Semi-automatic Sequential Sentence Classification in the Discourse Analysis Tool Suite

Tim Fischer, Chris Biemann
Language Technology Group, Department of Informatics, University of Hamburg, Germany
{firstname.lastname}@uni-hamburg.de

Abstract

This paper explores an AI-assisted approach to sequential sentence annotation designed to enhance qualitative data analysis (QDA) workflows within the open-source Discourse Analysis Tool Suite (DATS) developed at our university. We introduce a three-phase Annotation Assistant that leverages the capabilities of large language models (LLMs) to assist researchers during annotation. Based on the number of annotations, the assistant employs zero-shot prompting, few-shot prompting, or fine-tuned models to provide the best suggestions. To evaluate this approach, we construct a benchmark with five diverse datasets. We assess the performance of three prominent open-source LLMs — Llama 3.1, Gemma 2, and Mistral NeMo — and sequence tagging models based on SentenceTransformers. Our findings demonstrate the effectiveness of our approach, with performance improving as the number of annotated examples increases. Consequently, we implemented the Annotation Assistant within DATS and report the implementation details. With this, we hope to contribute to a novel AI-assisted workflow and further democratize access to AI for qualitative data analysis.

1 Introduction

The Discourse Analysis Tool Suite (DATS) (Schneider et al., 2023) is a platform developed at our university to empower Digital Humanities (DH) researchers in conducting qualitative data analysis (QDA). Developed collaboratively and tailored to the specific needs of DH scholars, the platform democratizes access to machine learning methods. It enables non-experts to effectively manage and analyze large-scale, unstructured, multi-modal data.

While the platform's overarching design is rooted in Grounded Theory-based research (Strauss and Corbin 1990, Strauss et al. 1996), its versatile features support various disciplines. Key functionalities encompass automated pre-processing of multi-modal data (text, image, audio, and video), comprehensive data exploration capabilities, and diverse quantitative analysis tools.

One of the core QDA tasks on our platform is in-depth qualitative analysis through the manual annotation of text documents. This process involves searching for documents relevant to the users' research question, creating and extending a category system (taxonomy), and diving into the material to annotate relevant text passages. While DATS currently supports span-level annotations (similar to named entity recognition), we observed during collaborations with researchers in various disciplines that they primarily annotate at the sentence or paragraph level (i.e., annotating a sequence of sentences). Moreover, they often want to analyze the distribution of their taxonomy across the entire dataset – for example, to see how a particular category usually occurs or how categories relate to each other. However, obtaining such quantitative insights requires annotating a large amount of data, which is often impractical to do manually.

To address this need, this work aims to support sequential sentence annotation within DATS and provide researchers with tools for efficient and practical analysis. In collaboration with our project partners, we have developed a new user interface specifically designed for sequential sentence annotation. Here, a single sentence is the smallest possible unit to annotate. This interface includes functionalities for comparing annotations between users and, most importantly, an Annotation Assistant that learns from existing annotations to provide suggestions for unseen data.

Central to our approach is a three-phase Annotation Assistant designed to provide progressively refined suggestions as users annotate data. Key requirements drive this design, including user control over the annotation process, optimal suggestions at each stage, and fast inference for a smooth user experience. The three phases are: (1) Zero-shot

prompting: When no annotations are available, the assistant leverages the taxonomy and definitions to generate initial suggestions via zero-shot prompting of Large Language Models (LLMs). (2) Few-shot prompting: As users annotate data, the assistant transitions to few-shot prompting, refining its suggestions by incorporating user-provided examples. (3) Fine-tuned model: With sufficient annotations, a sequence tagger can be fine-tuned to this data, further improving the accuracy of suggestions.

To evaluate the effectiveness of our three-phase approach and choose the best model to integrate into the DATS, we construct a benchmark consisting of five datasets on sequential sentence classification. We assess the zero- and few-shot performance of three prominent open-source LLMs: LLama 3.1, Gemma 2, and Mistral NeMo. Further, we fully fine-tune and evaluate sequence tagging models on this benchmark. Our approach works best with Mistral NeMo, steadily improving with an increasing number of annotated samples. The contributions of this paper are:

1. We formulate an AI-assisted sequential sentence annotation workflow as envisioned by us and our project partners, highlighting user needs and requirements.
2. We propose a three-phase Annotation Assistant that, depending on the number of annotated examples, utilizes zero-shot, few-shot prompting, or a fully fine-tuned model.
3. We construct a benchmark on sequential sentence classification to evaluate the approach.
4. We report on integrating AI-assisted sequential sentence annotation into DATS.

2 Related work

QDA Platforms and AI Integration Several platforms and software solutions for qualitative data analysis exist, each offering distinct functionalities to researchers. Some platforms have taken notable steps towards incorporating AI-powered features into their workflows.

CATMA (Gius et al., 2022) is a versatile QDA tool focusing on text and image analysis but has no built-in AI capabilities.

