Identifying AI-generated Content in New Articles

WENHAO LI, Northeastern University, US

ACM Reference Format:
Wenhao Li. 2024. Identifying AI-generated Content in New Articles. 1, 1 (April 2024), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 PROBLEM DESCRIPTION

Since the emergence of generative AI, we always anticipated that it would become efficient and focus on creative, rather than repetitive, tasks. However, in recent years, concerns have been raised in science and education as to how to distinguish between human intellect from AI-generated content [6]. The need to develop an effective detector of AI-generated content is growing.

2 INTRODUCTION

In their recent papers, Desaire H. et al. have identified 20 numerical features generated per paragraph, which were then used to train the classification model [3]. Their model has reached 92% - 98% accuracy in the detection of scientific articles in chemistry.

Similar strategies were utilized by Desaire H. et al. in another paper for the detection of scientific articles in all areas. 20 features same as the ones described in their previous paper were used to train their models. Their model reaches 99% accuracy in identifying human works from AI-generated content, showing that the features they select are effective in detecting AI-generated content in the scientific field [2].

The work done by Berglund, L. et al. concluded that if a model is trained on a sentence of the form "A is B", it will not automatically generalize to the reverse direction "B is A" [1]. They concluded by fine-tuning GPT-3 and Llama-1 on fictitious statements such as "Uriah Hawthorne is the composer of Abyssal Melodies" and showing that they failed to correctly answer "Who composed Abyssal Melodies?". This could be a potential feature to identify AI-generated content, by identifying "A is B" and "B is A" pairs in a particular article.

According to Elkhatat et al., their experiments compared the effectiveness of 5 different AI content detection tools available [4]. In their experiments, GPT-3.5 and GPT-4 were used to generate AI content. The results suggested that all five tools can detect GPT-3.5 rather accurately, but not for GPT-4 generated content. It can be seen that the selection of the database is a critical factor in determining the final accuracy.

Tf-idf and LSI methods also show potential in text classification. According to the work done by Wen Zhang et al., the performance of LSI method with support vector machine (SVM) surpasses that of the tf-idf method in text classification tasks [7].

Author's address: Wenhao Li, Northeastern University, US.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2024 Association for Computing Machinery.
Manuscript submitted to ACM

The combination of tf-idf and Word2vec Model, a pre-trained model for the prediction of given word through context, also gives promising results. Work done by Cai-zhi L. et al. shows that the combination methods have a 5% performance increase compared to tf-idf method alone [5].

Convolutional Neural Networks (CNN) with variant Long-Short Term Memory (LSTM) produce the best result by far. According to Hai Zhou, accuracy of this model can reach 99.9% in text classification tasks [8].

3 DATASET

Data set we used in this paper comes from NeuralNews DataSet, which contains 32k entries for both human-written new articles and ai generated ones. The human-written articles are extracted from the GoodNews dataset, which is extracted from the New York Times.

4 METHODOLOGY

20 features, which had been described by Desaire H. et al. [2] were extracted and loaded into a data frame. These data were then fed to a decision tree model for training. The 20 features are described in detail in the table below (table 1).

Table 1. Features in the model

Feature number	Feature type (1–4)	Short description	Greater in
1	1	sentences per paragraph	human
2	1	words per paragraph	human
3	2	“)” present	human
4	2	“-” present	human
5	2	“,” or “;” present	human
6	2	“?” present	human
7	2	“”“ present	ChatGPT
8	3	standard deviation in sentence length	human
9	3	length difference for consecutive sentences	human
10	3	sentence with <1 words	human
11	3	sentence with >34 words	human
12	4	contains “although”	human
13	4	contains “However”	human
14	4	contains “but”	human
15	4	contains “because”	human
16	4	contains “this”	human
17	4	contains “others” or “researchers”	ChatGPT
18	4	contains numbers	human
19	4	contains 2 times more capitals than “.”	human
20	4	contains “et”	human

Feature types: 1, paragraph complexity; 2, punctuation marks; 3, diversity in sentence length; and 4, popular words or numbers.

For tf-idf approach, punctuation and stopwords have been removed from the text data in the data set. Stemming of words has also been done. Both the stopwords library and stemming function are provided by nltk library.

The Gensim version of the word2vec model was used for training. Text data was tokenized and preprocessed to feed to the word2vec model for training. The resulting word2vec model was used to vectorize all text data in vector form. The vectorized text data was then fed to either a logistic regression model or a neural network for classification tasks.

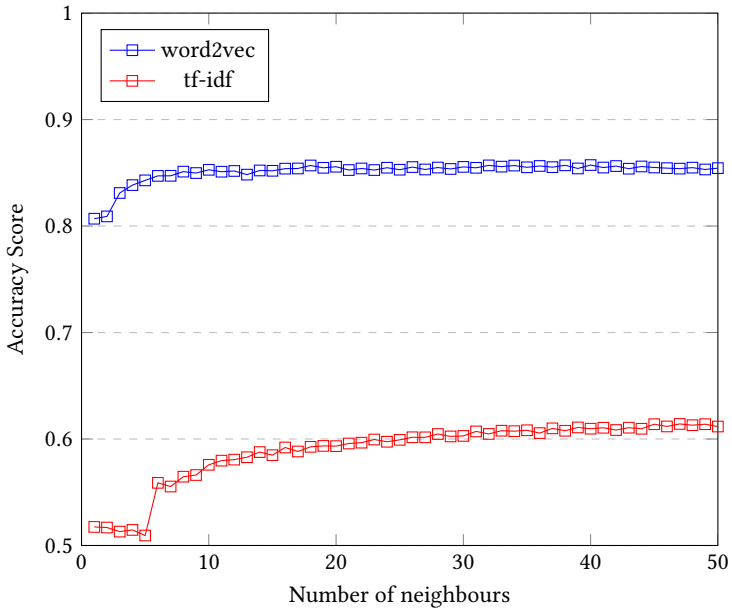
Logistic Regression, K-Nearest, and SVM models in the Sklearn library were used for classification tasks. C = 1.0 and linear kernel were chosen for SVM classification. For the K-Nearest model, 1 to 50 numbers of neighbors were fitted and plotted to find the best number of neighbors for classification tasks.

