documents), handwritten text recognition (HTR, for manuscripts) or automatic speech recognition techniques (ASR, for audio documents), the availability of textual transcriptions has not only improved the accessibility of historical documents, but has also opened up the possibility of applying machine-reading techniques to their content. Increasingly, research and initiatives are being undertaken to process and mine the rich information contained in unstructured heritage text data (Clausner et al., 2019; Ehrmann et al., 2022), or to develop computational approaches to its analysis (McGillivray et al., 2020; Bunout et al., 2022). While highly promising, these efforts face a persistent challenge that considerably impacts their effectiveness: the suboptimal quality of transcriptions.

Most text data derived from digitised historical documents contains transcription errors, for two reasons. Firstly, despite recent advances in the accuracy of text recognition –thanks to the adoption of neural approaches and robust transcription frameworks (Reul et al., 2019; Kahle et al., 2017; Engl, 2020) – the quality of digital documents and the diversity of document types, languages, scripts, fonts and handwriting still poses significant challenges to OCR, HTR and ASR approaches. Secondly, even though the latest transcription engines are much better than their predecessors, collections digitised long ago are rarely reprocessed, often for budgetary reasons. The impact of noisy transcriptions on downstream processes is well-documented, with a detrimental effect on search capacities (Chiron et al., 2017b), named entity processing (Linhares Pontes et al., 2019; Hamdi et al., 2020), language modeling (Todorov and Colavizza, 2022), and most natural language processing (NLP) tasks (van Strien et al., 2020).

Possible answers to this situation lie in the development of post-correction methods aimed at producing a better, corrected version of a transcription with respect to the corresponding original text and, more recently, in the targeted re-transcription of identified faulty text sections—a complex process that requires robust pipelines (Schneider and Maurer, 2022). Latest approaches to post-correction use sequence-to-sequence neural networks, with character-based translation models (Amrhein and Clematide, 2018) as well as LSTMs or transformer-

Proceedings of LaTeCH-CLfL 2024, pages 133–159

based models (Nguyen et al., 2020; Rigaud et al., 2019). Despite significant progress, performances vary greatly across the variety of historical texts (Chiron et al., 2017a; Rigaud et al., 2019), and systems still have difficulties dealing with extreme noise (Amrhein and Clematide, 2018), generalising (Todorov and Colavizza, 2020) and avoiding undesired changes to error-free text, a particularly important requirement with historical sources (Schaefer and Neudecker, 2020).

At the core of many approaches to NLP, language representations have evolved from auxiliaries to machine learning systems, such as n-gram models and word vectors, to specialised experts fine-tuned for specific tasks, such as transformer-based pre-trained language models. Current efforts are aimed at more versatile systems, training autoregressive generative models on ever larger amounts of data and model sizes. This results in the emergence of large language models (LLMs) with exceptional robustness and generalisability capabilities, even in zero and few-shot settings (Brown et al., 2020; Chowdhery et al., 2022; Scao et al., 2022; Zhang et al., 2022, to cite but a few). Since the launch of the GPT (Generative Pre-Trained) model series by OpenAI in 2018, and in particular the ChatGPT conversational robot released in November 2022 based on the GPT-3.5 or GPT-4 models, the LLM race is on. More and more models are being released, with impressive performance on various text-related tasks and results close to the state of the art, e.g. in question answering (Bang et al., 2023), machine translation (Jiao et al., 2023) and stance detection (Zhang et al., 2023).

Of all the ways in which ChatGPT can be used, its ability to revise and improve the quality of texts was quickly noticed. Having been trained on hundreds of billions of tokens to predict the next word in a sequence, such text editing ability is hardly surprising. While evaluations have been carried out on tasks involving language editing capacities such as text summarisation and grammatical error correction—with fairly good results for some metrics (Gao et al., 2023; Wu et al., 2023; Laskar et al., 2023)—the ability of LLMs to amend historical transcriptions has not yet been systematically studied. Can large language models that have powerful capabilities for generating and understanding language in different contexts also help to correctly rewrite texts that have been poorly transcribed by automated processes? Capitalising on the recent rise of generative LLMs, this study aims to investigate, beyond anecdotal observations, the effectiveness of large language models to post-correct historical transcripts. In particular, we want to gain a better understanding of whether, to what extent, and under what conditions LLMs can produce good corrections of historical transcriptions, and seek to answer the following questions:

– **Ability to correct.** When prompted to correct the transcription of a given historical document, do LLMs improve, degrade, or leave the input text intact?

– **Sensitivity to variations in input text and instructions.** Does the correction performance depend on the time period, language, type and noise of the original document? How sensitive is LLM-based post-correction to prompt instructions?

– **Real-world applicability.** How do open-access models compare to OpenAI GPT models? Could LLM-based post-correction be a valid and cost-effective option for efficiently correcting backlogs of millions of noisy historical documents?

To this end, we evaluate fourteen foundation language models from four model series against various post-correction benchmarks comprising different languages, time periods, document types, and different transcription quality and origins. We compare the performance of different prompts of increasing complexity in zero and few-shot settings and provide quantitative and qualitative analyses of the results.

2 Background

We briefly highlight some key facets of LLMs and refer to Zhao et al. (2023) for a detailed survey. LLMs are text generators that are trained on massive plain text data. Based on well-established technology – deep neural networks and self-supervised learning – their success is mainly due to two key factors: scaling up model size and the amount of training data (Zhao et al., 2023). The former was made possible by the Transformer architecture (Vaswani et al., 2017) and the latter by 20 years' worth of internet text data. Coupled with optimisation frameworks (Rasley et al., 2020; Shoeybi et al., 2020), this has led to the rapid emergence of large language models with hundreds of billions of parameters, with increasing capacity to

learn, generalise and act as general-purpose task solvers as more (clean) data is input. Quickly identified as a paradigm shift in artificial intelligence systems, LLMs, and more generally foundation models, possess the crucial ability to learn in context, i.e. to perform a task with only a few instructions and task demonstrations, generating the expected output without updating model parameters (Dong et al., 2023). Their behaviour is implicitly deduced and not explicitly constructed. Following OpenAI's initial releases, numerous models are being published by various commercial and academic players, sparking a global debate on the opportunities and risks from a scientific and societal point of view (Bommasani et al., 2022).

LLMs's text editing capabilities have particularly been assessed in the context of grammatical error correction (GEC). Several studies have compared state-of-the-art GEC systems with GPT models against various benchmarks and found that they can perform GEC tasks effectively (especially on sentence input) and that they are slightly better at error detection than at correction due to over-correction (Ostling and Kurfalı, 2022; Wu et al., 2023), a tendency that can be controlled by optimising prompt and example selection (Fang et al., 2023; Loem et al., 2023; Coyne et al., 2023). Few attempts have been made to post-correct transcriptions with LLMs, including LLM-based selection of the best post-corrections for contemporary documents (Gupta et al., 2021), and fine-tuning the BART autoregressive Transformer to post-correct historical newspapers (Soper et al., 2021). To our knowledge, no previous study has explored LLM-based post-correction of historical transcripts.

3 Approach

We aim to assess the ability of generative LLMs to correct machine transcriptions of historical documents and to provide insights into what works best. Using post-correction transcription benchmarks, our approach is, essentially, to compare the similarities between original automated transcriptions and their ground truth (GT), and between LLM-corrected versions of the original transcriptions and the same GT, and to observe their variation, i.e. whether the LLM-corrected version is a degradation or an improvement of the original transcriptions.

To better reflect the diversity of archival collections and obtain results that are, to some extent, generalisable, we consider various post-correction benchmarks covering different document types, languages and periods, as well as different transcription qualities and origins. Their different formats are standardised to ensure consistent handling of the data, especially as far as different levels of text segmentation (line, sentence region) are concerned. The selection of LLMs is based on model size, training data, resource requirements and accessibility. We consider fourteen models from four series and craft five prompts to guide their text generation. Various post-processing heuristics are applied to clean up the output.

Despite their astonishing performance, we can hypothesise that post-correcting historical transcripts may pose several challenges for LLMs. First, unlike typical text generation tasks where multiple answers can be valid (summarisation, translation, dialogue systems), transcription correction requires a single correct answer that exactly matches the GT. This, of course, runs counter to LLM's tendency to hallucinate. Second, LLMs are not specifically trained for post-correction but learn in context through natural language instructions that they must understand – an ability that is unequal between models. Third, the level and nature of noise in historical transcripts vary considerably from document to document, i.e. texts can range from minimally to extremely noisy, with different forms of noise. While this challenge affects all approaches, LLMs may not have encountered many noisy historical transcripts during their training.

3.1 Datasets

We use eight post-correction benchmarks, each consisting of two historical transcriptions: one from an automated system (to be corrected) and its ground truth counterpart. The two versions are aligned at different levels, and there are no images of the original documents. Besides diversity, the choice of benchmarks was guided by the requirement that transcripts should not be too short to provide sufficient context for the LLMs. Table 1 outlines the datasets (six from OCR, one from ASR, and one from HTR), which are also presented below.

`icdar-2017` & `icdar-2019` Two ICDAR evaluation campaigns on post-OCR text correction published two benchmarks in 2017 and 2019 (Chiron et al., 2017a; Rigaud et al., 2019). `icdar-2017` (12M characters) comprises monographs and newspapers in English and French originating from a

Dataset Alias	Document Type	Origin	Time Period	Language	# Lines*	# Sentences*	# Regions*
icdar-2017	newspapers,monographies	OCR	17C-20C	en, fr	0	461	28
icdar-2019		OCR	not specified	bg,cz,en,fr,de,pl,sl	0	404	41
overproof	newspaper	OCR	19-20C	en	2,278	399	41
impresso-nzz	newspaper	OCR	18-20C	de	1,256	577	203
ajmc-mixed	class. commentaries	OCR	19C	grc, de, en, fr	535	379	33
ajmc-primary	class. commentaries	OCR	19C	grc, de, en, fr	40	27	9
htrec	papyri and manuscripts	HTR	10C-16C	grc	180	8	8
ina	radio programs	ASR	20C	fr	201	290	6

Table 1: Overview of the datasets. (*): Figures correspond to the data used in this study, except for htrec, ina, ajmc-mixed, and ajmc-primary, where they correspond to the full dataset.

range of heritage institutions and initiatives (Papadopoulos et al., 2013; Neudecker and Antonacopoulos, 2016), and icdar-2019 (22M characters) expands to further types of printed documents, newspapers and shopping receipts in 10 European languages. No document dates are specified, but we estimate a 17C-20C time frame for icdar-2017 based on the original datasets, assuming a similar range for the second. The documents in the dataset correspond to different segments of historical records, with OCR transcriptions and GT aligned at character level. Detailed information about data quality is unavailable, yet documents may contain up to 50% of misrecognised characters. In this study, we use samples of 12% and 20% of the 2017 and 2019 data, respectively.

overproof Published as part of an OCR evaluation, the Overproof benchmarks were extracted from the National Library of Australia's Trove digitised newspaper collection (Evershed and Fitch, 2014)[1]. In this study, we use a 20% random sample from the first dataset, which consists of medium-size articles from the *Sydney Morning Herald* from 1842 to 1945. The documents in the dataset correspond to articles, with OCR transcriptions and GT aligned at line level. The ground truth was crowdsourced from users of the Trove website and may therefore be incomplete. We used the code of van Strien et al. (2020) to pre-process the data.

impresso-nzz Created as part of the first *impresso* project (Ehrmann et al., 2020), the impresso-nzz dataset consists of 167 front pages from the *Neue Zürcher Zeitung* newspaper, randomly selected from each year between 1780 and 1947 (Ströbel and Clematide, 2019)[2]. Documents correspond to pages, with OCR and GT aligned at region, line, and word levels. In this study, we use a 50% random sample of the data in its OCRed version from ABBYY FineReader Server11, that showed a low recognition rate on this black letter font (Ströbel et al., 2020).

ajmc This dataset was created as part of the Ajax Multi-Commentary project and consists of five 19C scholarly commentaries on Sophocles' *Ajax*. Commentaries are written in German, English, and Latin and contain a mix of Latin and polytonic Greek scripts (Romanello et al., 2021). Documents correspond to commentary pages, transcribed using Tesseract's de, en, la and grc models. OCR and GT are aligned at region and line level. In this study, we use two subsets: ajmc-primary with texts written only in Greek, and ajmc-mixed with mixed languages and scripts.

htrec Compiled for the Handwritten Text Recognition Error Correction (HTREC) shared task (Pavlopoulos et al., 2023), the htrec dataset comprises Byzantine papyri and manuscripts from 10C-16C in Byzantine Greek (between ancient and modern Greek) (Platanou et al., 2022). Documents correspond to pages, with HTR and GT transcriptions aligned at line level. In this study, we use the test set consisting of 180 lines.

ina Finally, the ina dataset consists of six French radio programmes of different types (political speech, news, fiction, entertainment), each from one decade between 1930 and 1980. Audio and ASR transcriptions were provided by the French National Audiovisual Institute to the authors, who transcribed them manually. Documents correspond to a programme, with ASR and GT aligned at the level of text 'sections'. These sections do not correspond to a speaker turn or anything else, and may contain less or more than one sentence. Background events (e.g. music) are not indicated.