Recognized for qualitative and mixed-methods research, MAXQDA[1] has introduced "MAXQDA AI Assist," offering AI-driven features like summarization, paraphrasing, concept explanation, and automatic transcription. Recently, "AI Coding" was made available as a beta feature that assists with annotating single documents.

NVivo[2], a platform for qualitative data analysis, integrates AI features in its latest beta version, including thematic coding, sentiment analysis, and text summarization.

Known for visual and network analysis tools, Atlas.ti[3] integrates OpenAI's GPT models with existing features like code suggestions, sentiment analysis, summarization, and entity recognition.

Remarkably, AI-powered features are only found in paid versions of such QDA platforms. Further, none of these tools currently offer functionalities to train models and automatically analyze large corpora for quantitative insights. In contrast, DATS aims to democratize access to state-of-the-art AI capabilities, making advanced functionalities such as the proposed three-phase annotation assistance freely available to researchers across disciplines. To avoid data protection issues often associated with cloud-based AI services, DATS can be run in-house without sending sensitive research data to third-party providers.

Benchmarking Several general LLM benchmarks like MMLU (Hendrycks et al., 2021), SuperGLUE (Wang et al., 2019), BIG-bench (Srivastava et al., 2023), HELM (Liang et al., 2023), and MTEB (Muennighoff et al., 2023) have emerged. While they cover many tasks, they may not necessarily be the most relevant for QDA. Ziems et al. (2024) evaluates LLMs' zero-shot performance on a range of Computational Social Science (CSS) tasks focusing on taxonomic labeling and free-form coding. Their experiments indicate LLMs' great potential to augment CSS research as zero-shot data annotators, strongly motivating our approach to using LLMs in zero- and few-shot scenarios.

Our benchmark evaluates sequential sentence classification across various domains and tasks related to CSS and informs the integration of our proposed Annotation Assistant into DATS.

3 Envisioned workflow

Inspired by project partners who actively work with the Discourse Analysis Tool Suite, this section describes an illustrative data analysis workflow highlighting potential areas where the Annotation Assistant could enhance productivity.

[1] https://maxqda.com

[2] https://nvivo.de/

[3] https://atlasti.com

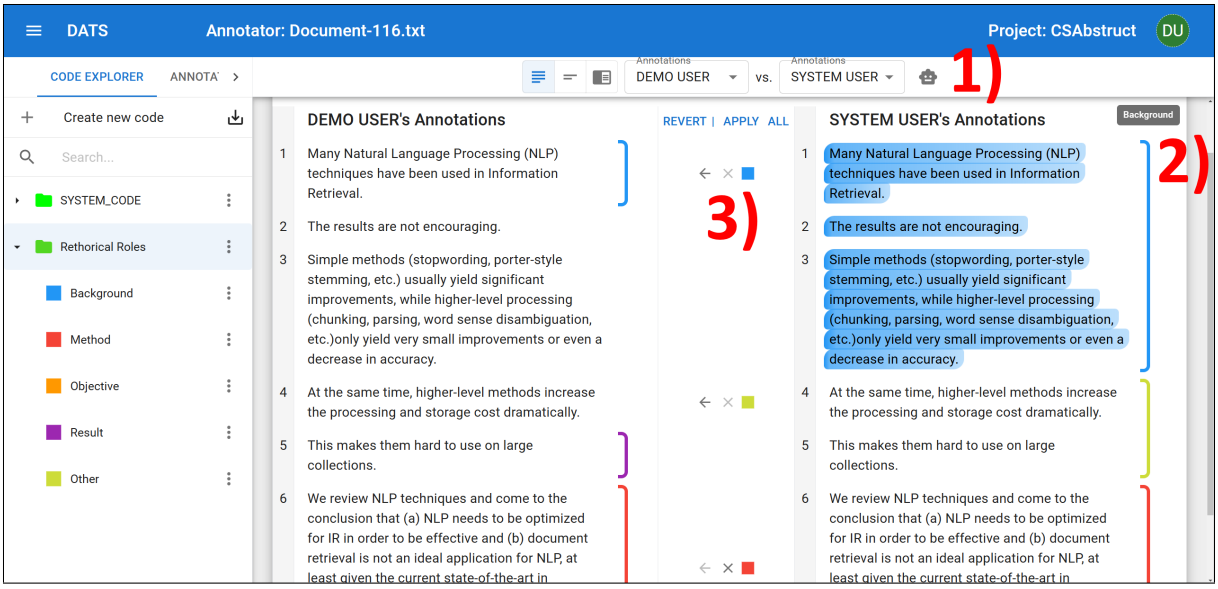

Figure 1: Sentence Annotator & Suggestions, with the taxonomy of rhetorical roles on the left and the comparison view of user and assistant annotations in the center. Button to open the Annotation Assistant Menu (1). Hovering over an annotation (2) highlights the corresponding text. Controls in the middle help to copy suggestions (3).

