Metaphor and Figurative Language Detection Using LLMs

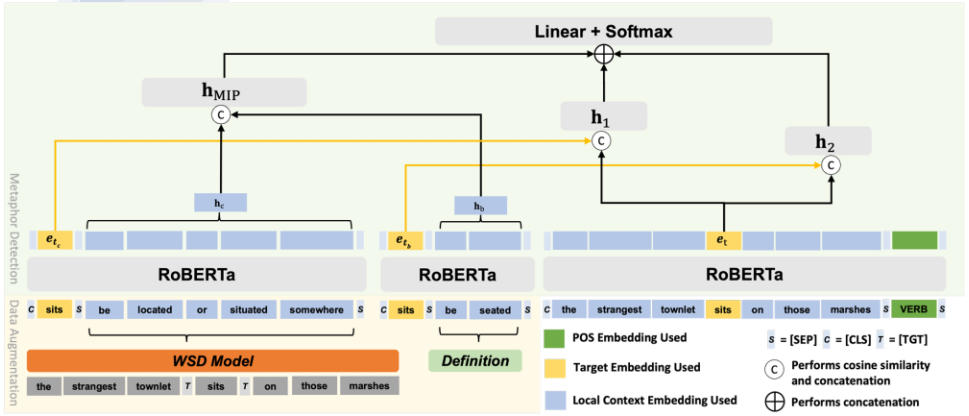
Preda Alexandru-Florin

Introduction

This project explores how large language models (LLMs) can be used to detect metaphorical and figurative language. Using DeepSeek-R1:14b, the model was tested on the VUA metaphor detection dataset. The study aims to assess the model's effectiveness in classifying words as metaphorical or literal.

State of the Art

Previous approaches relied on rule-based systems and custom linguistic features. Modern approaches use transformer-based models like BERT and GPT. ContrastWSD, a recent model, enhances metaphor detection by integrating Word Sense Disambiguation with contextual analysis.



Methodology

- Used the metaphor-detection-vua-wsd-augmented dataset.
- Tested the DeepSeek-R1:14b model via API calls using the Ollama server.
- Prompts were structured to include sentence, target word, part of speech, word sense, and definition.
- The model was asked if the target word was metaphorical, returning 'Yes' or 'No'.
- A confusion matrix was computed to evaluate performance.

Experimental Results

- DeepSeek-R1:14b achieved:
- Precision: 57.14 - Recall: 26.67 - F1-score: 36.36
- Outperformed smaller models like DeepSeek-R1:8b and Llama3.2.
- Fell short of state-of-the-art models like MelBERT and ContrastWSD.
- Results suggest that increasing model size improves performance.

Model	Rec	Prec	F1
SOTA Results			
MelBert	77.5	79.87	78.66
MsW_cos	77.88	80.31	79.07
FrameBERT	76.78	79.33	78.03
ContrastWSD	78.85	80.16	79.50
MY RESULTS			
Llama3.2-1b	73	18	29
Llama3.2-3b	07	50	12
deepseek-r1:8b	13	22	17
Deepseek-r1:14b	26.67	57.14	36.36

Conclusion

DeepSeek-R1:14b shows potential for metaphor detection but does not outperform state-of-the-art models.

Larger models in the DeepSeek-R1 series (up to 671B parameters) could improve results.

The unconventional approach of using direct prompting instead of feature extraction was necessary but may have limited performance.

Future Work

1

Explore larger DeepSeek-R1 models to improve performance.

2

Implement feature extraction for training a classification head (MLP or SVM).

3

Compare results with hybrid approaches that combine transformers with structured linguistic knowledge.