Team ID: 26
Team Name: Power^{Set}

AI-Generated Content Detection

UNIVERSITÉ Concordia UNIVERSITY

Abstract

- We worked on the *Academic Essay Authenticity Challenge*[1] to try and develop a model to detect AI-generated essays.
- We developed a model that can detect AI-generated content, but discovered countermeasures that can be used easily bypass detection.

Goal of the Project

- The goal of the project is to develop a model that can detect AI-Generated Content.
- We also want to thoroughly investigate the performance of the model beyond the specified datasets in the shared tasks, potentially discovering new avenues to advance in.

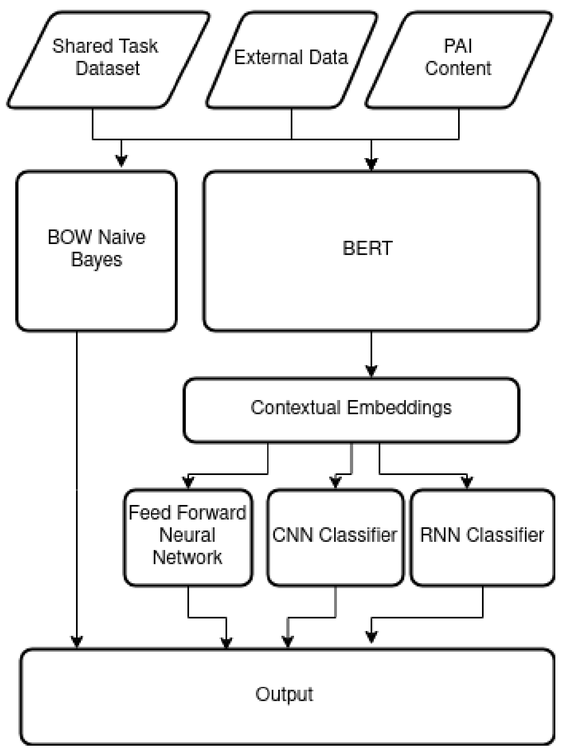
Methodology

Our project largely consists of three different datasets, namely:

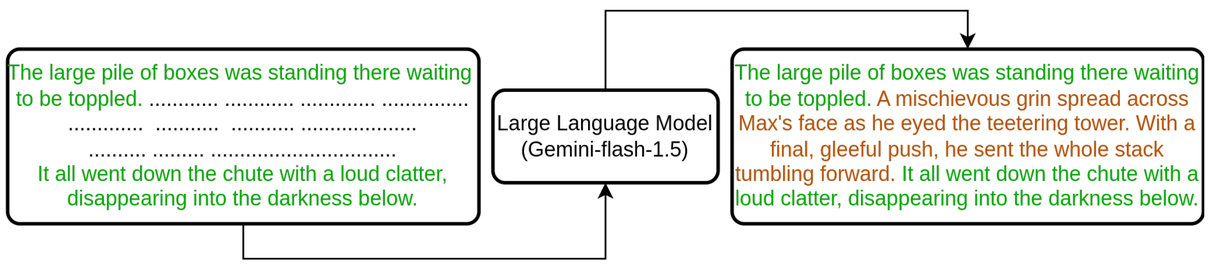
- The GenAI Content Detection dataset English texts (2096 training examples, 1626 testing examples)[1]
- The GenAI Content Detection dataset Arabic texts (2070 training examples, 481 testing examples)[1]
- A subset of the DAIGT dataset(3k used for training, 1k for testing)[2]

We developed four models to classify the texts:

- **Naive Bayes**: This model largely stands as a baseline to help understand and highlight the performance difference that is gained by the more advanced deep learning model.
- **Bert**: We use hugging face's transformers library for using pretrained
  - **NN**: The default head for BertForSequenceClassification, which uses a simple dense layer over the sentence embedding
  - **CNN**: Custom Model using a 1D CNN over the word embeddings
  - **LSTM**: Custom Model using a LSTM RNN over the word embeddings



Using the provided dataset, our model is trained to distinguish between AI-generated and human-written content by analyzing. However, this detection could potentially be bypassed by combining AI-generated text with human-written sentences. To explore this, we generated 100 essays by prompting an LLM to produce essays that incorporate specific sentences from human-written content verbatim. This approach aims to simulate a scenario where AI-generated text is subtly blended with human-written input. We then evaluated whether our model can still effectively detect the predominantly AI-generated nature of these essays.



Results

Classifier	Language	Precision	Recall	F_1-Score
Naive Bayes	English	0.56	1.00	0.72
	Arabic	0.72	1.00	0.83
	DGAIT	0.96	0.99	0.98
FFNN	English	0.81	1.00	0.90
	Arabic	0.996	0.946	0.971
	DGAIT	0.95	0.98	0.97
CNN	English	0.98	0.99	0.99
	Arabic	1.00	0.82	0.90
	DGAIT	0.97	0.97	0.97
RNN	English	0.99	0.77	0.87
	Arabic	1.00	0.87	0.93
	DGAIT	0.88	0.98	0.93

Table 1. Consolidated metrics for classifiers across languages and DGAIT[2].

	FFNN	CNN	RNN
Detection rate	3/100	5/100	1/100

Table 2. Model Performance on AI-Expanded Essays

Analysis

- **Naive Bayes Effectiveness**: Naive Bayes performed surprisingly well, likely due to its ability to leverage straightforward markers like extra punctuation or errant capitalization. While preprocessing and tokenization often discard these features, when they persist, they strongly indicate human-written text.
- **CNN vs. RNN on BERT**: The CNN classification head marginally outperformed the RNN head, and both performed similarly to standard BERT classification. We suspect this is for the same reason Naive Bayes excels—these models likely rely on simple, surface-level patterns unique to human and AI-generated text, rather than exploiting BERT's nuanced contextual embeddings.
- **Challenges with Hybrid Texts**: All models struggled to detect hybrid texts where AI expanded on human sentences, as the integration masks the surface-level markers they rely on. This underscores the models' reliance on shallow features rather than deeper semantic understanding.

Limitations and Future Work

- **Simplifications**:
  - The datasets implicitly assume a binary of *Human* and *AI*, whereas text can fall within that binary (i.e. A human-written essay with a few AI suggestions)
  - For small essays, there may not be any significant linguistic differences between a well-written human essay, and generated essays.
- **Research Question**:
  - Does the effectiveness of the classification depend on the tokenizer?
  - Are there other classification heads that could perform better on this task?

References

[1] GenAI Content Detection Dataset, *Academic Essay Authenticity Challenge*, GitLab repository, Consent form must be signed for access to the dataset. [Online]. Available: `https://genai-content-detection.gitlab.io/sharedtasks`.

[2] *Detect AI Generated Text Training Dataset*, Kaggle repository. [Online]. Available: `https://www.kaggle.com/datasets/thedrcat/daigt-v2-train-dataset/data`.

[3] Google AI, *Google Gemini API Quickstart*, Google AI Developer Documentation, Python SDK usage instructions. [Online]. Available: `https://ai.google.dev/gemini-api/docs/quickstart?lang=python`.

[4] D. Dimitrov, F. Alam, M. Hasanain, *et al.*, "Semeval-2024 task 4: Multilingual detection of persuasion techniques in memes," in *Proceedings of the 18th International Workshop on Semantic Evaluation*, ser. SemEval 2024, Mexico City, Mexico, Jun. 2024.