Imagine Alice, a social science researcher investigating the climate change debate in Germany. She is interested in understanding how different news outlets frame climate activism. To begin her research, Alice gathers a large dataset of relevant news articles and uploads them to DATS.

Before annotating, Alice and her project team brainstorm a taxonomy of frames relevant to climate activism. The taxonomy includes frames like "economic," "health," "fairness and equality," and others. They carefully define each frame and establish annotation guidelines.

With the taxonomy in place, Alice opens DATS' new sentence annotator (see Figure 1), eager to start labeling her data. But even before she annotates a single sentence, the Annotation Assistant is ready to help. Being provided with the taxonomy and its definitions, the assistant can offer initial suggestions. Alice clicks the Robot button (1) to activate the assistant, selects the frames she wants to focus on, and initiates the process. A side-by-side view appears, displaying Alice's annotations on the left and the assistant's suggestions on the right. Alice reviews these suggestions: Hovering over an annotation highlights the corresponding text (2). Hovering over the category name in the taxonomy highlights all annotated texts. She accepts the correct annotations with the help of the control buttons in the middle (3). She can easily create annotations by clicking and dragging sentences where needed.

After Alice annotates a few documents and reaches a certain threshold of annotations per frame, the assistant leverages these to refine its suggestions through few-shot prompting. The improved suggestions accelerate Alice's progress, allowing her and her team to annotate a significant portion of the dataset efficiently.

With a larger pool of annotated data, the assistant fine-tunes a specialized model on these examples, further improving the quality of its suggestions. Alice and her team begin accepting suggestions in batches, significantly boosting their productivity.

Confident in the assistant's capabilities, they let it automatically annotate the remaining documents. This process takes time. The next day, Alice reviews some of the assistant's annotations. Satisfied with the quality, she proceeds to quantitative analysis. For example, she can utilize DATS' built-in features to visualize how climate change framing evolves over time or export the annotated data for further analysis with her preferred tools.

4 Semi-automatic Sequential Sentence Classification

The workflow described in the previous section highlighted how the Annotation Assistant could empower researchers like Alice to analyze large textual datasets effectively. To achieve this, we designed the assistant with several key requirements in mind, focusing on user-centricity:

1. User Control: The user must always retain com-

plete control over the annotation process. The assistant should provide suggestions, but the final decision of accepting, rejecting, or modifying those suggestions rests with the user.

2. Optimal Suggestions: The assistant should provide the best possible suggestions at each stage of the annotation process. This requirement motivated our three-phase design, which leverages increasing amounts of user-provided data to refine its suggestions.
3. Fast Inference: Ensuring a smooth and interactive user experience within DATS requires efficient models and fast inference procedures.
4. Transparency: The assistant should provide insights into its decision-making process. Transparency is realized by attaching "Memos" to the automatic annotations containing its reasoning.

Based on the requirements, we propose a three-phase approach to AI-assisted annotation:

Phase 1: Zero-shot prompting When no annotations are available, the assistant leverages the provided taxonomy and definitions to generate initial suggestions through zero-shot prompting of LLMs. Please note that the user provides the taxonomy and category definitions, which are fully customizable.

Phase 2: Few-shot prompting After the user annotates a few documents and meets a certain threshold of labeled examples per category, the assistant transitions to few-shot prompting, refining its suggestions by incorporating the user-provided examples.

Phase 3: Fine-tuned model Once sufficient annotations are available, a sequence tagger is fine-tuned on this data, further improving the accuracy of the assistant's suggestions.

Notably, the thresholds for transitioning between phases (e.g., what constitutes "few" or "sufficient" annotations) are configurable. They can be adjusted based on the specific characteristics of the dataset and the user's preferences. Further, the reasoning is currently only supported during Phase 1 and 2, as the LLM is instructed to provide a reason alongside its prediction.

5 Benchmarking Sequential Sentence Classification

This benchmark evaluates LLMs and sequence taggers on sequential sentence classification tasks. As outlined in the previous section, both are crucial components of our proposed three-phase approach. We aim to identify the most suitable configuration for integrating practical annotation assistance into DATS. To this end, we carefully select datasets for sequential sentence classification with varying domains and tasks.

Sequential Sentence Classification is a sequence tagging task similar to, for example, named entity recognition, but instead of tagging tokens, sentences are tagged. Hence, every sample of a typical sequential sentence classification task consists of a list of sentences and tags assigned to each sentence. In this benchmark, we only investigate the single-label classification task, i.e., every sentence is classified into one of many classes.

5.1 Models

In alignment with the open-source principles of our Discourse Analysis Tool Suite, we exclusively utilize open-source and open-licensed models for the benchmark. This design choice is motivated by the data privacy considerations of our primary users – universities and researchers – who often work with sensitive data. Local execution of models is essential to ensure data confidentiality.