Keras version of Neural Network was used for classification tasks. For tf-idf approach, only the Dense layers and Dropout layers were used, whereas for the word2vec approach, Dense layers, Convolutional layers, MaxPooling layers, and Dropout layers were used.

5 EXPERIMENT RESULTS

The Accuracy Score of 1 - 50 neighbors for the K-Nearest model was calculated and plotted to choose the best number of neighbors for each embedding (Fig 1). The slope is close to 0 at 40 neighbors for tf-idf embedding, whereas for the word2vec approach, a similar trend was found at around 12 neighbors. Therefore 40 and 12 neighbors were chosen K-Nearest model for tf-idf and word2vec approach.

Fig 1. Accurarcy score of K-Nearest Model for word2vec approach



The accuracy score of the decision tree model is 66%, which is significantly lower than the model trained by Desaire H. et al. Possibly because of the work done by Desaire H. et al. is on chemistry articles and different keywords must be selected to differentiate news articles.

Tf-idf method combined with SVM or logistic regression model obtained 82% accuracy, similar results can be seen from the works done by Wen Zhang et al. and Cai-zhi L. et al., which suggested that the potential of this model has been reached and more advanced models are needed to achieve higher accuracy score.

Neural Network with tf-idf embedding reaches 83% accuracy, which is the highest among all models using tf-idf embedding.

The K-Nearest model with tf-idf embedding only yields 61% accuracy, which is significantly lower than the other 3 methods. This shows that the K-Nearest model is not suitable for classifying tf-idf data.

Table 2. Accuracy Score of different models with different word embedding

Word Embedding	Model Type	Accuracy Score
NA	Decision Tree	65.51
tf-idf	SVM	81.77
tf-idf	K-Nearest	60.87
tf-idf	Logistic Regression	82.55
tf-idf	Neural Network	83.34
word2vec	SVM	93.67
word2vec	K-Nearest	85.74
word2vec	Logistic Regression	93.63
word2vec	Neural Network	90.02
transformer	Bert (Deep Neural Network)	99.54

Word2Vec model with logistic regression and SVM yields the highest accuracy score, around 93%. Whereas the word2Vec model with neural network model obtained 91% accuracy. The neural network model yields a lower accuracy score than logistic regression, which suggests the neural network needs further tuning.

6 CONCLUSION AND DISCUSSION

For both Tf-idf and word2vec embedding, logistic regression, SVM, and Neural Network models have similar accuracy scores, whereas the accuracy score for the K-Nearest model is significantly lower. This shows that the K-Nearest model is not suitable for identifying AI-generated content versus human-written ones.

The Deep Neural Network combined with the transformer model yields the highest accuracy score. This might be due to either the use of the transformer model or the number of layers and neurons contained within the deep neural network. Therefore the performance of the neural network with tf-idf and word2vec embedding could potentially increase as we add more layers and neurons to the model.

All four models (SVM, logistic regression, K-Nearest, Neural Network) perform better with the word2vec model, which indicates that the word2vec model is more suitable for the classification of AI-generated content than tf-idf embedding.

7 FUTURE DIRECTIONS

The Neural Network model with both embeddings could be upgraded by adding more layers and neurons to the model. A great leap in performance might be seen as we increase the depth of the neural network.

A similar methodology could be applied to other datasets to see the performance.

REFERENCES

[1] Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". arXiv:2309.12288 [cs.CL]

[2] Heather Desaire, Aleesa E. Chua, Madeline Isom, Romana Jarosova, and David Hua. 2023. Distinguishing academic science writing from humans or ChatGPT with over 99% accuracy using off-the-shelf machine learning tools. *Cell Reports Physical Science* 4, 6 (2023), 101426. https://doi.org/10.1016/j.xcrp.2023.101426

[3] Heather Desaire, Aleesa E. Chua, Min-Gyu Kim, and David Hua. 2023. Accurately detecting AI text when ChatGPT is told to write like a chemist. *Cell Reports Physical Science* 4, 11 (2023), 101672. https://doi.org/10.1016/j.xcrp.2023.101672

[4] Ahmed M Elkhatat, Khaled Elsaid, and Saeed Almeer. 2023. Evaluating the efficacy of AI content detection tools in differentiating between human and AI-generated text. *International Journal for Educational Integrity* 19, 1 (2023), 17.

[5] Cai-zhi Liu, Yan-xiu Sheng, Zhi-qiang Wei, and Yong-Quan Yang. 2018. Research of Text Classification Based on Improved TF-IDF Algorithm. In *2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*. 218–222. https://doi.org/10.1109/IRCE.2018.8492945

[6] Jahna Otterbacher. 2023. Why technical solutions for detecting AI-generated content in research and education are insufficient. *Patterns* 4, 7 (2023), 100796. https://doi.org/10.1016/j.patter.2023.100796

[7] Wen Zhang, Taketoshi Yoshida, and Xijin Tang. 2011. A comparative study of TF*IDF, LSI and multi-words for text classification. *Expert Systems with Applications* 38, 3 (2011), 2758–2765. https://doi.org/10.1016/j.eswa.2010.08.066

[8] Hai Zhou. 2022. Research of Text Classification Based on TF-IDF and CNN-LSTM. *Journal of Physics: Conference Series* 2171, 1 (jan 2022), 012021. https://doi.org/10.1088/1742-6596/2171/1/012021