[1] https://overproof.projectcomputing.com/evaluation

[2] Refer to the Zenodo and GitHub repositories.

Model	Release date	Used sizes	Access	Max length
GPT-2	11.2019	1.5B	open	1,024
GPT-3	06.2020	175B	limited	2,049
GPT-3.5	03.2023	unknown	limited	4,096
GPT-4	03.2023	unknown	limited	8,192
BLOOM	07.2022	560M, 3B, 7.1B	open	2,048
BLOOMZ	11.2022	560M, 3B, 7.1B	open	2,048
OPT	05.2022	350M, 6.7B	open	2,048
LLaMA	02.2023	7B	open	2,048
LLaMA-2	07.2023	7B	open	4,096

Table 2: Overview of LLMs used in this study.

Overall, these datasets provide challenging material for LLMs. In addition to the variety of error types and languages, models have to deal with a wide range of document lengths, some of which are exceptionally long, as well as with truncated text regions due to incorrect segmentation. We have not evaluated the ground truths of these benchmarks and assume that they are acceptable since they were created for evaluation purposes. It should be noted, however, that their quality is certainly not perfect.

3.2 Models

We consider fourteen LLMs from four model series, which differ in size, training settings, data, and accessibility. All models, summarised in Table 2, are decoder-only autoregressive LLMs.

GPT OpenAI's GPT model series consists of powerful models that grow in capability as training data and model size increase. In this study, we use GPT-2, GPT-3, GPT-3.5, and GPT-4. Only GPT-2 (Radford et al., 2019) is freely available to everyone, the others are accessible via OpenAI's commercial API and their training conditions and features have not been fully disclosed. While GPT-3 proved the impact of scaling and demonstrates in-context learning ability (Brown et al., 2020), the next capacity improvements came from training on code and reinforcement learning through human feedback (GPT-3.5), as well as increased maximum context length (Ouyang et al., 2022). GPT-4 has an even larger context window, and multimodal input (OpenAI, 2023).

BLOOM(Z) The BigScience Large Open-science Open-access Multilingual language model, developed by the BigScience project, handles 46 languages, is open source and, at the time of release, was larger than GPT-3 (176B). The initiative produced models of different sizes trained on the same dataset. Aimed at improving generalisation, the BLOOMZ series was subsequently released, with BLOOM and mT5 models fine-tuned on cross-lingual variants of the P3 dataset (a collection of prompts covering various NLP tasks) (Scao et al., 2022; Muennighoff et al., 2023). In this study, we use BLOOM(Z) 560M, 3B and 7.1B.

OPT The Open Pre-trained Transformers is a series of open-source LLMs developed by Meta AI. Trained on English data and ranging from 125M to 175B parameters, the OPT models are large causal language models designed to be comparable in size and performance to GPT-3, but transparent and with a lower training carbon footprint (Zhang et al., 2022). We use two model sizes.

LLaMA Also released openly and aimed at the research community, the Large Language Model Meta AI (LLaMA) model has been trained on twenty languages and, according to its developers, outperforms GPT-3 on many tasks while using fewer resources (Touvron et al., 2023a). LLaMA-2 is trained on 40% more data and with twice the context length (Touvron et al., 2023b).

4 Experimental Setting

4.1 Data Preparation

Data preparation consists of two processes: the homogenisation of text structures and their formats, and the definition of OCR quality bands.

Historical documents have different layouts (single or multiple columns, text subdivisions, presence of images), as do the selected datasets, with documents corresponding to different elements (page, article) and transcriptions corresponding to different levels of text segmentation (line, article, region). To ensure consistent data handling, facilitate fair performance comparison across datasets, and study the importance of context in post-correction with generative LLMs, we define three levels of text units. First, a line level – commonly found in historical documents – is already present in all datasets except `icdar`. Second, a sentence level, a linguistically meaningful unit of text that is not present in any of the datasets. For sentence splitting, we first align transcription and GT tokens using a fast recursive text alignment scheme (Yalniz and Manmatha, 2011), before applying a sentence splitter (Sadvilkar and Neumann, 2020). This process is applied to all datasets. Finally, we

Basic-1	`Correct the text:\n {{TEXT}}`
Basic-2	`Correct the spelling and grammar of the following text:\n {{TEXT}}`
Complex-1	`Correct the spelling and grammar of the following incorrect text from on optical character recognition (OCR) applied to a historical document:\n Incorrect text: {{TEXT}}\n The corrected text is:`
Complex-2	`Please assist with reviewing and correcting errors in texts produced by automatic transcription (OCR) of historical documents. Your task is to carefully examine the following text and correct any mistakes introduced by the OCR software. The text to correct appears after the segment "TEXT TO CORRECT:". Please place the corrected version of the text after the "CORRECTED TEXT:" segment. Do not write anything else than the corrected text.\n\n TEXT TO CORRECT:\n {{TEXT}} \n CORRECTED TEXT:`
Complex-3	Complex-2 translated to fr, de, etc.

Table 3: Prompt templates.

consider a region level, which corresponds to the whole text of a document in a dataset.

We further qualify each text unit according to its (original) transcription quality, expressed by the Levenshtein similarity measure between the transcription and the ground truth (the measure is presented in Section 4.3). Transcriptions are classified into one of five percentage quality bands: $0-40$, $40-60$, $60-80$, $80-99$ and $99-100$; the higher, the better.

4.2 Prompt Templates and Setup

Guiding models toward the intended output relies on prompts (Liu et al., 2023). Given a fixed LLM, prompting involves converting each test input into a prompt based on a template and inputting it into the model to generate the response.

We manually design five prompt templates that provide small to strong guidance, presented in Table 3. `Basic-1` simply instructs the model to correct any errors present in the input text. `Basic-2` is a bit more explicit and tells the model to focus on spelling and grammar errors. This prompt may be useful for text editing in general, but may still be too imprecise for OCR, ASR, and HTR material. `Complex-1` informs the model that the input is from an automatic transcription of a historical document (OCR, ASR or HTR), and `Complex-2` additionally asks it to shape its response according to an explicit format. Such context awareness and format guidance may improve the quality of corrections and the cleanliness of the output. Finally, `Complex-3` translates `Complex-2` in all languages of the datasets.

Models are prompted in zero-shot (ZS) and few-shot (FS) settings. In ZS, the model has access to the test input only, whereas in FS, three demonstration examples are provided. The examples are randomly selected from three of the lowest transcription quality bands for each dataset. For all experiments, we perform a single-pass generation, i.e. without aggregation of multiple runs.

LLM output often do not match the expected response shape and require post-processing. Where necessary, we trim the output from unnecessary spaces or response presentation formulas, remove repeated (parts of) the prompt, and discard any text that is longer than 1.5 times the input. Post-processing is further described in Appendix B.

4.3 Evaluation Metric

We determine the difference in quality between an LLM-generated post-correction of a transcription and the original automatic transcription using a Post-Correction Improvement Score (PCIS)[3]. This relative score measures the positive or negative improvement, in terms of Levenshtein similarity, between the two transcriptions and the same ground truth. The Levenshtein similarity (*lev_sim*) is based on the Levenshtein distance between a machine transcription (`transcr`) and a ground truth (`GT`), and is computed as follows:

$$lev\_sim = \frac{length - lev\_dist(\texttt{transc}, GT)}{length} \quad (1)$$

where *lev_dist* is a string metric that measures the difference between two textual sequences based on the number of single-character edits (Levenshtein, 1966) and *length* is the length of the longer string (*max(len(*`transc`*), len(*`GT`*))*). The Levenshtein similarity provides a measure between 0 and 1, with higher values indicating higher similarity.

The PCIS is calculated based on the Levenshtein similarity between an original transcription and the GT (*orig_sim*), and between the LLM-generated post-correction and the GT (*llm_sim*), as follows:

$$\text{PCIS} = \begin{cases} \min(\max(llm\_sim, -1), 1), \\ \quad if\ orig\_sim = 0 \\ \min(\max(\frac{llm\_sim - orig\_sim}{orig\_sim}, -1), 1), \\ \quad if\ orig\_sim \neq llm\_sim \\ 0, \quad if\ orig\_sim = llm\_sim. \end{cases} \quad (2)$$

The improvement score ranges from -1 to 1: negative values indicate deterioration, positive values indicate improvement, and 0 indicates no change.

[3]Please note that this measure is not our invention but a classical way to calculate the relative change or difference from an initial value to a new value.

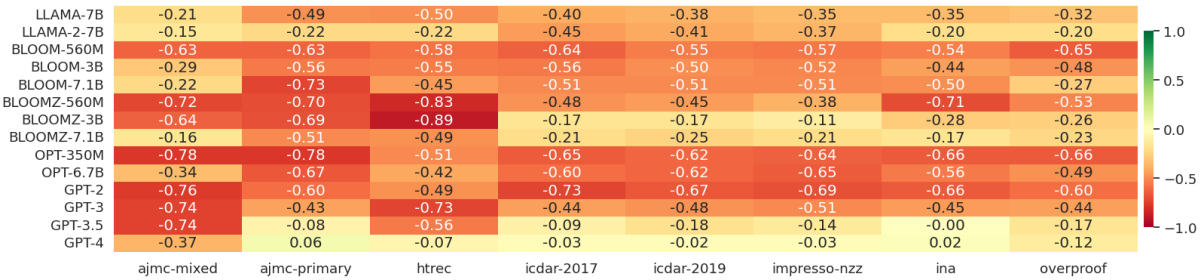

Figure 1: Average of post-correction improvement score per model and dataset, based on post-processed responses to sentence-level input with the best prompt template `Complex-2` in the zero-shot setting.

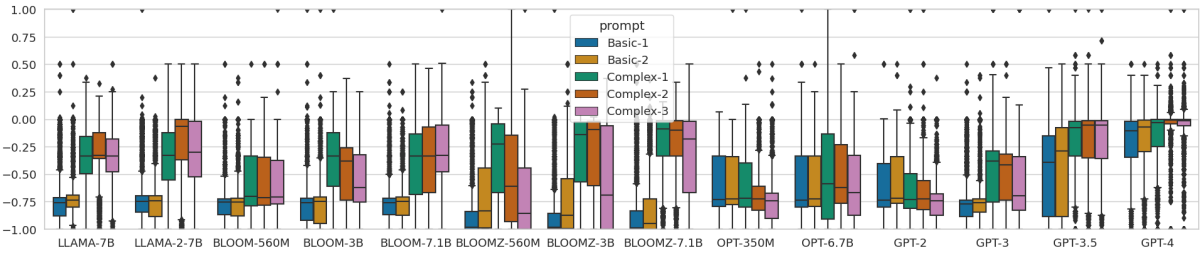

Figure 2: Average of post-correction improvement score across datasets, per model and per prompt, considering post-processed responses to sentence-level text input in the zero-shot setting.

5 Results and Discussion

With eight datasets, three text unit levels, multiple quality bands, fourteen models, five prompts and zero- and few-shot settings, experiments cover many dimensions and yield many results. We first present and discuss metric-based results by iteratively removing dimensions before manually exploring performance factors. Due to to space constraints, some figures are included in the Appendix.

5.1 Metric-based Evaluation

Overall, the performance of LLM-based post-correction is very poor, with a considerable degradation in the quality of original transcriptions across all models, prompts and datasets. Even when considering the most effective setup (sentence level and `Complex-2`), Figure 1 shows that LLMs mostly degrade the input text, occasionally leave it unchanged, and rarely improve it. It is therefore a matter of understanding which setup is the least worst.

Impact of post-processing and text unit level The basic conditions of our experiments include the use or not of response post-processing and the choice of text unit level. Regarding the former, experiments showed that post-processing benefits all models and text units (see Appendix B.2), with GPT-3.5 and 4 requiring the least post-processing and `Complex-2` often being difficult to open models, i.e. requiring most post-processing. Regarding the latter, sentence-level input text yields better results (see Appendix D). Subsequent analyses are therefore based on results from post-processed responses to sentence-level text input.