For zero- and few-shot experiments, we evaluate three state-of-the-art decoder-only language models: Llama 3.1 (8B parameters), Gemma 2 (9B parameters) (Gemma Team, 2024), and Mistral NeMo (12B parameters) (Mistral AI Team 2024, Jiang et al. 2023). These variants are readily deployable in resource-constrained environments due to comparatively small parameter size and availability in half-precision.

Llama 3.1, released under the Llama 3 Community License by Meta AI, is trained on a massive, multilingual dataset of approximately 15 trillion tokens and has a context window of 128,000 tokens. Gemma 2 is a lightweight model from Google built upon the same technology as their Gemini models. It is trained primarily on English web documents, code, and mathematical text, encompassing 8 trillion tokens. Mistral NeMo, developed by Mistral AI in collaboration with NVIDIA, is released under the Apache 2.0 license. It is trained on multilingual and code data and supports a context window of up to 128,000 tokens. We only employ instruct fine-tuned models in half-precision (FP16) and set the context window to 32K tokens.

For fine-tuning experiments, we evaluate a SentenceTransformer-based sequence tagger. We employ an established model architecture (Huang

Table 1: Performance of prominent LLMs in zero-shot (0), few-shot (2, 4), and fully fine-tuned (All) settings on five sequential sentence classification datasets. F1-score and accuracy (Acc) are reported. ↑ indicates overall improvement compared to the previous step, ↓ an overall decrease. The SOTA row lists the best performing approaches to date: 1) SciBERT + MLP by Cohan et al. (2019) 2) Neural Semi-Markov CRFs by Yamada et al. (2020) 3) Contrastive Training of BigBird by Laboulaye (2021) 4) Multi-modal features (text, image, audio) + Graph Neural Network by Shou et al. (2024) 5) Graph Neural Network by Liang et al. (2022)

Dataset		CSAbstruct		Pubmed200k		CoarseDiscourse		EmotionLines		DailyDialog	
Model	Shot	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Llama 3.1	0	35.68	49.67	33.91	60.27	27.64	23.59	23.19	28.61	26.16	30.60
Gemma 2	0	**40.02**	55.08	**52.88**	**73.32**	**33.28**	**32.36**	**33.26**	**46.92**	**31.32**	**40.83**
Mistral	0	39.39	**56.63**	45.85	70.98	31.64	30.73	18.68	27.64	26.25	31.14
Llama 3.1	2 ↑	37.71	51.96	48.43	71.54	27.54	23.95	23.53	28.94	27.86	33.39
Gemma 2	2 ↓	**45.82**	**60.79**	**56.93**	72.02	18.21	11.77	**29.86**	**33.06**	**30.85**	**37.51**
Mistral	2 ↑	41.35	57.60	53.95	**76.06**	**31.94**	**31.47**	22.49	28.61	27.46	36.59
Llama 3.1	4 ↓	35.09	50.63	49.22	69.93	26.79	23.11	22.19	27.98	25.83	30.02
Gemma 2	4 ↓	**45.78**	57.89	50.43	51.93	13.59	08.16	**26.85**	29.55	22.60	26.44
Mistral	4 ↑	44.57	**60.27**	**54.04**	**76.95**	**32.85**	**31.96**	23.99	**32.61**	**27.80**	**37.86**
allmpnetv2	All	37.43	61.23	72.89	85.78	29.15	41.48	26.06	47.14	22.03	51.81
NV-Embed	All	**55.65**	**71.98**	**81.72**	**90.20**	**42.40**	**51.05**	**34.86**	**52.89**	**31.77**	**57.67**
SOTA	All	83.1[1]	–	93.1[2]	–	84.0[3]	–	69.90	68.7[4]	64.2[5]	–

et al., 2015; Panchendrarajan and Amaresan, 2018) consisting of four layers: (1) SentenceTransformer to compute sentence embeddings, (2) BiLSTM (Hochreiter and Schmidhuber, 1997) to capture dependencies between sentences, (3) a linear layer to project from the BiLSTM's hidden dimension to the number of tags, and (4) a CRF (Lafferty et al., 2001) layer to model dependencies between tags. We compare two embedding models: The default *all-mpnet-base-v2* (Song et al., 2020) with NV-Embed (7B parameters) (Lee et al., 2024), the best-performing model on the English MTEB (as of December 2024).

5.2 Experiment construction

We conduct zero- and few-shot experiments using a single, clear prompt across all datasets, avoiding extensive prompt engineering. The system prompt specifies the role of the model. The user prompt includes a short task description, the annotation guidelines (i.e., taxonomy categories and descriptions), and the sentences to annotate. (dataset taxonomies are listed in Appendix A, prompts are detailed in Appendix B). We use structured generation enforcing models to generate a JSON response containing *sentence id*, *classification*, and optional *reasoning*. This eases the parsing of model responses but may impact the performance slightly.

Few-shot examples are taken from the training splits and computed in advance using a $k-2k$ sampler similar to the implementation of Ding et al. (2021). For example, in $k = 2$, we find the minimum number of training samples required to satisfy the following constraint: every category must occur at least two but no more than four times. This means that k does not correspond to a number of documents or sentences. Instead, the number of fully-annotated training documents provided as few-shot examples depends on the dataset.

