LLM - Detect AI Generated Text (Visual Analytics - UPF)

This report aims to give a general overview of the project. Too see all the results with the corresponding plots we encourage you to run the streamlit application and go through the different sections.

Analysis of our training corpus

The data we used to train the model [1] contained 347K rows, each of them containing a `text` and a `label` field. The label was 0 if the text was written by a human, and 1 if it was AI generated.

First of all we checked which was the proportion each of the two classes, AI generated or human written, in our training data. There were more human written essays (63%) than AI generated(37%). This difference was not too large so we decided to keep the data as it was.

Calculated fields

In order to understand better the distribution of our data, we decided to compute some `calculated fields` for each of the essays [2]. The ones which we found more relevant and interesting were the **number of words per essay** and the **average word length per essay**.

When we were performing the analysis we found out there were some outliers. Despite this, we decided to keep them and analyze the data in both cases, with and without them.

The number of words per essay seems to follow in both cases a Poission distribution. There is not a significative difference between both distributions apart from the skeweness. The human written essays tend to be larger than the AI generated one. This is arbitrary since AI can generate texts of any length. Therefore, we do not consider it as a valuable variable.

Regarding the average word length, when considering the outilers we can not analyze the distribution of the average word length. We can see that there is some non-overlapping zone. On the other hand, once we remove them there is a substantial difference in the average word length between the two classes. Both distributions are Normal. Despite there is a clear overlap between both of them, AI generated essays clearly have a higher word length than human ones, which can be clearly seen in the boxplot. So, this would be a meaningful variable if we built a predictive model with it.

When we compared jointly the two previously seen features, we can observe that AI generated essays tend to have a higher average word length for both long and short essays. On the other hand, human written essays have a lower average word length for all text lenghts.

As a general conclusion, we can observe that a predictive model based on these two features would not be very precise since there is only a feature that differentiates the two classes, and not very clearly.

Analysis of a sample of the corpus

The high computation requirements along with the length of the texts resulted into a very time consuming process to do the plots and the posterior analysis. This is why we took a 10.000 sample from our training dataset to get the following insights.

First we analyzed which were the more frequent words in the sample. The most repeated terms were student, school, people, help..., which gave us a hint that most of the essays were related whith academic topics. Then, we applied Named Entity Recognition (NER) [3] the most frequent type of terms are organizations, then person names and then cardinal numbers.

Finally we embedded the sample essays and applied a dimensionality reduction method, TSNE [4]. There is a clear split between the AI generated and the human written ones. In the real world this difference is not that clear. So, this training data we have will in fact help the model to differentiate between the two classes in purpose, but not by nature.

Model Development

To develop the model we followed a text classification guide from Huggingface [5]. They make the process very straightforward and the underlying model uses transformers, which are the state of the art in Natural Language Processing (NLP).

Model Results

When we were testing our model, we realized from the beginning that its performance was not the desirable one. It ended the training with an accuracy higher than 99% (in a test split of the training dataset mentioned in previous sections). This was a clear symptom that the training dataset is not challenging enough.

Therefore, we decided to also use a state of the art model [6] to compare it with. In order to analyze the performance of our model properly, we built our semi-custom test dataset from a couple of corpuses [7] [8]. We used the classical evaluation metrics.

With this new test dataset we got much more realistic results. Our model has a clear bias towards predicting that texts are AI generated with a `99.5%` of recall. It has an overall acccuracy of the `51.2%`, which is very low.

Explainable AI plots interpretation

In the streamlit app we also use SHAP plots to explain inference results. However, we believe that the results provided by SHAP are not very useful. The plot highlights tokens of the text that have been relevant for the prediction, but more often than not it only highlights single words or even punctuation characters. We believe that this is not very useful for a professor looking for plagiarism, for example, and it would be better to highlight suspicious sentences instead of words.

1. https://www.kaggle.com/datasets/jdragonxherrera/augmented-data-for-llm-detect-ai-generated-text/data?select=final_train.csv ↵
2. https://neptune.ai/blog/exploratory-data-analysis-natural-language-processing-tools ↵
3. https://www.turing.com/kb/a-comprehensive-guide-to-named-entity-recognition ↵
4. https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding ↵
5. https://huggingface.co/docs/transformers/tasks/sequence_classification ↵
6. https://arxiv.org/abs/2301.07597 ↵
7. https://arxiv.org/abs/2304.12008 ↵
8. https://arxiv.org/abs/2303.14822 ↵