Impact of prompt template and setup From weak to strong guidance, which prompt template is the best (or causes the least degradation)? Figure 2 provides insights that lead to two observations. First, it is beneficial to provide specific information about the input text, as can be seen with the `Basic-1/2` prompts which systematically produce the strongest degradation. Second, none of the 'best' `Complex` prompts is a clear winner across models and datasets, producing more or less the same magnitude of degradation. Also, changing the execution setup from zero- to few-shot does not bring any improvement. Figure 3 shows that adding three demonstration examples almost systematically further degrades the results for all models except for GPT-3.5 and 4. This is in line with Zhao et al. (2021) who shows the high volatility of results depending on examples and their order.

Impact of models and document type Having eliminated the worst setups, and focusing on zero-shot responses to sentence-level input with the `Complex-2` prompt, which models perform

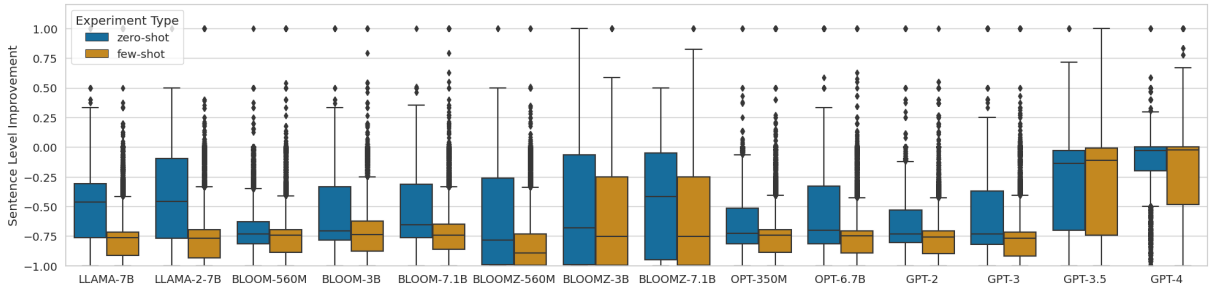

Figure 3: Average of the post-correction improvement score across datasets, per model and in ZS and FS settings, considering post-processed answers to sentence-level text inputs with prompt `Complex-2`.

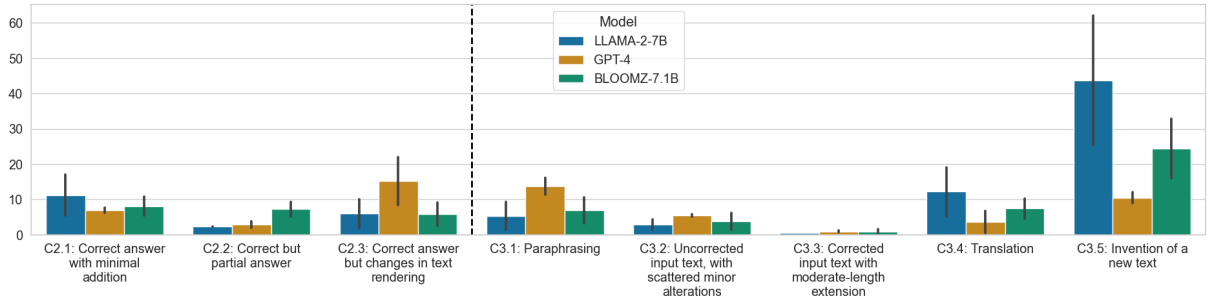

Figure 4: Error category distribution in manually annotated samples across three models, for groups C2 (slight deviation from GT, on the left) and C3 (strong deviation from GT, on the right).

best for which dataset? Regardless of the model, LLM-based post-correction of atypical text material and languages such as in `ajmc` and `htrec` lead to the most severe degradation (Figure 1). For `ajmc-mixed`, the largest multilingual BLOOM and LLaMA models offer some hope, while for `htrec` only GPT-4 seems to be able to save the results from degradation. The situation is less dramatic for `icdar`, `impresso-nzz`, `overproof` and `ina`, with documents closer in language, topics and nature to the training data of the models. As for LLMs, they generally show stability across the datasets, with a few exceptions. Not surprisingly, GPT-3.5 and 4 show the best results. With the exception of `ajmc-mixed`, the cross-lingual multitask fine-tuned BLOOMZ models perform better than their BLOOM counterparts, with no clear difference between model sizes. The smallest models, whether from the LLaMA, BLOOM or OPT series, generally perform on par with GPT-2 and GPT-3.

Impact of OCR quality Finally, the performance distribution among the original transcription quality bands shows that, overall, the noisiest input texts are those that show the least degradation, and sometimes even improvement. This underlines the ability of LLMs to make corrections where necessary, but not to leave almost error-free texts intact (more details in Appendix F).

5.2 Manual Analysis

In addition to improvement scores, we aim to understand the factors influencing transcription quality by manually inspecting around 2,500 LLM-corrected / ground truth transcription pairs, sampled across all datasets and a selection of models, prompts and languages (see Appendix C).

Following initial inspection, we established a tentative taxonomy of LLM errors or behaviours, comprising ten categories organised into four groups:

- C1: unanswered prompts;
- C2: responses with corrected input text but with slight deviations from the GT, thus invalid in terms of PCIS but potentially acceptable within an information retrieval context;
- C3: responses that deviate significantly from the GT, or hallucinations; and
- C4: instances where the GT itself is incorrect.

The error taxonomy, detailed in Appendix C.3, represents cases of instruction inconsistencies, where the model does not do what it is asked to do, and context inconsistencies, where its answer is incorrect (Huang et al., 2023). Figure 4 shows the distribution of errors for three models between C2

and C3, which represent two thirds of the sample. We note that most cases correspond to strong deviations from the GT (C3), with a majority of pure hallucination (C3.5, especially LLaMA-2-7B and BLOOMZ-7.1B), as well as a slight tendency of GPT to propose paraphrases rather than just correct the text. Models also produce smaller deviations (C2), with LLaMA-2-7B marginally continuing the text, BLOOMZ-7.1B giving partial answers, and GPT-4 embellishing the text as it sees fit.

6 Conclusions and Future Work

This exploratory study shows that LLMs are not good at correcting transcriptions of historical documents of any kind, at least in the applied experimental setting. Not only do they not improve the original transcriptions, they usually degrade them, making LLM-based post-correction of historical transcripts a rather distant prospect. Nevertheless, we have found that instructing models about the nature of the input and guiding their output format leads to better results, and that large open-access and multilingual models from the BLOOM(Z) and LLaMA series can compete with commercial GPT models.

On the basis of these findings, future work should investigate in more detail some of the elements that could not be studied due to the scale of the experiments presented here. These include, among others: testing prompts that are even more tailored to the specifics of each document type, distinguishing between error detection and correction prompts (in a chain-of-thought fashion), searching for the temperature hyperparameter, attempting model fine-tuning and model self-evaluation, and consolidating the error taxonomy and error analysis on a few datasets.

Limitations

- Due to time, budget, and computational resource constraints, results are based on single-pass generation.
- Due to the complexity of the materials, text units may be incorrectly segmented and aligned with the GT. Also, the GT may not be 100% correct. This may affect the results.
- Demonstration examples in the few-shot scenario were randomly selected; a manual curation of these could lead to better results in this setting.
- The numerous experiments produced many results that could be further explored and analysed at a finer level for each setting. Nevertheless, we believe that the aggregated results remain informative, further complemented by manual inspection.

Author Contributions

EB and ME formulated the research concept, objectives and methodology, with the contributions of MR and FK. EB was responsible for the code implementation and result visualisation. EB, ME and SNM contributed to data curation and EB and MR worked on data annotation. EB, ME and MR worked on investigation and formal analysis, with the contribution of SNM. EB wrote sections of the original draft, ME wrote the manuscript, and all authors contributed to manuscript revision, read, and approved the submitted version. ME and FK contributed to supervision.

Code and Data Availability

The code used in this work and some datasets are available at `https://github.com/impresso/llm-transcript-postcorrection`.

Acknowledgements

Authors warmly thank the Research and Innovation Department of the French National Audiovisual Institute (INA) for having shared a few radio broadcast transcriptions. Authors also gratefully acknowledge the financial support of the Swiss National Science Foundation (SNSF) for the research projects '*Impresso* - Media Monitoring of the Past II. Beyond Borders: Connecting Newspaper and Radio' under the Sinergia grant CRSII5_213585 and 'Ajax Multi-Commentary' under the Ambizione grant PZ00P1_18603.

References

Chantal Amrhein and Simon Clematide. 2018. Supervised OCR Error Detection and Correction Using Statistical and Neural Machine Translation Methods. *Journal for Language Technology and Computational Linguistics (JLCL)*, 33(1):49–76.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity.

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, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, 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, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan 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 Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the Opportunities and Risks of Foundation Models.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Estelle Bunout, Maud Ehrmann, and Frédéric Clavert, editors. 2022. *Digitised Newspapers – A New Eldorado for Historians?: Reflections on Tools, Methods and Epistemology*. Studies in Digital History and Hermeneutics. De Gruyter Oldenbourg.

Guillaume Chiron, Antoine Doucet, Mickaël Coustaty, and Jean-Philippe Moreux. 2017a. ICDAR2017 Competition on Post-OCR Text Correction. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 1423–1428.

Guillaume Chiron, Antoine Doucet, Mickaël Coustaty, Muriel Visani, and Jean-Philippe Moreux. 2017b. Impact of OCR errors on the use of digital libraries: Towards a better access to information. In *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries*, pages 249–252, Toronto, ON, Canada. IEEE, IEEE Press.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. PaLM: Scaling Language Modeling with Pathways.

Christian Clausner, Apostolos Antonacopoulos, and Stefan Pletschacher. 2019. ICDAR2019 Competition on Recognition of Documents with Complex Layouts - RDCL2019. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1521–1526.

Steven Coyne, Keisuke Sakaguchi, Diana Galvan-Sosa, Michael Zock, and Kentaro Inui. 2023. Analyzing the Performance of GPT-3.5 and GPT-4 in Grammatical Error Correction.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A Survey on In-context Learning.

Maud Ehrmann, Matteo Romanello, Simon Clematide, Phillip Benjamin Ströbel, and Raphaël Barman. 2020. Language Resources for Historical Newspapers: The Impresso Collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 958–968, Marseille, France. European Language Resources Association.

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. *CEUR Workshop Proceedings*, (3180):1038–1063.

Elisabeth Engl. 2020. Volltexte für die Frühe Neuzeit. *Zeitschrift für Historische Forschung*, 47(2):223–250.

John Evershed and Kent Fitch. 2014. Correcting noisy OCR: Context beats confusion. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, DATeCH '14, pages 45–51, New York, NY, USA. Association for Computing Machinery.

Tao Fang, Shu Yang, Kaixin Lan, Derek F. Wong, Jinpeng Hu, Lidia S. Chao, and Yue Zhang. 2023. Is ChatGPT a Highly Fluent Grammatical Error Correction System? A Comprehensive Evaluation.

Mingqi Gao, Jie Ruan, Renliang Sun, Xunjian Yin, Shiping Yang, and Xiaojun Wan. 2023. Human-like Summarization Evaluation with ChatGPT.

Harsh Gupta, Luciano Del Corro, Samuel Broscheit, Johannes Hoffart, and Eliot Brenner. 2021. Unsupervised Multi-View Post-OCR Error Correction With Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8647–8652, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions.

Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Xing Wang, and Zhaopeng Tu. 2023. Is ChatGPT A Good Translator? Yes With GPT-4 As The Engine.

Philip Kahle, Sebastian Colutto, Günter Hackl, and Günter Mühlberger. 2017. Transkribus - A Service Platform for Transcription, Recognition and Retrieval of Historical Documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 04, pages 19–24.

Md Tahmid Rahman Laskar, M. Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Xiangji Huang. 2023. A Systematic Study and Comprehensive Evaluation of ChatGPT on Benchmark Datasets.

Vladimir I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, (10):707.

Elvys Linhares Pontes, Ahmed Hamdi, Nicolas Sidere, and Antoine Doucet. 2019. Impact of OCR Quality on Named Entity Linking. In *Digital Libraries at the Crossroads of Digital Information for the Future*, Lecture Notes in Computer Science, pages 102–115, Cham. Springer International Publishing.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9):195:1–195:35.