We conduct fine-tuning experiments by training the sequence tagger for at most 100 epochs on the datasets' training split. We use AdamW optimizer, gradient clipping, early stopping with three epochs patience, and freeze the embedding model's weights. The BiLSTM has one layer with a hidden dimension of 256.

All experiments were conducted on a single A100 GPU (80GB) and repeated three times per configuration. The results are averaged across runs to mitigate fluctuations. We only report results on the test sets.

5.3 Datasets

This benchmark encompasses various domains, including online discussions, conversational dialogues, as well as scientific and medical abstracts.

It covers the tasks of discourse act classification, emotion recognition, and rhetorical role labeling.

Coarse Discourse Corpus (Zhang et al., 2017) comprises approximately 9,000 Reddit threads with over 100,000 comments annotated for discourse acts. Each comment is classified into one of ten categories, such as Question, Answer, or Disagreement.

CSAbstruct (Cohan et al., 2019) consists of 2,000 computer science research abstracts from Semantic Scholar. Each sentence is annotated with one of five rhetorical roles, like background, objective, and method, cf. example in Figure 1.

PubMed 200k (Dernoncourt and Lee, 2017) is a dataset comprising approximately 200,000 abstracts from randomized controlled trials in the medical domain. Each sentence is annotated with one of five rhetorical roles similar to CSAbstruct.

EmotionLines (Hsu et al., 2018) is a dataset of 2,000 dialogues, comprising 29,245 utterances from Friends TV scripts and Facebook Messenger conversations. Each utterance is annotated with one of eight emotions, including Surprise, Sadness, and Joy.

DailyDialog (Li et al., 2017) is a dataset of 13,118 multi-turn dialogues, totaling approximately 100,000 utterances, designed to reflect everyday conversations across diverse topics. Each utterance is annotated with emotion labels similar to EmotionLines.

5.4 Results

The benchmark results are presented in Table 1. Gemma 2 exhibits excellent zero-shot performance, but its performance generally decreases with the addition of few-shot examples. Llama 3.1 consistently performs the worst in zero- and few-shot settings, with few-shot examples having a negligible impact. Mistral demonstrates consistently good performance, improving with more few-shot examples. However, even in the 4-shot scenario, it often fails to surpass Gemma 2's zero-shot performance. Exploratory experiments with more examples in the few-shot learning scenario were inconclusive and could not consistently improve over the 4-shot setting. As expected, the fully fine-tuned sequence tagger based on NV-Embed achieves the best results across all datasets, with improvements of up to 15 points in F1-score (Pubmed) and 19 points in accuracy (CoarseDiscourse) compared to the best few-shot results. Notably, rhetorical role labeling appears to be an easier task than emotion detection and discourse act classification. The relative simplicity of rhetorical role labeling stems partly from the lower number of classes compared to other tasks. The taxonomies of CSAbstruct and PuMmed utilize 5 classes, while DailyDialog, EmotionLines, and CoarseDiscourse employ 8, 9, and 10 classes.

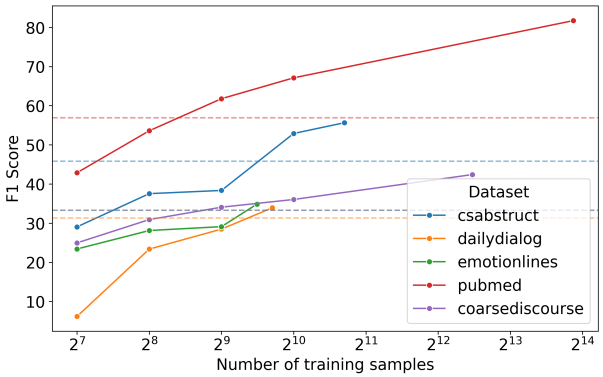

Figure 2: Evaluation of NV-embedd-based sequence tagger with increasing number of training examples. The dashed line denotes the highest F1-score achieved by LLMs in few-shot settings.

5.5 Discussion

While Gemma 2's zero-shot performance and the overall performance of the fully fine-tuned models are encouraging, the few-shot results are less satisfactory. We will need to investigate alternative few-shot learning approaches that scale more effectively. Regarding DATS integration, we selected Mistral because it is the only model in our benchmark that consistently improves with increasing examples, which was our desired property. We hypothesize that this trend will continue with more annotated data, allowing the assistant to provide increasingly accurate suggestions as users progress with their analysis.

