Exploring the Distinction: Investigating the Recognition of Automatic Text Generation Systems and Differentiating Human Text from Language Models

Manex Agirrezabal

Centre for Language Technology (CST), Department of Nordic Studies and Linguistics, University of Copenhagen, Emil Holms kanal 2, 2300 Copenhagen, Denmark

Abstract

In this paper we present our approach for the detection of text written by Language Models (task 1). We further train models to classify the specific Language Model that creates the texts (task 2). Our final model ranks 49/76 for task 1 and 30/38 for task 2. Even though results are not among the best ones, we provide a more in-depth analysis of the employed features, which show the importance of character-level features and the low importance of TF-IDF weighted word-level features.

Keywords

text classification, language model detection, human-generated text, automatically generated text

1. Introduction

Language models are able to simplify and represent language based on a number of assumptions. Using these language models we can obtain the probability of a sequence to be part of the represented language, or we can use them for autoregressive generation and generate sequences that resemble that language.

There are different implementations of these models, some of which make use of limited memory assumptions, such as the n-gram models, and some others are, in theory, less limited and thus, more powerful. Among these last models we can find the ones based on Recurrent Neural Models (RNN), RNNs with Long Short-Term Memory or the more recent Transformer models. In order to see how language models have evolved recently, let's suppose we train a trigram (n-gram with $n = 3$) language model on John Milton's *Paradise Lost* and we try to generate some text with the prompt "And Adam".

> *and Adam to himself Reserving , human, rational, though immortal. Easy it may be quit, Fairest and easiest, of envy, and due praise, And disobedience: on me: flesh of thy train; there first received, so refused,*

IberLEF 2023, September 2023, Jaén, Spain

manex.aguirrezabal@hum.ku.dk (M. Agirrezabal)
https://manexagirrezabal.github.io/ (M. Agirrezabal)
0000-0001-5909-2745 (M. Agirrezabal)

we can see that the text resembles John Milton's words. Without attempting to compare this tiny language model to a Large Language Model as GPT, we tried to obtain some similar text using the popular chatbot ChatGPT. We provided the following prompt and we got the following answer on the 22nd of May, 2023:

> **Prompt**: *In John Milton's style, how would you follow the sentence "And Adam"?*
> **Response**: *And Adam, with steadfast gaze and resolute countenance, beheld the vast expanse before him, a tapestry of untamed wilderness, teeming with uncharted wonders and boundless potential.*

We further continued the interaction with ChatGPT and asked the following: "can you do it in iambic pentameter?". The response was:

> *And Adam, with his steadfast gaze and mien,*
> *Surveyed the vast expanse before his eyes,*
> *A tapestry of wilds yet unexplored,*
> *Where wonders dwelled, untamed and unrevealed,*
> *A realm of boundless promise, yet unknown.*

It is no doubt that recent language models sound notably more natural and fluent, especially when we compare them to older models. This brings many opportunities, but it also poses many dilemmas and dangers.

On the bright side, many repetitive tasks that involve reading the same data and writing very similar reports, could be simplified by the use of these models, such as, table report writing. These models seem to work also very well in style change, for instance writing a paragraph in a very different register, for instance writing an academic text for teenagers.

What seems to be a nice help, though, it can also be seen as a dangerous step. For instance, until recently, spam messages or scams were rather easy to spot (fool language, typical repeated scenarios, and so on). But with the advent of new LLMs, we, e-mail recipients, can now be tricked in an easier way. For the prompt "*Can you write a catchy SMS message to buy an umbrella?*" we get the following answer on the 23rd of May, 2023 (the answer even included emojis that could not be incorporated because of encoding issues):

> *Don't let the rain spoil your day! Stay dry and stylish with our fabulous umbrellas! Whether it's a sudden shower or a drizzle, our umbrellas have got you covered. Get yours now and dance in the rain with confidence! Shop today and embrace the rainy season with a touch of elegance. Hurry, limited stock available! Don't miss out on this weather-defying deal. #StayDryStayHappy*

Then, even though there are many benefits with these models, we believe it is relevant to be able to detect whether a text was created by a human or a computer.