Mengsay Loem, Masahiro Kaneko, Sho Takase, and Naoaki Okazaki. 2023. Exploring Effectiveness of GPT-3 in Grammatical Error Correction: A Study on Performance and Controllability in Prompt-Based Methods.

Barbara McGillivray, Beatrice Alex, Sarah Ames, Guyda Armstrong, David Beavan, Arianna Ciula, Giovanni Colavizza, James Cummings, David De Roure, Adam Farquhar, Simon Hengchen, Anouk Lang, James Loxley, Eirini Goudarouli, Federico Nanni, Andrea Nini, Julianne Nyhan, Nicola Osborne, Thierry Poibeau, Mia Ridge, Sonia Ranade, James Smithies, Melissa Terras, Andreas Vlachidis, and Pip Willcox. 2020. The challenges and prospects of the intersection of humanities and data science: A White Paper from The Alan Turing Institute. Technical report, Alan Turing Institute.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M. Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual Generalization through Multitask Finetuning.

Clemens Neudecker and Apostolos Antonacopoulos. 2016. Making Europe's Historical Newspapers Searchable. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 405–410, Santorini, Greece. IEEE.

Thi Tuyet Hai Nguyen, Adam Jatowt, Nhu-Van Nguyen, Mickael Coustaty, and Antoine Doucet. 2020. Neural Machine Translation with BERT for Post-OCR Error Detection and Correction. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, JCDL '20, pages 333–336, New York, NY, USA. Association for Computing Machinery.

OpenAI. 2023. GPT-4 Technical Report.

Robert Ostling and Murathan Kurfalı. 2022. Really good grammatical error correction, and how to evaluate it. In *The Ninth Swedish Language Technology Conference (SLTC2022)*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John

Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Thomas Padilla. 2019. Responsible Operations: Data Science, Machine Learning, and AI in Libraries. OCLC Research Position Paper. Technical report, ERIC.

Christos Papadopoulos, Stefan Pletschacher, Christian Clausner, and Apostolos Antonacopoulos. 2013. The IMPACT dataset of historical document images. In *Proceedings of the 2nd International Workshop on Historical Document Imaging and Processing*, HIP '13, pages 123–130, New York, NY, USA. Association for Computing Machinery.

John Pavlopoulos, Vasiliki Kougia, Paraskevi Platanou, Stepan Shabalin, Konstantina Liagkou, Emmanouil Papadatos, Holger Essler, Jean-Baptiste Camps, and Franz Fischer. 2023. Error Correcting HTR'ed Byzantine Text.

Paraskevi Platanou, John Pavlopoulos, and Georgios Papaioannou. 2022. Handwritten Paleographic Greek Text Recognition: A Century-Based Approach. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6585–6589, Marseille, France. European Language Resources Association.

Alec Radford, Jeff Wu, Rewon Child, D. Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pages 3505–3506, New York, NY, USA. Association for Computing Machinery.

Christian Reul, Dennis Christ, Alexander Hartelt, Nico Balbach, Maximilian Wehner, Uwe Springmann, Christoph Wick, Christine Grundig, Andreas Büttner, and Frank Puppe. 2019. OCR4all—An Open-Source Tool Providing a (Semi-)Automatic OCR Workflow for Historical Printings. *Applied Sciences*, 9(22):4853.

Christophe Rigaud, Antoine Doucet, Mickaël Coustaty, and Jean-Philippe Moreux. 2019. ICDAR 2019 competition on post-OCR text correction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1588–1593. IEEE.

Matteo Romanello, Sven Najem-Meyer, and Bruce Robertson. 2021. Optical character recognition of 19th century classical commentaries: The current state of affairs. In *The 6th International Workshop on Historical Document Imaging and Processing*, HIP '21, pages 1–6, New York, NY, USA. Association for Computing Machinery.

Nipun Sadvilkar and Mark Neumann. 2020. PySBD: Pragmatic Sentence Boundary Disambiguation. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 110–114, Online. Association for Computational Linguistics.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model.

Robin Schaefer and Clemens Neudecker. 2020. A Two-Step Approach for Automatic OCR Post-Correction. In *Proceedings of the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 52–57, Online. International Committee on Computational Linguistics.

Pit Schneider and Yves Maurer. 2022. Rerunning OCR: A Machine Learning Approach to Quality Assessment and Enhancement Prediction. *Journal of Data Mining & Digital Humanities*, 2022(Digital humanities in...):8561.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism.

Elizabeth Soper, Stanley Fujimoto, and Yen-Yun Yu. 2021. BART for Post-Correction of OCR Newspaper Text. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 284–290, Online. Association for Computational Linguistics.

Phillip Ströbel and Simon Clematide. 2019. Improving OCR of black letter in historical newspapers: The unreasonable effectiveness of HTR models on low-resolution images. In *Proceedings of the Digital Humanities 2019, (DH2019)*.

Phillip Benjamin Ströbel, Simon Clematide, and Martin Volk. 2020. How Much Data Do You Need? About the Creation of a Ground Truth for Black Letter and the Effectiveness of Neural OCR. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3551–3559, Marseille, France. European Language Resources Association.

Melissa M. Terras. 2011. The Rise of Digitization. In Ruth Rikowski, editor, *Digitisation Perspectives*, Educational Futures Rethinking Theory and Practice, pages 3–20. SensePublishers, Rotterdam.

Konstantin Todorov and Giovanni Colavizza. 2020. Transfer Learning for Historical Corpora: An Assessment on Post-OCR Correction and Named Entity

Recognition. In *Proceedings of the Workshop on Computational Humanities Research (CHR 2020)*, pages 310–325.

Konstantin Todorov and Giovanni Colavizza. 2022. An Assessment of the Impact of OCR Noise on Language Models.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. LLaMA: Open and Efficient Foundation Language Models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open Foundation and Fine-Tuned Chat Models.

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.

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, pages 5998–6008, Long Beach, California, US. Curran Associates, Inc.

Haoran Wu, Wenxuan Wang, Yuxuan Wan, Wenxiang Jiao, and Michael Lyu. 2023. ChatGPT or Grammarly? Evaluating ChatGPT on Grammatical Error Correction Benchmark.

Ismet Zeki Yalniz and R. Manmatha. 2011. A Fast Alignment Scheme for Automatic OCR Evaluation of Books. In *2011 International Conference on Document Analysis and Recognition*, pages 754–758.

Bowen Zhang, Daijun Ding, and Liwen Jing. 2023. How would Stance Detection Techniques Evolve after the Launch of ChatGPT?

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A Survey of Large Language Models (v11).

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate Before Use: Improving Few-shot Performance of Language Models. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12697–12706. PMLR.

A LLMs hyperparameters

Experiments with OpenAI API were conducted using `gpt2`, `davinci`, `gpt-3.5-turbo` and `gpt-4` models with the default temperature of 0.7. Experiments with the open-source models were conducted using a default temperature of 1.0.

B Post-processing

B.1 Post-processing heuristics

Post-processing of LLMs answers involves the following heuristics:

- Removal of leading and trailing white spaces and double quotes from the response.
- Removal of (parts of) prompts from the response.
- Trimming of the predicted text so that it does not exceed 1.5 times the length of the input text. This constraint ensures that the prediction does not deviate excessively in length from the original digitised text.
- Removal of specific phrases such as "There is no text provided to correct" or "No correction needed" from the response.

B.2 Impact of post-processing on post-correction improvement score

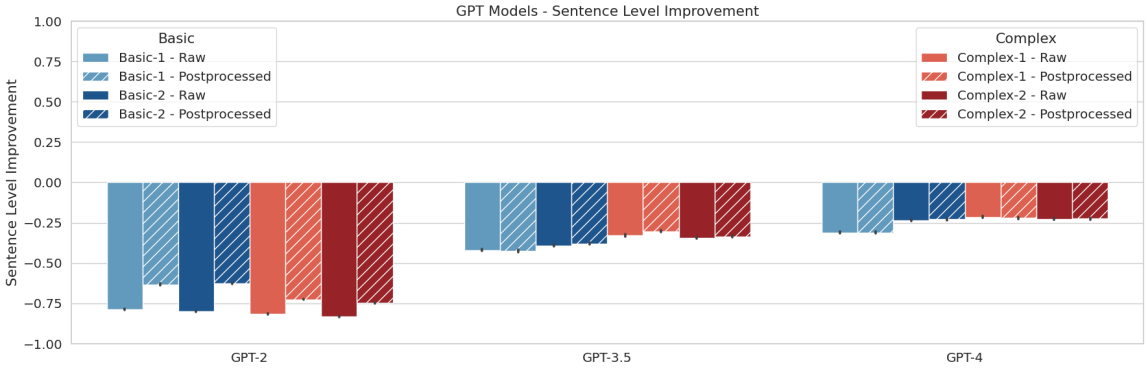

Figure 5: Post-correction improvement scores across datasets for GPT-2, GPT-3.5 and GPT-4 per prompt to sentence level input with (post-processed) and without (raw) post-processing.

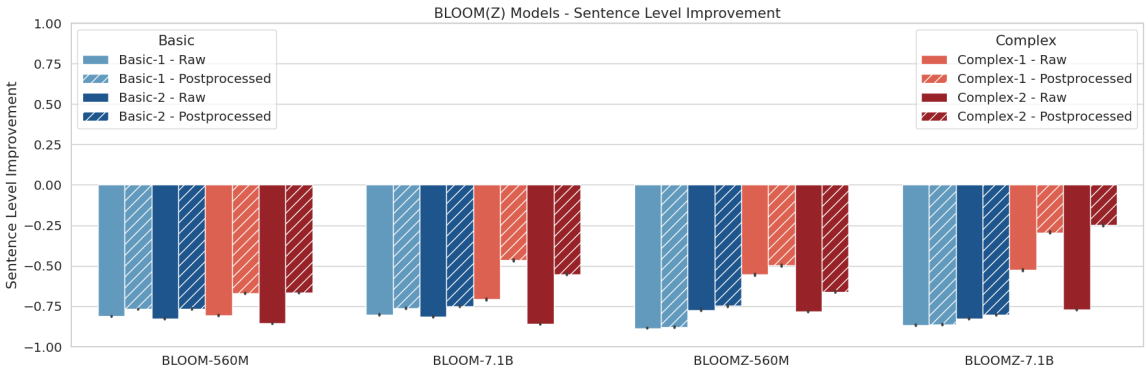

Figure 6: Post-correction improvement scores across datasets for BLOOM-560M, BLOOM-7.1B, BLOOMZ-560M and BLOOMZ-7.1B per prompt to sentence level input with (post-processed) and without (raw) post-processing.

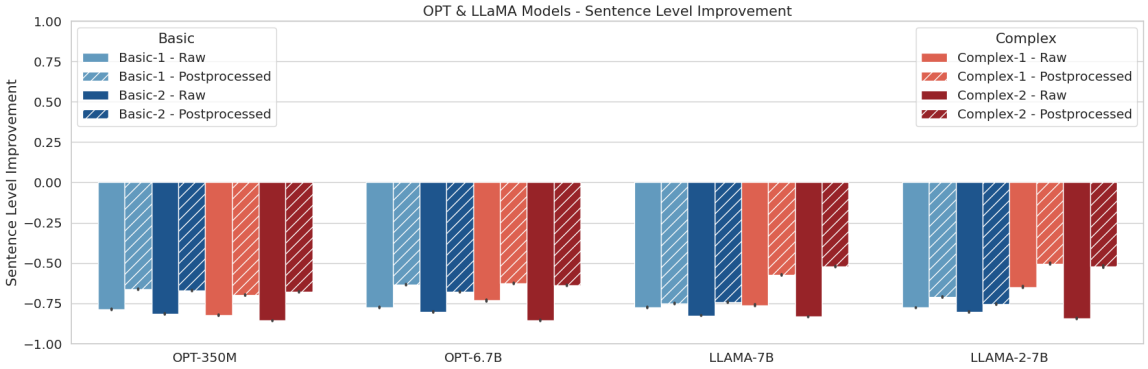

Figure 7: Post-correction improvement scores across datasets for OPT-350M, OPT-6.7B, LLaMA-7B and LLaMA-2-7B per prompt to sentence level input with (post-processed) and without (raw) post-processing.