5.6 Transitioning Phase 2 → 3

We aim to identify thresholds for transitioning from Phase 2 (few-shot prompting) to Phase 3 (fine-tuning a sequence tagger). The optimal threshold is the number of training examples the sequence tagger requires to surpass LLMs in the few-shot setting. For this, we train the NV-embed-based sequence tagger on an increasing number of fully annotated training examples: 128, 256, 512, 1024, and the whole training split. Figure 2 shows the

performance improvements of the sequence tagger as the number of training examples increases. The dashed lines correspond to the best F1-score achieved by LLMs in few-shot settings, which were evaluated before as part of this benchmark. Hence, the minimum number of fully annotated documents that is needed to train a sequence tagger superior to LLMs is given by the intersection of the dashed line with the corresponding solid line. For example, approximately 380 fully annotated documents are necessary to surpass LLMs in the PubMed dataset, about 770 documents for CSAbstruct, 500 for CoarseDiscourse, and 700 documents for DailyDialog and EmotionLines. As this number is highly dependent on the dataset, we decided to make this threshold for transitioning between phases configurable in the User Interface, allowing users to adjust it based on their project and preferences.

6 Integration within DATS

The Annotation Assistant is built using React for the frontend and Ollama, FastAPI, and Celery[4] for the backend. Celery handles background job processing, including fine-tuning and inference of the sequence tagger model. It ensures that assistance tasks run without interrupting user workflow. Ollama hosts the Mistral NeMo model. It also handles the structured generation in combination with Pydantic. We reuse the prompts utilized in the benchmark. DATS already offers a semantic similarity search based on sentence embeddings, which are computed during import. We reuse these pre-computed sentence embeddings to significantly reduce the training and inference time of the sequence tagger model.

7 Conclusion

This paper explored a three-phase approach to AI-assisted annotation to enhance qualitative data analysis workflows, focusing on sequential sentence classification within our open-source Discourse Analysis Tool Suite. We designed a benchmark and evaluated the performance of three prominent open-source LLMs and sequence tagging models based on SentenceTransformers. Our findings demonstrate the effectiveness of our proposed three-phase Annotation Assistant. Performance steadily improves as more annotated data becomes available when using Mistral. However, few-shot prompting does not scale as well as expected and may require a different approach. We integrated this assistant into DATS, providing researchers with a valuable semi-automatic sequential sentence annotation tool. We see this integration as a significant step towards democratizing access to AI for data analysis, empowering researchers with a transparent and user-controlled system that augments, rather than replaces, their expertise.

In future work, we plan to investigate parameter-efficient fine-tuning of LLMs to bridge the performance gap observed between few-shot prompting and full fine-tuning. Additionally, we aim to expand the benchmark to include German datasets, catering to DATS's user base that works with German-language texts. Code for replicating the benchmark[5], the repository of DATS[6] and a live demo are available[7].

8 Limitations & Ethics Statements

While the proposed Annotation Assistant represents a step toward enhancing qualitative data analysis workflows, it's important to acknowledge its current limitations. The Annotation Assistant is an ongoing work in progress, and we are actively exploring avenues for improvement.

One limitation is the performance gap between few-shot prompting (utilizing, e.g., 2, 4, 8, 16 examples) and full fine-tuning (using hundreds of examples). This gap can lead to a period where the assistant's suggestions do not significantly improve despite ongoing user annotations. We will explore techniques like parameter-efficient fine-tuning to address this issue.

Another limitation is the current focus on English datasets in our benchmark. It does not cater to the needs of all DATS users, particularly those working with German-language data. Finding more diverse datasets and expanding the benchmark remains a challenge.

Furthermore, the current implementation of the Annotation Assistant offers limited configurability. While we benchmarked and integrated the best-performing model, users might benefit from the ability to select and configure different models based on their specific needs and preferences.

Finally, it's crucial to understand the inherent limitations of LLMs in general. These models can exhibit biases, struggle with reasoning or common

[4]Links: React, Ollama, FastAPI, Celery

[5]https://github.com/uhh-lt/seq-sentence-classification

[6]https://github.com/uhh-lt/dats

[7]https://dats.ltdemos.informatik.uni-hamburg.de/

sense knowledge, and generate outputs that require careful review. This underscores the importance of maintaining user control and manually validating the assistant's suggestions.

References

Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Dan Weld. 2019. Pretrained Language Models for Sequential Sentence Classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3693–3699, Hong Kong, China. Association for Computational Linguistics.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a Dataset for Sequential Sentence Classification in Medical Abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-NERD: A Few-shot Named Entity Recognition Dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 3198–3213, Online. Association for Computational Linguistics.

Gemma Team. 2024. Gemma 2: Improving Open Language Models at a Practical Size. *ArXiv*, abs/2407.21783.

Evelyn Gius, Jan Christoph Meister, Malte Meister, Marco Petris, Christian Bruck, Janina Jacke, Mareike Schumacher, Dominik Gerstorfer, Marie Flüh, and Jan Horstmann. 2022. CATMA: Computer Assisted Text Markup and Analysis.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, and Mantas Mazeika et al. 2021. Measuring Massive Multitask Language Understanding. In *Proceedings of the International Conference on Learning Representations*, Online.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780.