In this paper, we present our approach to detect whether a text was written by a Language Model or by a human, as part of the AuTexTification shared task [1, 2]. Our model is part of an attempt to create generic text classification models for a number of tasks, in which the majority of the features are inspired by authorship attribution and profiling literature. The paper is

Task	#classes	#instances (train)	#instances (test)	FE. time (mins)
Task 1	2	33845	11169	62.5
Task 2	6	22416	7398	43.2

Table 1
Number of instances for each task in the training and testing sets and required time to extract features.

structured as follows. In the next section we give a brief overview of some related works. After that, we present the data and our employed methodology. Then we present our results followed by a more in-depth analysis of the results and the models. We then conclude the paper and propose some possible future directions.

2. Related work

With the improvement of language models, it is no surprise that recognizing whether a text was written by a human or a computational model is becoming more challenging. We review some works that attempted to solve this problem in the following lines.

Some researchers have worked on deepfake detection in tweets. In their work [3], they collected text from 23 bots and from the 17 human accounts they imitate. The texts from those accounts was generated using different techniques, such as GPT-2, RNNs, LSTMs and Markov Chains. Their experiments show that generative methods based on the Transformer architecture, for instance GPT-2, can produce high quality short texts, making it difficult to distinguish even for expert human annotators. They employ common features including Bag-of-Words, TF-IDF weighted BOWs, BERT representations and character level representations. They train several classifiers such as Logistic Regression, Random Forest or Support Vector Machines. As the best results were obtained by character-level models in detecting GPT-2 based texts, they mention that these features might be relevant to be further explored.

Many general classification works rely on the availability of large language models. In spite of that, Frohling and colleagues [4] propose a simple feature-based classifier for the detection of language model generated text, by the use of carefully crafted features that attempt to capture the main differences between human and machine-generated text. The main aspects that they model are: (1) lack of syntactic and lexical diversity, (2) lack of coherence, and (3) lack of purpose. They employ a number of features to model each of those aspects and besides, they make use of common author profiling features, such as character, syllable or word counts, amonf others. They train commonly used classifiers, namely, Logistic Regression, Support Vector Machines, Neural Networks (NN) and Random Forests.

3. Data and Method

We use the data [5] provided by the organizers of the shared task [1]. Please find relevant figures in the table below. As it can be seen, we have around 44,000 instances for the first task and around 30,000 instances for the second task. Both datasets are balanced.

	dummy	KNN	LR	LSVM	MLP	DT	RF	GB
weightedf1s	33.74	71.52	80.87	80.81	**85.03**	74.25	80.50	81.97
accuracy	50.36	71.52	80.88	80.86	**85.03**	74.25	80.52	81.97
Training time	$< 1s$	$< 1s$	$< 10s$	$< 1m$	$< 5m$	$< 5m$	$< 1m$	$< 25m$

Table 2
AuTexTification task 1 results (2 labels: generated, human).

This task is tackled following a general methodology that we expected to work for many tasks, including bot vs. human discrimination, subjectivity detection, sexism detection, and many others. Our original goal in the framework is to faithfully compare feature importance across different tasks, but in this article we will emphasize in the characteristics of bot vs. human and bot discrimination problems. Our feature set is mainly inspired by work on authorship attribution and profiling [6, 7]. We employed commonly used stylometric features, such as word-level bag-of-words (unigrams), TF-IDF weighted bag-of-words (unigrams), character-level bag-of-words (1-4grams), POS tag bag-of-words (1-4grams), morphological features as returned by the Stanza package (specify model) and BERT encoding of the input text by the `bert-base-cased` model [8].

In this study, we employ seven distinct classifiers for the purpose of training. These classifiers consist of K-Nearest Neighbors (with $K = 5$), Logistic Regression, Linear Support Vector Machine, Multilayer Perceptron, Decision Tree, Random Forest, and Gradient Boosting.

To ensure model evaluation, we adopt a train/test validation procedure. Specifically, we allocate two thirds of the available data for training purposes, while the remaining one third is reserved for testing. This approach allows for reliable assessment of the classifiers' performance and generalization capabilities.

4. Results

The results for task 1 and task 2, obtained from various classifiers, are provided in Tables 2 and 3, respectively. The evaluation metrics considered include the weighted F1-score, accuracy, and training time. We believe that training time holds significant importance in practical applications of such models, particularly in real-world scenarios involving private companies or public institutions.

Notably, the inference time is not included in the reported results as it demonstrates relatively similar performance across all classifiers, except for the K-Nearest Neighbors classifier. However, due to its comparatively low performance, the inference time is considered less relevant in this context.