C Manual Analysis

C.1 Sampling

To manually inspect pairs of LLM-corrected vs. ground truth transcriptions, we sampled a total of 2,459 post-correction items at sentence level. One item was sampled from each dataset (from different quality bands), considering three models (GPT-4, LLaMA-2, BLOOMZ), two prompts (`Basic-2` and `Complex-2`), and four languages (en, de, fr, grc). Table 4 shows the number of items selected per datasets.

Dataset	# items	Percentage
`ajmc-mixed`	41	10%
`ajmc-primary`	14	51%
`htrec`	8	100%
`icdar-2017`	85	18%
`icdar-2019`	61	15%
`impresso-nzz`	45	7%
`ina`	44	15%
`overproof`	36	9%

Table 4: Number and percentage of sampled items per dataset.

C.2 Error category distribution

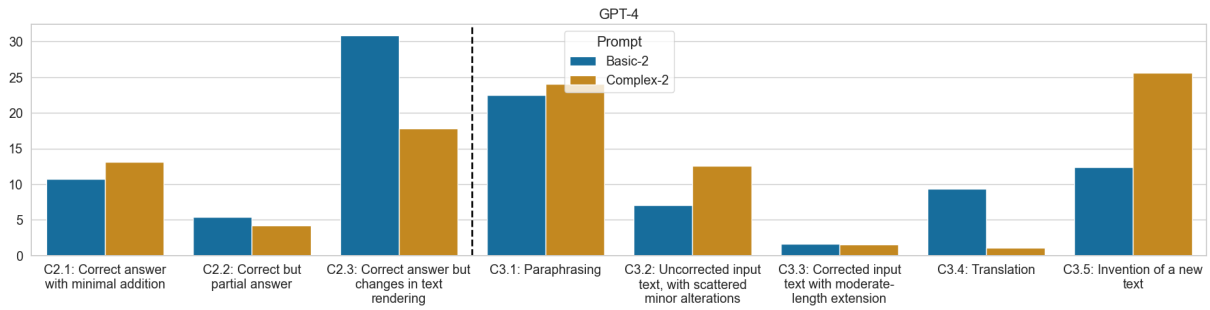

Figure 8: Error category distribution in manually annotated samples per prompt across datasets for GPT-4.

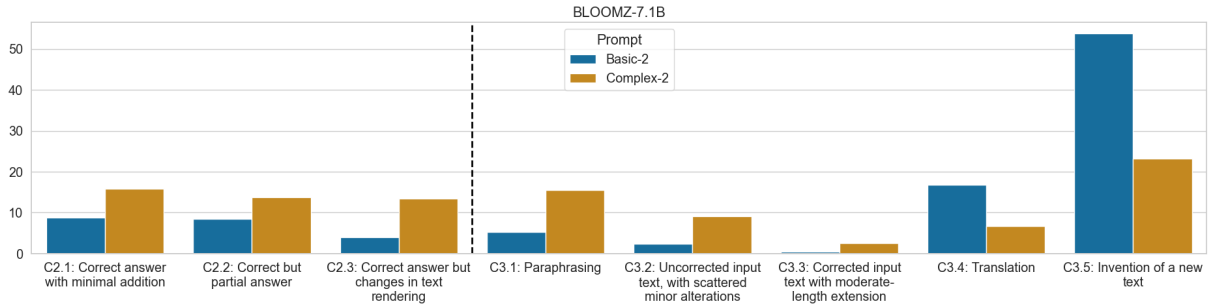

Figure 9: Error category distribution in manually annotated samples per prompt across datasets for BLOOMZ-7.1B.

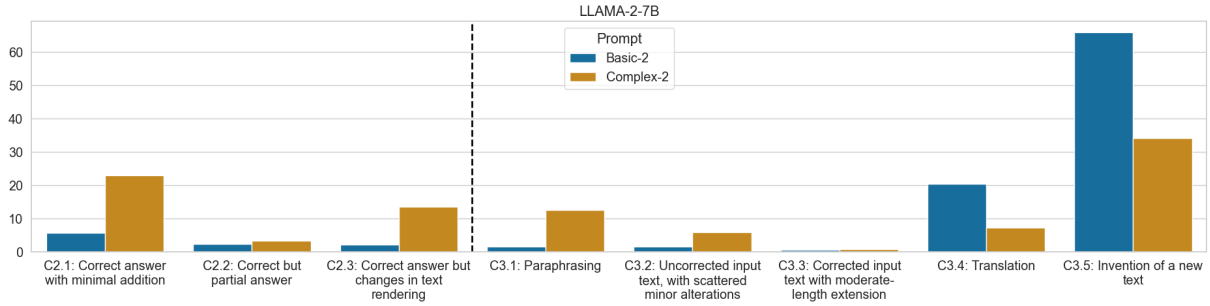

Figure 10: Error category distribution in manually annotated samples per prompt across datasets for LLAMA-2-7B.

C.3 Taxonomy of errors

	CATEGORY	DESCRIPTION	EXAMPLE
		Post-correction task not completed	
C1	**Empty answer** (*instruction inconsistency*)	Unanswered prompt. This includes: empty answers, answers containing some introductory text, and repetitions of (part of) the prompt.	**GT:** - καί gehört zu θανών. **Input:** — zei: gehört zu θανών. **Output:** **Dataset:** `ajmc-mixed` **Model:** BLOOMZ-7.1B **Prompt:** `Basic-2`
		Post-correction task partially completed	
C2.1	**Correct answer with minimal addition** (*context inconsistency*)	The input text is corrected, but extra characters or words (up to a small sentence) are added without changing the meaning.	**GT:** The occupation of the Trobriands and the Woodlark area was successfully accomplished without opposition, and apparently unbeknown to the enemy. **Input:** The occupation of the Trobriands and the Woodlark area was success- fully accomplished without opposition, and apparently unbeknown to the enemy. **Output:** "The occupation of the Trobriands and the Woodlark area was successfully accomplished without opposition, and apparently unknowingly to the enemy. Thank you for your assistance</s>" **Dataset:** `overproof` **Model:** BLOOMZ-7.1B **Prompt:** `Complex-2`
C2.2	**Correct but partial answer** (*context inconsistency*)	The input text is corrected but incomplete (the model did half of the work).	**GT:** Les deux hommes ont été arrêtés, ils se réclament de l'Armée de Libération palestinienne, on ignore leur motif. **Input:** les deux hommes ont été arrêtés il se réclame de l' armée de libération palestinienne en irlande u **Output:** les deux hommes ont été arrêtés **Dataset:** `ina` **Model:** LLAMA-2-7B **Prompt:** `Basic-2`

C2.3	**Correct answer but changes in text rendering** (*instruction inconsistency*)	The input text is corrected, but the model makes changes that improve the formatting and/or readability of the text (carriage returns, case adjustment, de-hyphenation, diacritic correction, spelling normalisation).	**GT:** Fryer Bacon wondered at this man's resolution then asked hee of him, who set him on worke to bee a murderer ? **Input:** Fryer Bacon wondered at this man's resolution then asked hee of him, who set him on worke to bee a murderer ? **Output:** Friar Bacon wondered at this man's resolution, then asked him, "Who set you to work to be a murderer? **Dataset:** `icdar-2017` **Model:** GPT-4 **Prompt:** `Basic-2`
	Post-correction task not completed, with various types of hallucination		
C3.1	**Paraphrasing**	The produced text is close to the input text, but it is a rephrased version of it.	**GT:** Il gémissait des paroles que l'histoire a fidèlement gardées. **Input:** il paie des paroles que l'histoire fidèlement garder **Output:** Il peint des paroles que l'histoire fidèlement garde **Dataset:** `ina` **Model:** GPT-4 **Prompt:** `Complex-2`
C3.2	**Uncorrected input text, with scattered minor alterations** (*instruction inconsistency*)	The original text has not been corrected. Small changes have been made throughout.	**GT:** So little is, in this Cafe, the Do ctrine of Nature beholding to the Philosophy of Des-Cartes. **Input:** So little is, in this Case, the Do- ctrine of Nature beholding to the Philosophy of Des-Cartes. **Output:** It is so little the doctrine of nature that it looks upon the philosophy of Descartes.</s> **Dataset:** `icdar-2019` **Model:** BLOOMZ-7.1B **Prompt:** `Complex-2`
C3.3	**Corrected input text with moderate-length extension** (*instruction inconsistency*)	Some elements of the input text are retained in the answer, which is also continued over several sentences that still relate to the input text.	**GT:** Diesbach hatte nach Wurstisen für die päbstliche Anerkennung seiner vom Bischof und Domkapitel getroffenen Wahl zum Coadjutor zu entrichten: **Input:** Diesbach hatte nach Wurstisen für die päbstliche Anerkennung seiner vom Bischof und Domkapitel getroffenen Wahl zum Coadjutor ru cntrtdbten: **Output:** Dießbach hatte nach Wurstisen für die päbstliche Anerkennung seiner vom Bischof und Domkapitel getroffenen Wahl zum Coadjutor für den tts-oder die nach Wurstisen für die</s> **Dataset:** `impresso-nzz` **Model:** BLOOMZ-7.1B **Prompt:** `Complex-2`
C3.4	**Translation** (*instruction inconsistency*)	The model translates the input text or generates another text in another language.	**GT:** Menelaos, früherer Dienste vergessend, redet, als hätte Aias eben erst Gelegenheit gege- ben, ihn zu erproben. **Input:** Menelaos, früherer Dienste vergessend, redet, als hätte Aias eben erst Gelegenheit gege- ben, ihn zu erproben. **Output:** "Menelaos, forgetting the previous services, speaks as if Aias had recently provided him an opportunity to be tested." **Dataset:** `ajmc-mixed` **Model:** LLAMA-2-7B **Prompt:** `Complex-2`
C3.5	**Invention of a new text** (*context inconsistency*)	The input text disappears and the model hallucinates by inventing a completely different story, usually in the same style as the input text.	**GT:** Nostro in aliis et alieno in omnibus jure salvo. **Input:** Nostro in aliis et alieno in omnibus jure salvo. **Output:** Eodem tempore ut etiam ab aliis ad aliendos personam ad omnes juris iuri **Dataset:** `icdar-2019` **Model:** LLAMA-2-7B **Prompt:** `Complex-2`

D Detailed overviews of results with different text unit levels

D.1 Line level

D.1.1 `overproof` and `impresso-nzz`

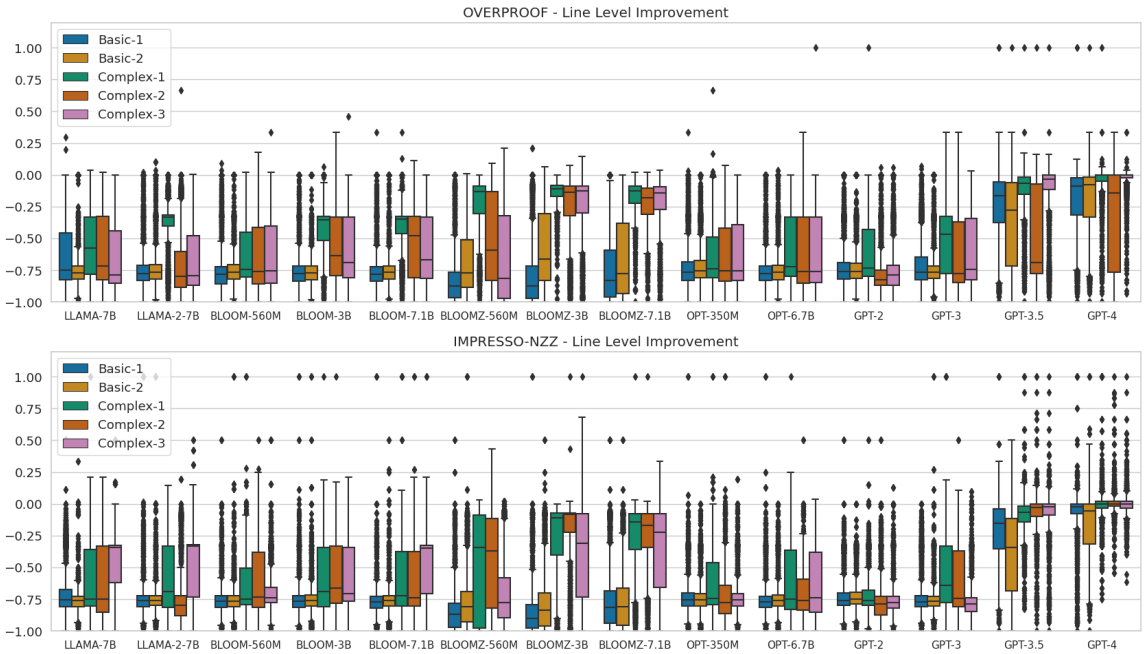

Figure 11: Post-correction improvement score for **`overproof`** and **`impresso-nzz`** datasets, considering all models and prompts based on post-processed responses to **line-level input** in the zero-shot setting.