Chao-Chun Hsu, Sheng-Yeh Chen, Chuan-Chun Kuo, Ting-Hao Huang, and Lun-Wei Ku. 2018. EmotionLines: An Emotion Corpus of Multi-Party Conversations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *ArXiv*, abs/1508.01991.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, and Devendra Singh Chaplot et al. 2023. Mistral 7B. *ArXiv*, abs/2310.06825.

Roland Laboulaye. 2021. Turn of Phrase: Contrastive Pre-Training for Discourse-Aware Conversation Models. Master's thesis, Brigham Young University.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. *ArXiv*, abs/2405.17428.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Chen Liang, Jing Xu, Yangkun Lin, Chong Yang, and Yongliang Wang. 2022. S+PAGE: A Speaker and Position-Aware Graph Neural Network Model for Emotion Recognition in Conversation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 148–157, Online only. Association for Computational Linguistics.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, and Dilara Soylu et al. 2023. Holistic Evaluation of Language Models. *Transactions on Machine Learning Research*, 1525(1):140–146.

Mistral AI Team. 2024. Mistral NeMo. `https://mistral.ai/news/mistral-nemo/`.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Rrubaa Panchendrarajan and Aravindh Amaresan. 2018. Bidirectional LSTM-CRF for Named Entity Recognition. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*, Hong Kong. Association for Computational Linguistics.

Florian Schneider, Tim Fischer, Fynn Petersen-Frey, Isabel Eiser, Gertraud Koch, and Chris Biemann.

2023. The D-WISE tool suite: Multi-modal machine-learning-powered tools supporting and enhancing digital discourse analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 328–335, Toronto, Canada. Association for Computational Linguistics.

Yuntao Shou, Wei Ai, Jiayi Du, Tao Meng, Haiyan Liu, and Nan Yin. 2024. Efficient Long-distance Latent Relation-aware Graph Neural Network for Multi-modal Emotion Recognition in Conversations. *ArXiv*, abs/2407.00119.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, and Abubakar Abid et al. 2023. Beyond the imitation game: quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023(5):1–95.

Anselm Strauss and Juliet Corbin. 1990. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. SAGE Publications, Inc.

Anselm Strauss, Juliet Corbin, Solveigh Niewiarra, and Heiner Legewie. 1996. *Grounded Theory: Grundlagen Qualitativer Sozialforschung*. Beltz, Psychologie-Verlag-Union Weinheim.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA. Curran Associates Inc.

Kosuke Yamada, Tsutomu Hirao, Ryohei Sasano, Koichi Takeda, and Masaaki Nagata. 2020. Sequential Span Classification with Neural Semi-Markov CRFs for Biomedical Abstracts. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 871–877, Online. Association for Computational Linguistics.

Amy Zhang, Bryan Culbertson, and Praveen Paritosh. 2017. Characterizing Online Discussion Using Coarse Discourse Sequences. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 357–366, Montreal, Quebec, Canada. PKP Publishing Services Network.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can Large Language Models Transform Computational Social Science? *Computational Linguistics*, 50(1):237–291.

A Dataset details

Tables 3 - 6 list the categories and their definitions of all datasets used in the benchmark. We used the original authors' definitions from their papers or annotation guidelines whenever possible. In all other cases, we wrote the definitions ourselves. The categories and definitions are used in the zero- and few-shot prompts to give the LLM context beyond the labels and help it solve the task correctly.

B Prompts

Table 2 lists the system and user prompt templates used in our benchmark and implementation. The system prompt contains details about the LLM's role and instructions about the output formatting. The user prompt template briefly explains the task, lists the annotation guidelines consisting of categories and definitions, and includes the document as a numbered list of sentences. Finally, the key constraint of using categories of the annotation guideline is reiterated.

In few-shot prompting, we provide the model with a series of examples. This is done by presenting the following sequence:

1. System Prompt
2. Few-Shot Examples
   (a) Modified User Prompt: A version of the user prompt without the annotation guidelines.
   (b) Correct Answer: The correct labels for the sentences, formatted according to the answer template.
3. User Prompt: The full user prompt, including the annotation guidelines

Table 2: The prompts used in the benchmark and implementation. < > are placeholders for dataset-dependent input.

System Prompt

You are a professional annotator specialized in annotating sentences of a document with the help of annotation guidelines. You strictly adhere to the guidelines and follow the desired output format.

Output Format:
You MUST answer in this JSON format, but the reason is optional:

```
[
  {
    "text_id": 1,
    "reason": "The sentence provides context for the research.",
    "category": "background"
  },
  {
    "text_id": 2,
    "reason": "The sentence presents the research findings.",
    "category": "result"
  },
...
]
```

User Prompt

Please annotate each sentence of the following document.

Annotation Guidelines:
<annotation_guidelines>

Document:
<document>

Remember to annotate every provided sentence.
You MUST use the categories provided in the Annotation Guidelines!

Answer Template

<text_id>: <classification>
<text_id>: <classification>
<text_id>: <classification>
...

