FRI Natural language processing course 2024
Project report

University *of Ljubljana*
Faculty *of Computer and*
Information Science

Qualitative Research on Discussions - text categorization

Ana Petrova, Žan Korošak, and Bojan Ilić

Abstract

The abstract goes here. The abstract goes here.

Keywords
discourse analysis, text categorization, language model

Advisors: Slavko Žitnik

Introduction

In the evolving landscape of social science research, qualitative discourse analysis emerges as a pivotal methodology for understanding the complexities of human interaction. This intricate process involves the categorization of text within discussions, demanding a deep comprehension of context, participant perspectives, and the linkage between them that weave through conversations. Traditionally, this task has been the domain of human coders, whose role in ensuring inter-rater reliability is both crucial and labor-intensive. The development of large language models (LLMs) introduces a transformative potential for automating and enhancing the reliability of such qualitative analyses.

This paper researches the development and application of a novel approach to categorize postings in online discussions, using a case study centered around the corpus of an online dialogue about the story ”The Lady, or the Tiger?”. Leveraging a dataset coded with high inter-rater reliability and an accompanying codebook, our project aims to construct and train a language model with an ability to address this complex coding task. The goal is to achieve a model that not only demonstrates high reliability but also generalizes effectively across various online discussion contexts.

Our methodology is anchored in a comprehensive literature review that explores existing discourse and dialogic analysis frameworks, with a particular focus on the intersection of these fields with natural language processing (NLP). This review lays the groundwork for understanding the coding criteria and methodologies that have shaped the field. Following this, we begin a detailed examination of the provided coded discourse dataset, focusing on understanding its complexities and the challenges associated with coding it.

The core of our research involves the intricate process of building and fine-tuning LLMs. This process is not merely technical; it requires a delicate understanding of the discourse context, the dynamics between participants, and the interplay of ideas within the discussion. The use of high-performance computing (HPC) is crucial in this phase, enabling the processing and analysis of complex datasets. Performance evaluation forms a critical component of our methodology, involving a comparison of our model's performance with that of human coders and other computational models. This iterative process of comparison and refinement is crucial for enhancing the model's accuracy and reliability.

A novel aspect of our research is the use of a separately fine-tuned LLM to generate explanations for the model's categorization decisions. This not only adds a layer of transparency to the model's workings but also provides valuable insights into its decision-making processes.

Related works

The article by Devlin et al. [1] introduces BERT, a novel language representation model acronym for Bidirectional Encoder Representations from Transformers. Unlike previous models, BERT is uniquely designed to pre-train deep bidirectional representations from unlabeled text, capturing both left and right context in all layers. This allows for fine-tuning with minimal modifications to achieve state-of-the-art performance on various tasks, including question answering and language inference. BERT exhibits simplicity in concept and remarkable empirical effectiveness, evidenced by significant improvements on eleven natural language processing tasks.

In the study ”How to Fine-Tune BERT for Text Classification” by Sun et al. [2], the researchers examine various methods for optimizing the BERT model to enhance its effectiveness in text classification tasks. They perform detailed experiments across several well-known datasets, establishing new performance benchmarks. Their proposed methodology includes a pre-fine-tuning phase using multitask learning and targeted pre-training with in-domain data. This preparatory step is followed by specific fine-tuning adjustments tailored to the text classification task at hand. The paper offers a comprehensive exploration of factors such as the selection of neural network layers, addressing catastrophic forgetting, and managing long text inputs, all of which significantly influence the performance of the model. This approach highlights the adaptability of BERT in managing a range of text classification problems and provides a robust framework for leveraging deep learning models in natural language processing tasks.

Ye et al. [3] focused on prompt engineering, a critical task for optimizing the performance of large language models on customized tasks. It highlights the complexity involved in analyzing model errors, identifying deficiencies in prompts, and articulating tasks clearly. While existing research suggests that large language models can be meta-prompted for automatic prompt engineering, the authors argue that this approach is limited by a lack of guidance for nuanced reasoning. To address this limitation, the authors propose PE2, a method that enhances the meta-prompt with detailed descriptions, context specification, and a step-by-step reasoning template. PE2 demonstrates remarkable versatility across various language tasks, surpassing competitive baselines on tasks such as MultiArith and GSM8K. Additionally, the method excels in making targeted prompt edits, rectifying erroneous prompts, and generating multi-step plans for complex tasks.

Corpus analysis

The main dataset, sourced from an online discussion on ”The Lady, or the Tiger,” is accessible on our Github repository [1]. Each entry includes the user’s pseudonym and message, categorized with high inter-rater reliability. Entries were labeled primarily by *Discussion type*, aided by provided definitions and examples. Coders also marked messages with classifications like *Dialogic spell*, *Uptake*, *Question*, and *Pivot*. As a primary classification class, we used column *R2 Discussion Type*. The distribution graph (Figure 1) showcases the popularity of each discussion type. Notably, *Seminar* dominates, defined as discussions on content meaning or interpretation, followed by *Deliberation*, focusing on content-related decision-making.