D.1.2 `ajmc`, `htrec` and `ina`

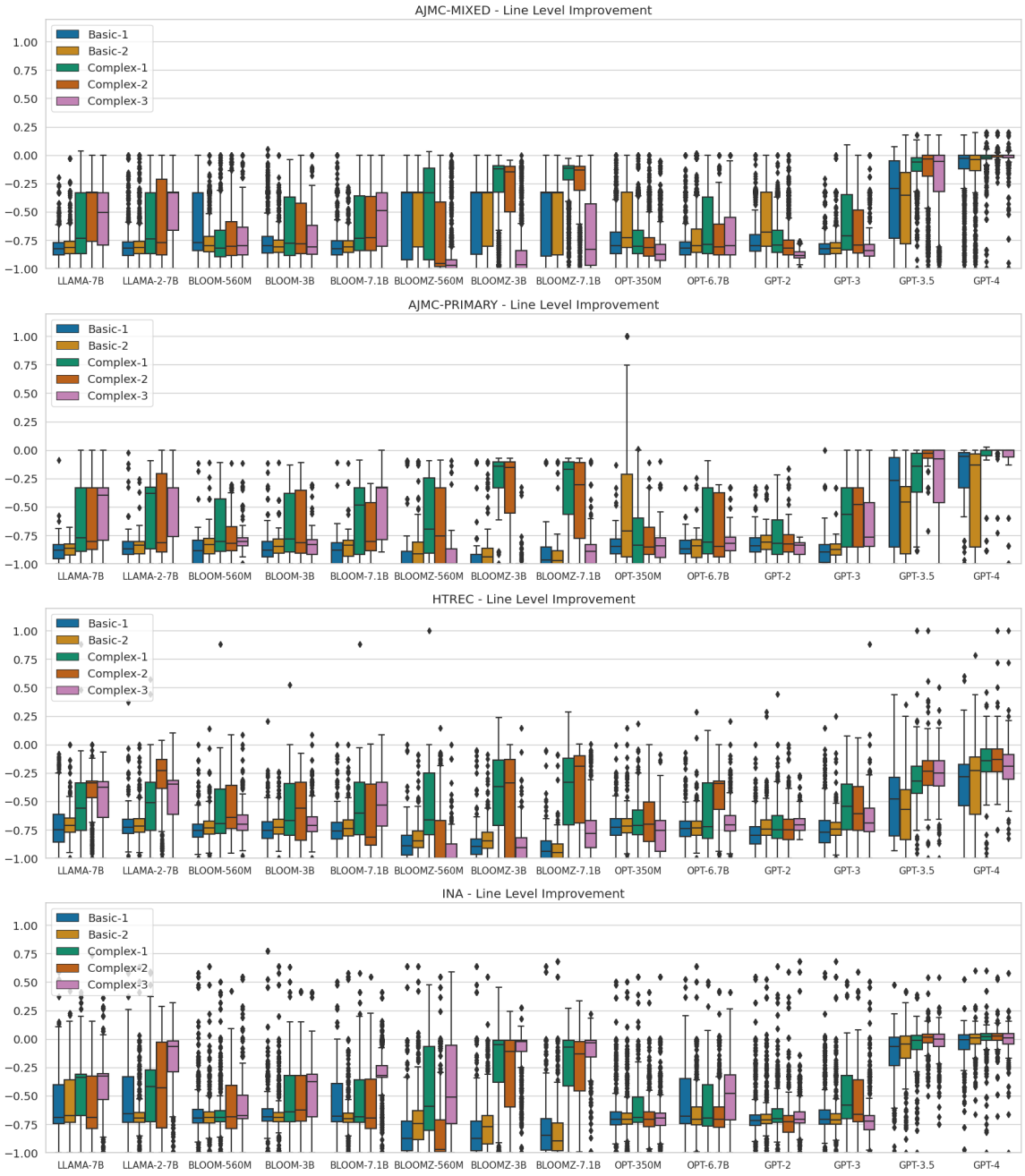

Figure 12: Post-correction improvement score for **`ajmc`**, **`htrec`** and **`ina`** datasets, considering all models and prompts based on post-processed responses to **line-level inputs** in the zero-shot setting.

D.2 Sentence level

D.2.1 `icdar`, `overproof` and `impresso-nzz`

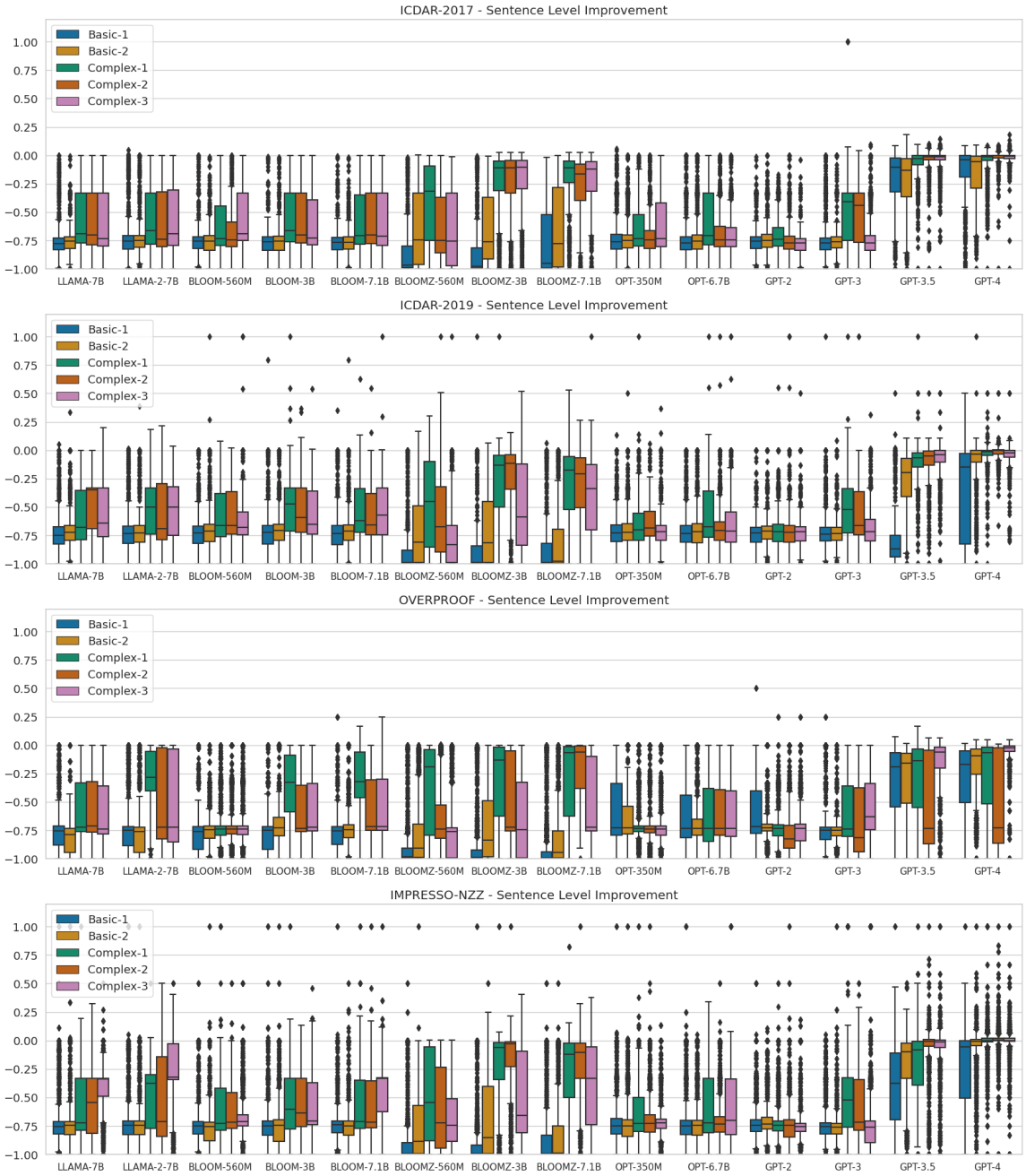

Figure 13: Post-correction improvement score for **`icdar`**, **`overproof`** and **`impresso-nzz`** datasets, considering all models and prompts based on post-processed responses to **sentence-level input** in the zero-shot setting.

D.2.2 `ajmc`, `htrec` and `ina`

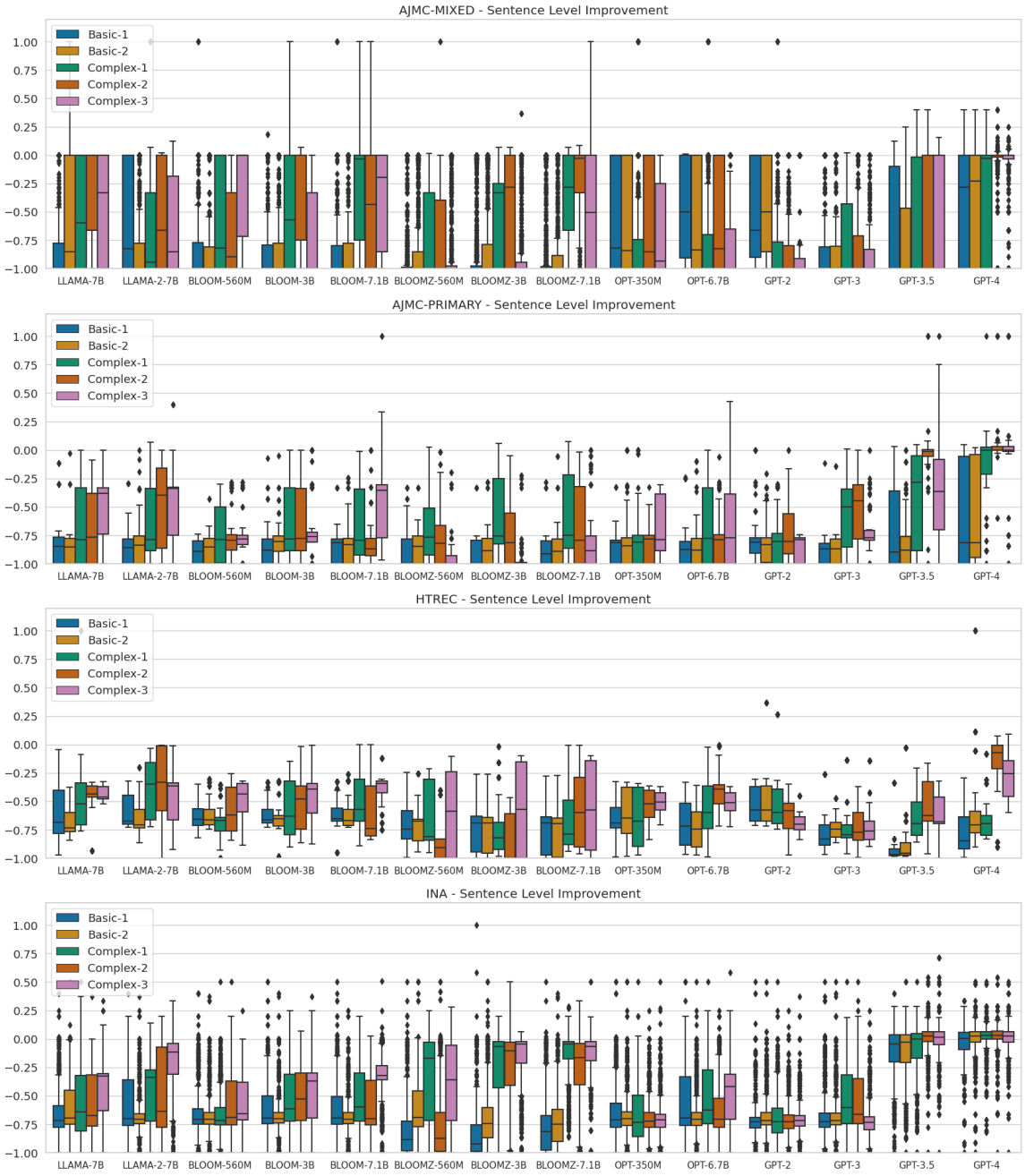

Figure 14: Post-correction improvement score for **`ajmc`**, **`htrec`** and **`ina`** datasets, considering all models and prompts based on post-processed responses to **sentence-level inputs** in the zero-shot setting.