Table 3: The category system used in the CSAbstruct (Cohan et al., 2019) dataset.

Category	Description
Background	Provides context or previous knowledge relevant to the research topic. Think of it as setting the stage for the study.
Method	Describes the procedures and techniques used in the research. This includes the study design, data collection, and analysis methods.
Objective	States the main goal or purpose of the research. This could be discussion, analysis, limitations, or concluding remarks.
Result	Presents the findings or outcomes of the research. This often includes statistical data, tables, and figures.
Other	Any sentence that doesn't fit into the above categories.

Table 4: The category system used in the Pubmed200k (Dernoncourt and Lee, 2017) dataset.

Category	Description
Background	Provides context or previous knowledge relevant to the research topic. Think of it as setting the stage for the study.
Methods	Describes the procedures and techniques used in the research. This includes the study design, data collection, and analysis methods.
Objective	States the main goal or purpose of the research.
Results	Presents the findings or outcomes of the research. This often includes statistical data, tables, and figures.
Conclusions	Summarizes the key findings of the research and draw inferences from those findings. They provide closure to the abstract, summarizing the overall contribution of the research.

Table 5: The category system used in EmotionLines (Hsu et al., 2018) and DailyDialog (Li et al., 2017) datasets. EmotionLines also includes the Non-Neutral label.

Category	Description
Fear	A feeling of apprehension or dread in response to a perceived threat or danger. It can range from mild anxiety to intense terror.
Disgust	A feeling of revulsion or aversion, often triggered by something perceived as unpleasant, unsanitary, or morally offensive.
Excited	A state of heightened arousal and positive anticipation. It often involves feelings of enthusiasm, eagerness, and energy.
Anger	A feeling of intense displeasure or hostility, often triggered by a perceived wrong or injustice. It can manifest as irritation, frustration, rage, or fury.
Surprise	A brief emotional state in response to an unexpected event. It can be positive, negative, or neutral, depending on the nature of the surprise.
Sadness	A feeling of sorrow, grief, or disappointment. It can range from mild melancholy to intense despair.
Joy	A feeling of happiness, contentment, or pleasure. It can manifest as amusement or love.
Neutral	A state of emotional balance or equilibrium, where no particular emotion is dominant.
Non-Neutral	Other or multiple of the above emotions are present

Table 6: The category system used in the Coarse Discourse (Zhang et al., 2017) dataset.

Category	Description
Question	A comment with a question or a request seeking some form of feedback, help, or other kinds of responses. While the comment may contain a question mark, it is not required. For instance, it might be posed in the form of a statement but still soliciting a response. Also, not everything that has a question mark is automatically a Question. For instance, rhetorical questions are not seeking a response. Relation: This comment might be the first in a thread and have no relation to another comment. Or, it could be a clarifying or followup Question linking to any prior comment.
Answer	A comment that is responding to a Question by answering the question or fulfilling the request. There can be more than one Answer responding to a Question. Relation: An Answer is always linked to a Question.
Announcement	A comment that is presenting some new information to the community, such as a piece of news, a link to something, a story, an opinion, a review, or insight. Relation: This comment has no relation to a prior comment and is always the initial post in a thread.
Agreement	A comment that is expressing agreement with some information presented in a prior comment. It can be agreeing with a point made, providing supporting evidence, providing a positive example or experience, or confirming or acknowledging a point made. Relation: This comment is always linked to a prior comment to which it is agreeing.
Appreciation	A comment that is expressing thanks, appreciation, excitement, or praise in response to another comment. In contrast to Agreement, it is not evaluating the merits of the points brought up. Comments of this category are more interpersonal as opposed to informational. Relation: This comment is always linked to a prior comment for which it is expressing appreciation.
Disagreement	A comment that is correcting, criticizing, contradicting, or objecting to a point made in a prior comment. It can also be providing evidence to support its disagreement, such as an example or contrary anecdote. Relation: This comment is always linked to a prior comment to which it is disagreeing.
Negative reaction	A comment that is expressing a negative reaction to a previous comment, such as attacking or mocking the commenter, or expressing emotions like disgust, derision, or anger, to the contents of the prior comment. This comment is not discussing the merits of the points made in a prior comment or trying to correct them. Relation: This comment is always linked to a prior comment to which it is negatively reacting.
Elaboration	A comment that is adding additional information on to another comment. Oftentimes, one can imagine it simply appended to the end of the comment it elaborates on. One can elaborate on many kinds of comments, for instance, a questionasker elaborating on their question to provide more context, or someone elaborating on an answer to add more information. Relation: This comment is always linked to a prior comment upon which it is elaborating.
Humor	This comment is primarily a joke, a piece of sarcasm, or a pun intended to get a laugh or be silly but not trying to add information. If a comment is sarcastic but using sarcasm to make a point or provide feedback, then it may belong in a different category. Relation: At times, this comment links to another comment but other times it may not be responding to anything.
Other	A comment that does not fit any of the previous definitions.