For task 1, the classifiers achieved a ranking of 49 out of 76, with a macro F1-score of 57.35. In the case of task 2, the classifiers attained a ranking of 30 out of 38, yielding a macro F1-score of 45.72. These metrics provide an overview of the classifiers' performance in the respective tasks.

	dummy	KNN	LR	LSVM	MLP	DT	RF	GB
weightedf1s	5.08	31.94	42.75	34.73	44.75	31.90	40.43	**45.41**
accuracy	17.26	32.98	43.52	37.46	45.04	31.90	41.58	**46.01**
Training time	$< 1s$	$< 1s$	$< 10s$	$< 5m$	$< 5m$	$< 30s$	$< 30s$	$< 1h$

Table 3
Autextification task 2 results (6 labels: A, B, C, D, E, F).

5. Discussion

Rather unexpectedly, we can observe that our test results from task 1 are rather low, compared to the validation results from table 2. It could have been a case of overfitting to the training data or also that the domain shift of the test data affected more than what we expected, making our model domain dependent. In task 2 we observed that the GB method showed slightly better results but a significantly larger training time (as it can be seen in table 3). Consequently, we ultimately opted to utilize the Multilayer Perceptron as our chosen classifier for the test set, as it demonstrated similar results to GB.

Based on the feature importance of the Logistic Regression models trained, it seems like BERT related features can be good for finding out whether a text is written by a human or a bot. But they do not seem to hold the same power for establishing the system used (task 2).

The analysis of feature performance in our study reveals certain trends. Firstly, the utilization of Character Bag-of-Words features exhibits favorable predictive capability for both tasks under investigation. On the other hand, the incorporation of morphological features extracted through the Stanza library does not appear to provide predictive value for either of the tasks.

Notably, POS uni- or ngrams demonstrate relatively strong predictive potential as feature representations. These features, based on part-of-speech information, exhibit promising predictive power for the tasks at hand.

However, our findings indicate that TFIDF normalization does not significantly contribute to improved performance. The TFIDF normalized BOW features do not appear to possess substantial relevance in the context of the tasks examined, especially when compared to the regular BOW features.

6. Conclusion and Future work

In this paper, we presented our approach for tasks 1 and 2 of the AuTexTification Shared Task [1, 2], where we trained models for distinguishing between human vs. Language Model generated text, and besides, we trained models to predict the actual language model that was used to generate the text. Our models were inspired by authorship attribution and profiling literature, and we believe that results were fair, considering their simplicity.

Considering that our results are still quite low, there is still room for improvement. We would like to include more complex features, such as the ones proposed in [4] and check their importance with respect to other purely stylistic features. Besides, we included basic BERT features by making use of the `bert-base-cased` model. We believe that there might be more relevant models for the current tasks. Considering that POS-tag n-grams showed relatively high importance, a possibility could be to use the hidden representations of a Transformer-based POS-tagger, assuming that their hidden representations will have structural information encoded as vectors.

References

[1] A. M. Sarvazyan, J. Á. González, M. Franco Salvador, F. Rangel, B. Chulvi, P. Rosso, Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains, in: Procesamiento del Lenguaje Natural, Jaén, Spain, 2023.

[2] S. M. Jiménez-Zafra, F. Rangel, M. Montes-y Gómez, Overview of IberLEF 2023: Natural Language Processing Challenges for Spanish and other Iberian Languages, Procesamiento del Lenguaje Natural 71 (2023).

[3] T. Fagni, F. Falchi, M. Gambini, A. Martella, M. Tesconi, Tweepfake: About detecting deepfake tweets, Plos one 16 (2021) e0251415.

[4] L. Fröhling, A. Zubiaga, Feature-based detection of automated language models: tackling gpt-2, gpt-3 and grover, PeerJ Computer Science 7 (2021) e443.

[5] J. Ángel González, A. Sarvazyan, M. Franco, F. M. Rangel, M. A. Chulvi, P. Rosso, Autextification, 2023. URL: https://doi.org/10.5281/zenodo.7692961. doi:10.5281/zenodo.7692961.

[6] M. Koppel, S. Argamon, A. R. Shimoni, Automatically categorizing written texts by author gender, Literary and linguistic computing 17 (2002) 401–412.

[7] E. Stamatatos, A survey of modern authorship attribution methods, Journal of the American Society for information Science and Technology 60 (2009) 538–556.

[8] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.