D.3 Region level

D.3.1 `icdar`, `overproof` and `impresso-nzz`

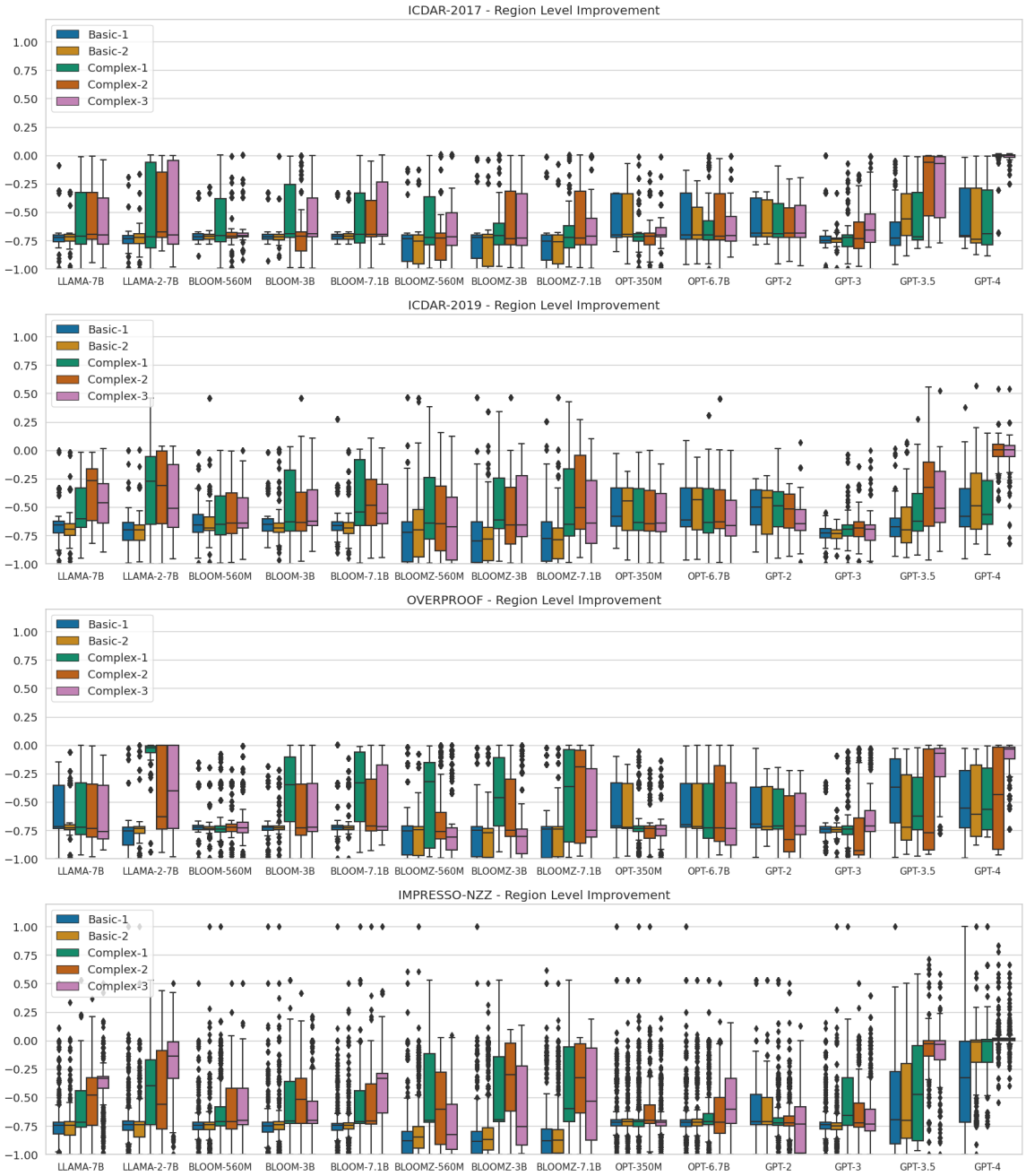

Figure 15: Post-correction improvement score for **`icdar`**, **`overproof`** and **`impresso-nzz`** datasets, considering all models and prompts based on post-processed responses to **region-level input** in the zero-shot setting.

D.3.2 `ajmc`, `htrec` and `ina`

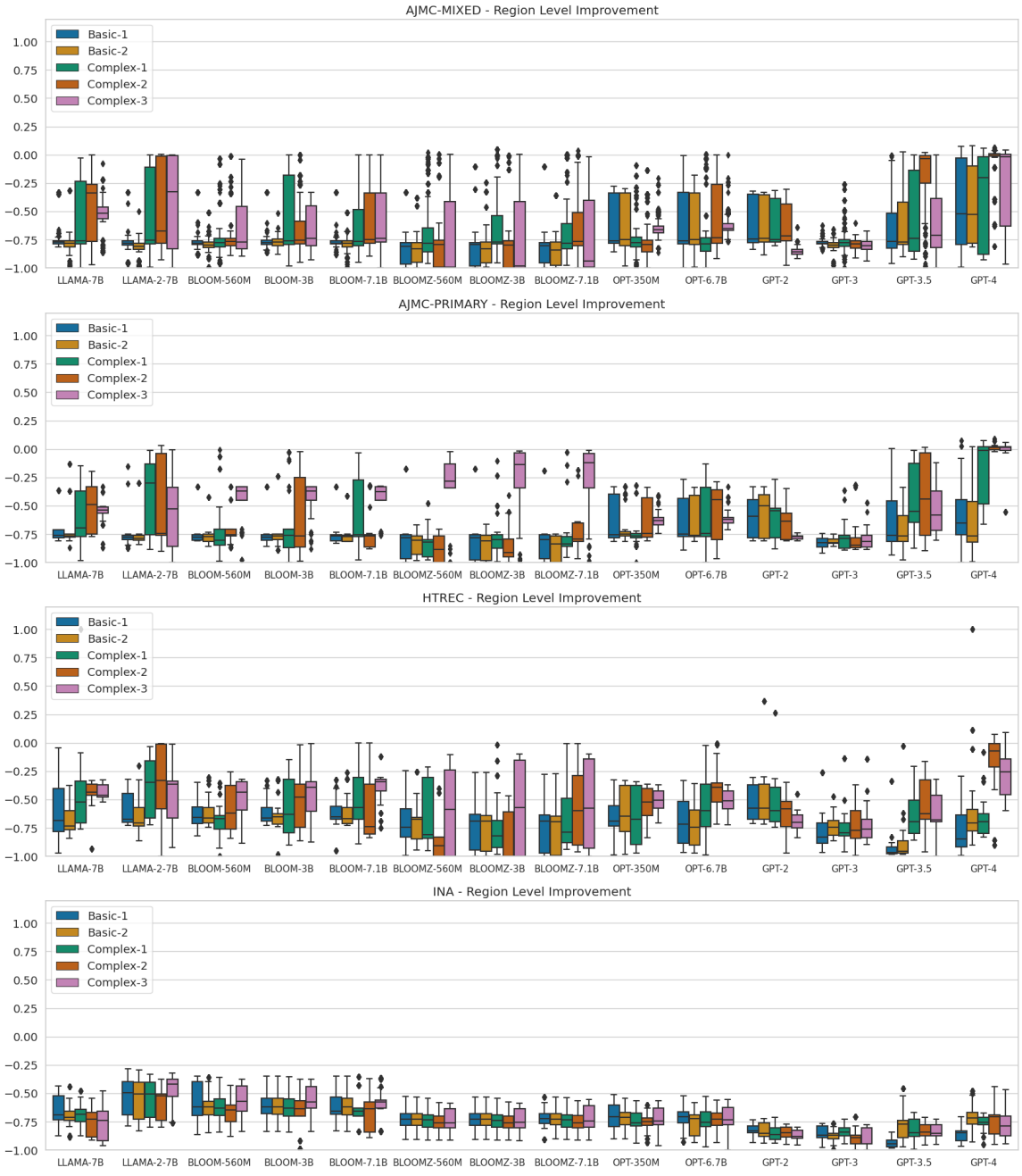

Figure 16: Post-correction improvement score for **`ajmc`**, **`htrec`** and **`ina`** datasets, considering all models and prompts based on post-processed responses to **region-level inputs** in the zero-shot setting.

E Detailed overviews of results in the zero- and few-shot scenarios

E.1 `icdar`, `overproof` and `impresso-nzz`

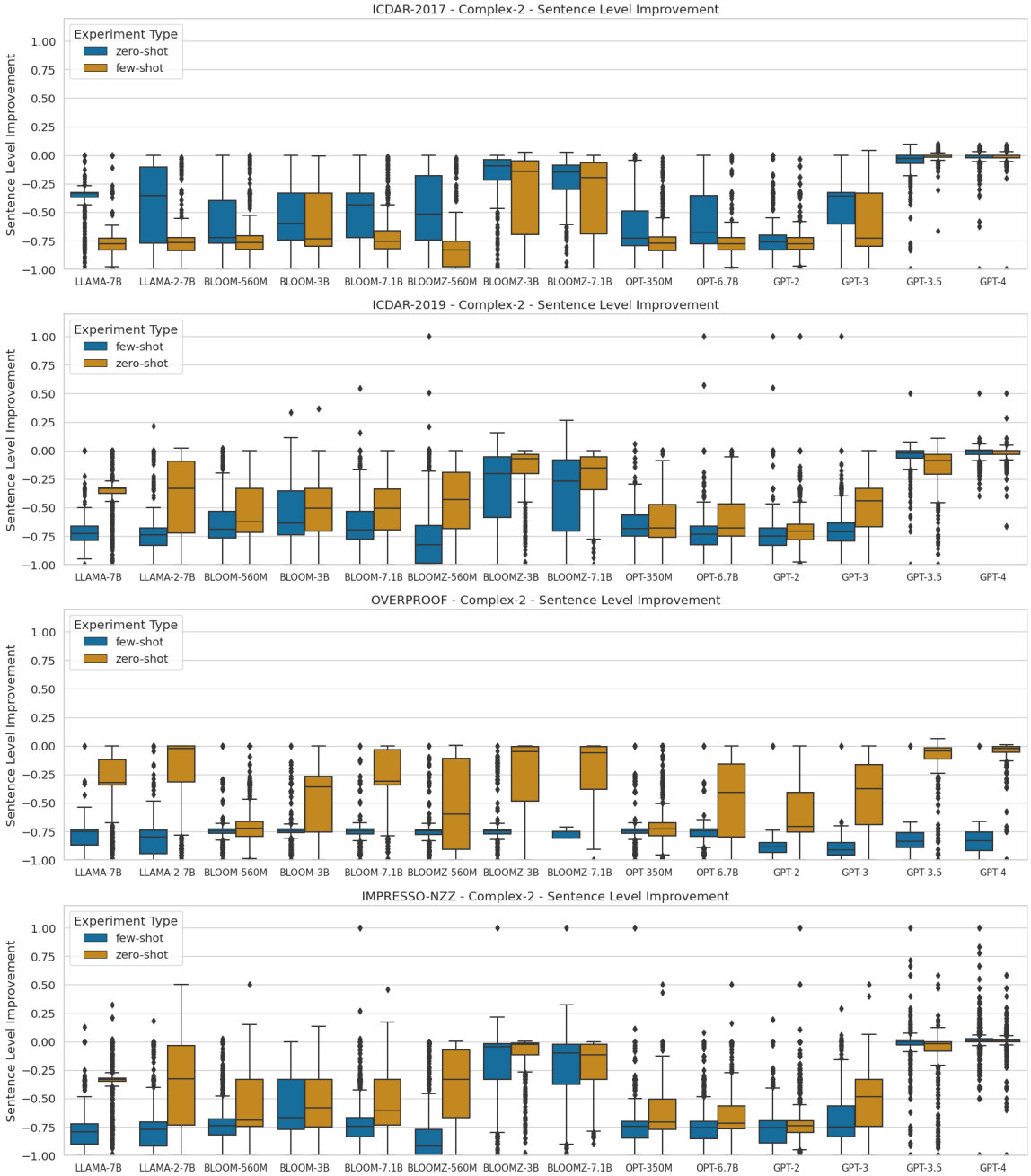

Figure 17: Post-correction improvement scores per model for **`icdar`**, **`overproof`** and **`impresso-nzz`** datasets, considering post-processed responses to **sentence-level input** with `Complex-2` prompt in the zero and few-shot settings.