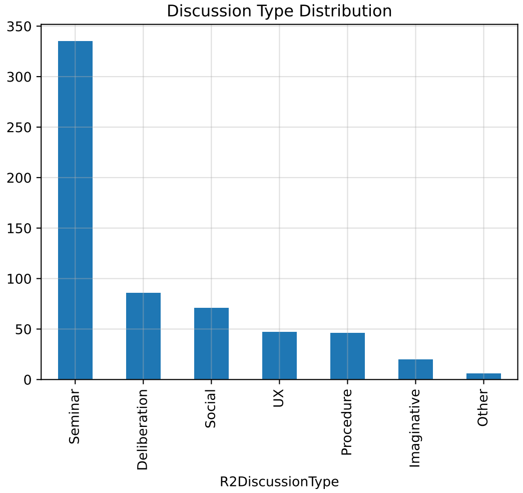

Figure 1. Distribution of different discussion types. We can see that ’Seminar’ is the predominant type, which is expected from the definition and the type of discourse. The least common type is ”Imaginative” which ”places the learner in the discussion as an active participant”.

Methods

For the initial phase of our project, we addressed the classification challenge. We chose to utilize the *R2DiscussionType* column as it consistently defines the decision task and serves as the primary class for classification by test subjects. As a benchmark, we employed a pre-trained BERT model (*bert-base-cased*) to classify Discussion types within our dataset. Implementation was carried out using the PyTorch library in Python, which gave us access to Trainer API that we used to fine-tune the pre-trained BERT model. Baseline BERT model was trained with 3 epochs, utilizing a batch size of 8, and employing the standard learning rate of 2×10^{-5}.

We divided our dataset into three parts: train, test and validation. In the training dataset we have 427 samples or 70% of the dataset, and the test and validation datasets both have 92 samples or 15% of the dataset each. Due to the class imbalance in our dataset, certain less represented classes have only a few samples in the divided datasets. The test dataset is used as an evaluation dataset during the training phase, and we use the validation dataset as a final score of the model.

Using the Trainer API, we could also fine-tune the BERT model by increasing number of epochs. To setup optimizers, we used Adam optimizer by PyTorch and tested a scheduler with implemented *ReduceLROnPlateau*, to reduce learning rate when a metric stops improving.

We also tried using DistilBERT[4], which is a lightweight BERT model fine-tuned for text classification.

Our current plan of work contains several approaches:

- Implement a different LLM, which will classify discussion types and try to explain how the model made the decision.
- Try to use prompt engineering.

[1] https://github.com/UL-FRI-NLP-2023-2024/ul-fri-nlp-course-project-azb

- Combine two or more models (ensemble learning).
- Further exploration of the dataset, possibly filtering out ”bad” samples.
- Fine-tune the large BERT model.

We plan to evaluate all our models on a separate discourse dataset, created in a similar research with test subjects being high school students.

In our exploration of advanced NLP techniques for text categorization, we implemented various methodologies centered around the LLAMA model [5] and prompt engineering, aiming to enhance the model’s ability to perform few-shot prompting. In order to choose relevant prompts, we experimented with BERT embeddings followed by K-means and hierarchical clustering techniques, which initially did not meet the expected clarity in categorization. Most of the relevant examples chosen by these algorithms were short statements without enough context. Subsequently, we shifted our strategy to use TF-IDF vectorization instead of BERT embeddings. This approach was taken to retain more textual context. However, it didn’t prove to be essentially effective in clustering and categorization. The primary limitation we encountered comes from the dataset’s composition—specifically, its imbalance and the prevalence of context-poor texts for some categories significantly impacted the LLAMA model’s ability to distinguish between categories accurately. Additionally, the names of the categories were also quite abstract for the model to extract features for categorization. This experiment highlighted the critical relationship between dataset quality and the performance of machine learning models in text-based categorization tasks.

Results

For evaluating our BERT models, we will use accuracy as an evaluation metric to determine how well it is performing.

Model	Epochs	Batch size	Test accuracy	Validation accuracy
bert-base-cased	3	4	0.728	0.652
bert-base-cased	3	8	0.75	0.685
bert-base-cased	10	8	0.77	0.696
distil-base-uncased	10	8	0.728	0.641
distil-base-uncased	50	16	0.77	0.641
bert-large-uncased	10	8	0.72	0.72
bert-large-uncased	20	16	0.75	0.72

Currently our best performing model is the large BERT model (*bert-large-uncased*), which has the best accuracy score on the validation dataset. The accuracy is greatly affected by the batch size and the number of epochs in which the model was trained.

Discussion

Acknowledgments

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[2] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune BERT for text classification?, 2020.

[3] Qinyuan Ye, Maxamed Axmed, Reid Pryzant, and Fereshte Khani. Prompt engineering a prompt engineer, 2024.

[4] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[5] The FAIR team of Meta AI. Llama-7b converted to work with transformers/huggingface., 2023.