Assignment 2

Tancredi Bosi and **Giovanni Grotto**
Master's Degree in Artificial Intelligence, University of Bologna
{ tancredi.bosi, giovanni.grotto }@studio.unibo.it

Abstract

This report explores the performance of two pre-trained large language models (LLMs), **Mistral v3** and **Llama v3.1**, for classifying tweets as containing sexism or not, using zero-shot and few-shot prompting methods. Both models were tested without further training, and evaluation metrics included accuracy, F1-score, precision, recall, and confusion matrices. The results indicated that Mistral showed a significant improvement from zero-shot to few-shot, while Llama outperformed Mistral in zero-shot prompting but showed a slight decline in few-shot performance.

1 Introduction

There are various methods to approach this task. The state-of-the-art methods rely on transformers like BERT, which leverage bidirectional attention to capture the contextual meaning of tweets. However, these methods come with the downsides of being computationally intensive to train and lacking interpretability.
In this report, we will try to solve this task by making pre-trained LLMs choose the label of the classification problem.
We will compare these models on the same dataset using zero-shot prompting and few-shot prompting; then we will conduct analyses on the generated responses and on the general behavior of the LLMs. Additionally, we will investigate how these models perform on the Assignment 1 dataset, comparing the results with the models used in the previous experiment.

2 System description

In this assignment we used some pre-trained LLM's, specifically: Mistral v3 and Llama v3.1.

Both models were used to classify tweets, but no further training was performed. Therefore, there was no gradient step in this assignment. The only techniques employed were zero-shot prompting and few-shot prompting, where we included task examples directly in the prompt to guide the models' behavior.

The models were quantized using 4-bit quantization (with double quantization enabled), and each model received input that was tokenized using its respective tokenizer, which is important because different LLMs have different tokenization processes, meaning each model requires its specific tokenizer to correctly process the input, while the compute precision was set to bfloat16 for efficient computation.

3 Experimental setup and results

In the zero-shot prompting configuration, the prompt given to the LLMs is composed of two parts: a system content (*You are an annotator of sexism detection.*) and a user content (*Your task is to classify input text as containing sexism or not. Respond only YES or NO. TEXT: {text} ANSWER:*), where the "{text}" tag is filled with the tweet's text.
In the few-shot configuration, the prompt is the same, with an additional tag "{examples} to be filled with some demonstrations. We selected demonstrations from the provided dataset and decided to load each prompt with two randomly chosen examples from each class, ensuring that the examples were different for each prompt.
The evaluation metrics used to compare the two models are mainly accuracy and fail-ratio (how frequent the LLM fails to follow instructions and provides incorrect responses that do not address the classification task). Additionally, we considered precision and recall per class, F1-score and the confusion matrices for a better understanding of the behavior of the models. A more detailed analysis of the differences in the results can be seen in 1.
The fail-ratio metric resulted to be always 0, so we decided not to show it in the table.

Metric	Mistral ZS	Llama ZS	Mistral FS	Llama FS
Accuracy	0.59	0.65	**0.73**	0.64
Macro F1	0.52	0.63	**0.73**	0.60
Class 0 Precision	**0.89**	0.78	0.81	0.85
Class 0 Recall	0.21	0.42	**0.62**	0.34
Class 1 Precision	0.55	0.60	**0.69**	0.59
Class 1 Recall	**0.97**	0.88	0.85	0.94

Table 1: Mistral and Llama performance metrics (zero-shot and few-shot prompting).

4 Discussion

Since the error analysis also considered the performance of the two LLMs using the Assignment 1 dataset (referred to as A1 from now on), all subsequent considerations will take into account the performance of the methods on both A1 and A2.

Discussion of quantitative results

Mistral Model: Showed a significant performance boost from zero-shot (59.3%) to few-shot (73.3%) on dataset A2, indicating that providing task examples helps the model make better predictions. Similarly, on dataset A1, Mistral's accuracy improved from 76.5% to 81.1%, suggesting that few-shot prompting can effectively boost performance.

Llama Model: Outperformed Mistral in zero-shot on A2 (65% vs. 59.3%) and had same performance on dataset A1 (76.2% vs. 76.5%), but slightly decreased in few-shot (64% vs. 65% on A2, 72.7% vs. 76.2% on A1).
This shows that Llama has an inherent advantage in zero-shot tasks, likely due to its pre-training or architecture. However, in the few-shot configuration, Llama's performance slightly declined. This decrease may be due to the model's difficulty in handling longer prompts or the additional context provided by examples, which could have led to confusion or overfitting to the specific task details.

Error analysis

The error analysis conducted focused not only on comparing the performance between the two LLMs, but also on the quality of the generated text. It explored specific generation errors and compared the methods used in this assignment with those from the previous assignment.

Undesired Outputs: Both models rarely provided additional explanations beyond the required "YES" or "NO" responses, particularly in the few-shot configurations. This behavior can be attributed to the complexity of longer prompts, which included both the task description and multiple examples. Such prompts may have led the models to over-interpret the instructions, resulting in verbose outputs rather than the concise answers expected.

A1 Comparison: As can be seen in 2, a transformer specifically trained for sexist tweet classification outperformed both Mistral and Llama, suggesting that smaller, task-specific models may be more effective when trained or fine-tuned for a particular task or dataset.
It is also interesting to notice, though, that Mistral, with few-shot prompting, was able to get very close to the performance of the transformer, without any need for training.

Model	Accuracy (%)
Baseline	73
Model 1	74
Transformer	**84**
Mistral ZS	73
Mistral FS	81
Llama ZS	76
Llama FS	72

Table 2: Model Comparison by Classification Accuracy

5 Conclusion

As expected, both Mistral and Llama have good performances with a well structured prompt, also with zero-shot prompting, reaching, respectively, 59% and 65% accuracy.
The use of few-shot prompting led to different results for the two models. The Mistral model greatly improved the performances (73% accuracy), showing that some demonstrations can help the model to answer better. The Llama model, instead, kept the same performances (64% accuracy), maybe due to the fact that this model have some difficulties when the context length increases.

References