E.2 `ajmc`, `htrec` and `ina`

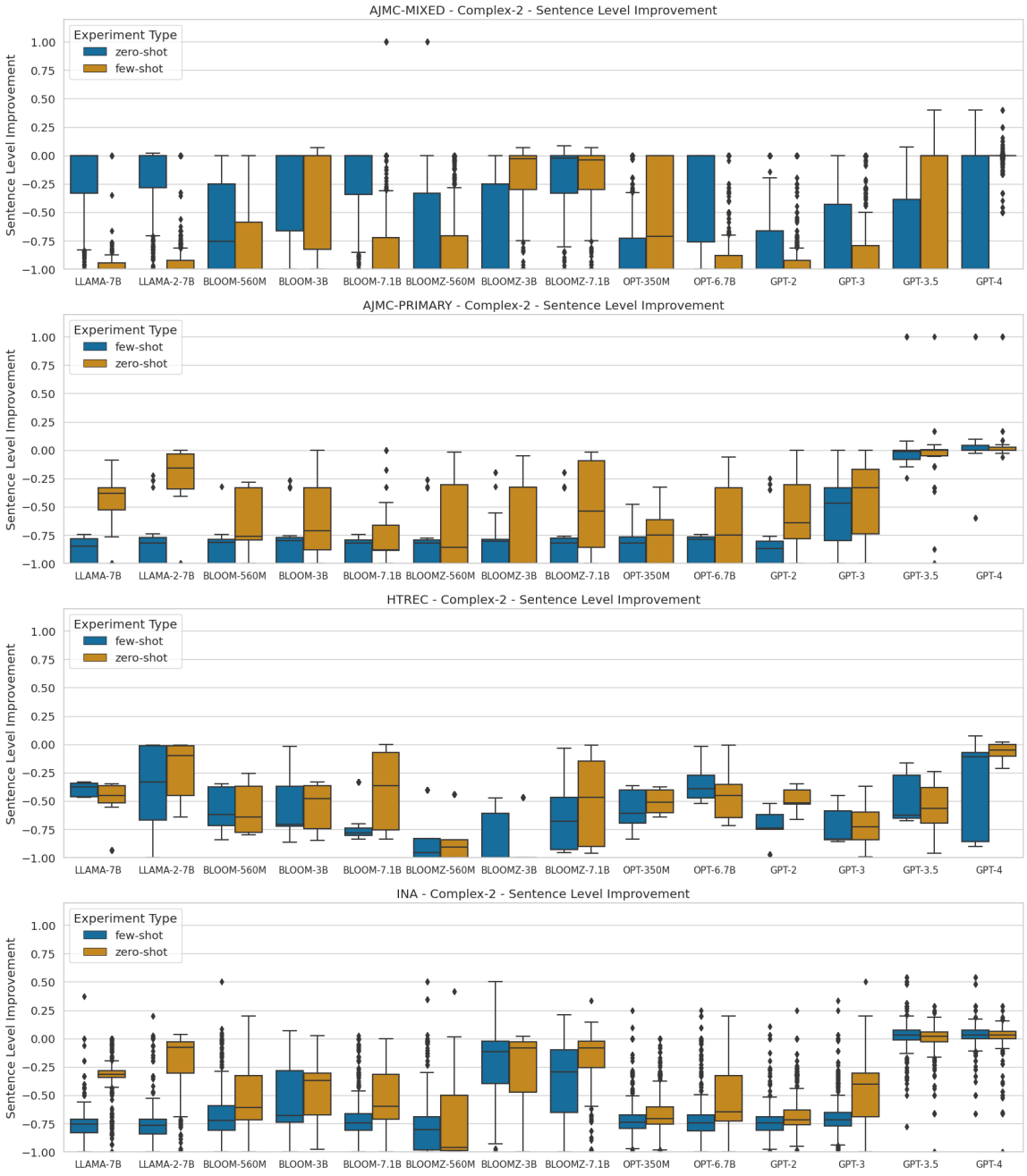

Figure 18: Post-correction improvement scores per model for **`ajmc`**, **`htrec`** and **`ina`** datasets, considering post-processed responses to **sentence-level input** with `Complex-2` prompt in the zero and few-shot settings.

F Detailed overviews of results with different quality bands

F.1 `icdar`, `overproof` and `impresso-nzz`

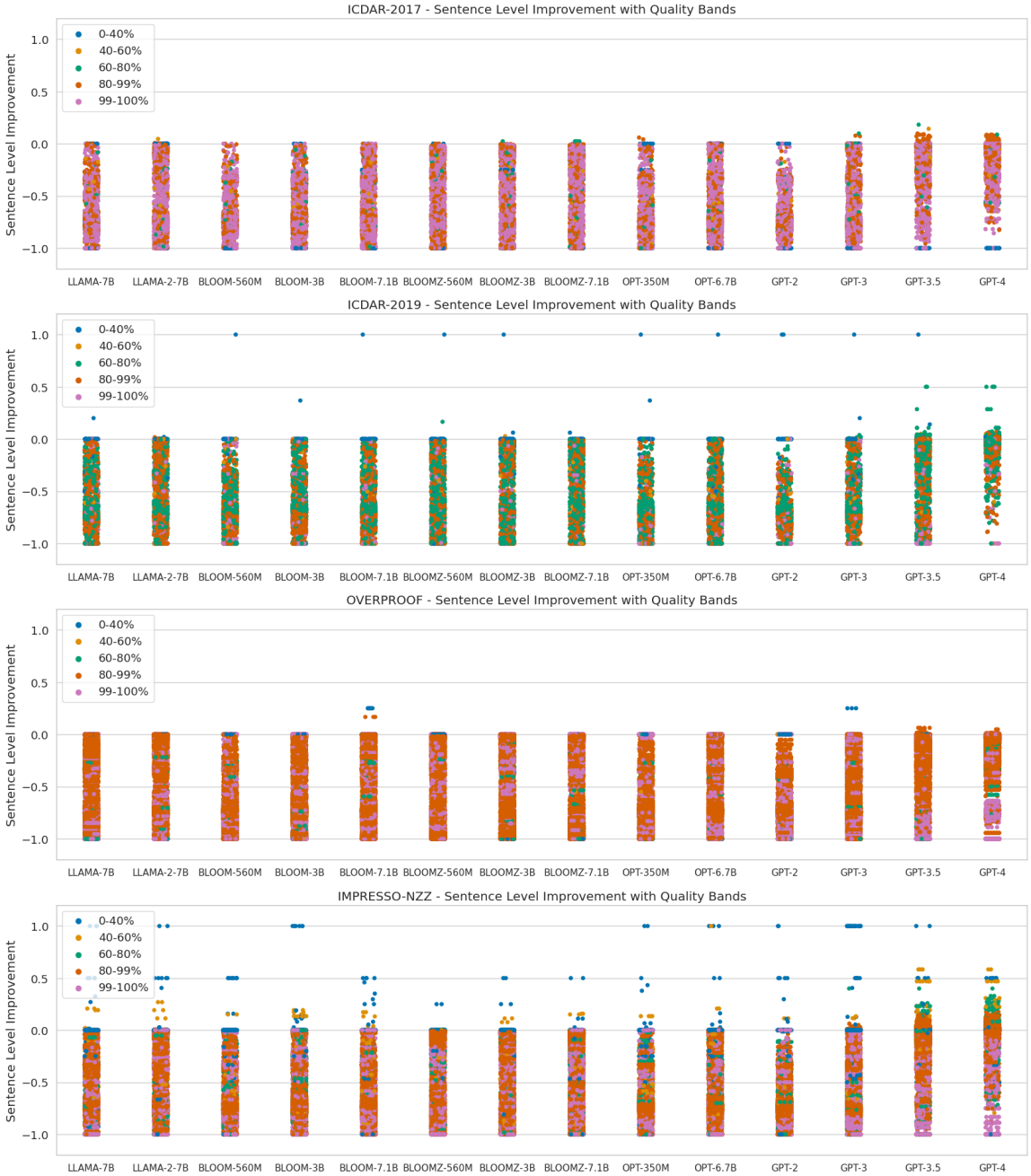

Figure 19: Post-correction improvement scores per quality band and model for **`icdar`**, **`overproof`** and **`impresso-nzz`** datasets, considering post-processed responses to **sentence-level input** in the zero-shot setting.

F.2 `ajmc`, `htrec` and `ina`

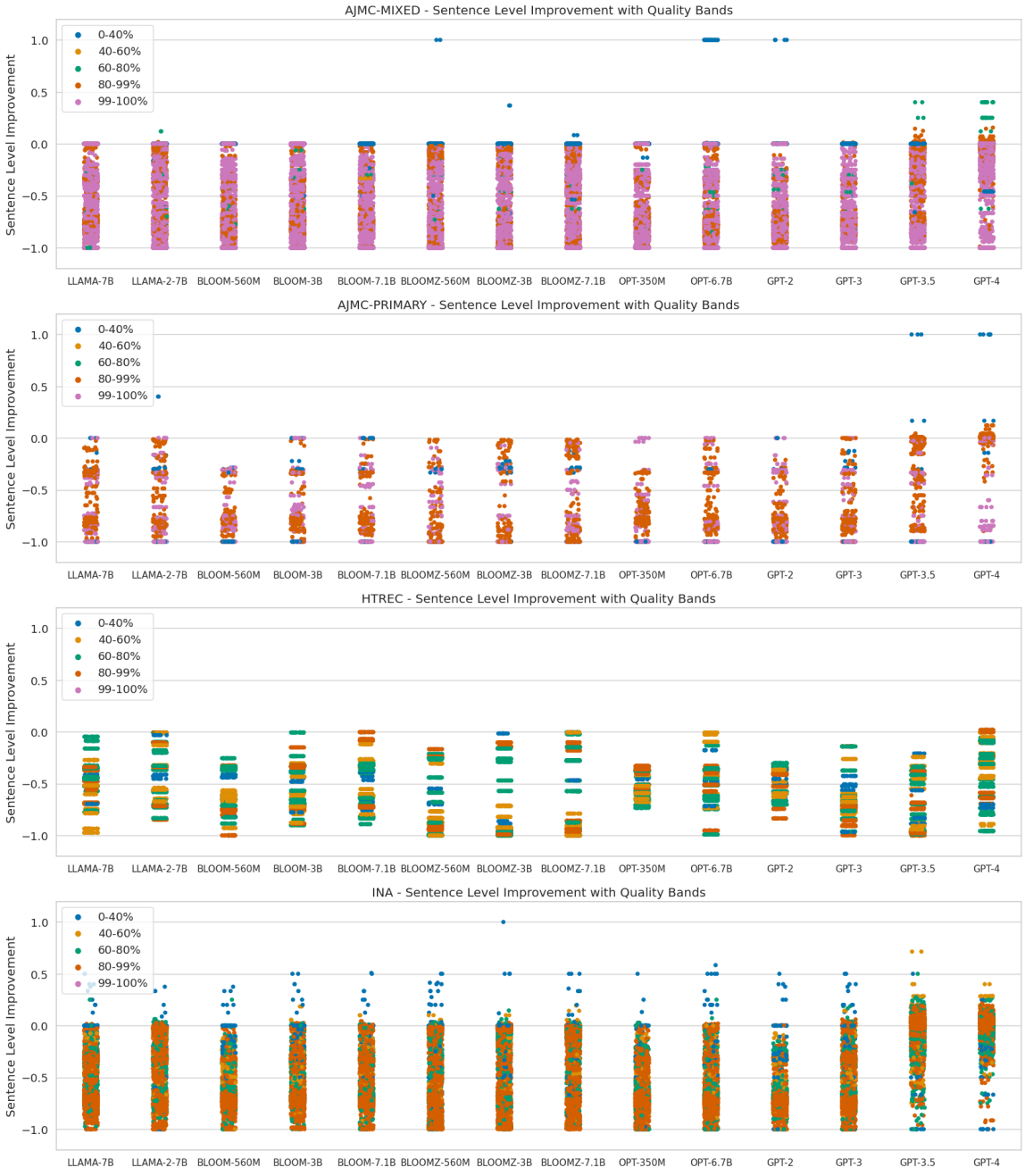

Figure 20: Post-correction improvement scores per quality band and model for **`ajmc`**, **`htrec`** and **`ina`** datasets, considering post-processed responses to **sentence-level input** in the zero-shot setting.