A classifier of naturalness of text style - authentic or synthetic

Jan Tobolewski
Faculty of Electronics, Telecommunications and Informatics.
Gdańsk University of Technology
Gdańsk, Poland
s175502@student.pg.edu.pl

Karol Baran
Faculty of Electronics, Telecommunications and Informatics.
Gdańsk University of Technology
Gdańsk, Poland
s171426@student.pg.edu.pl

Natalia Rucińska
Faculty of Electronics, Telecommunications and Informatics.
Gdańsk University of Technology
Gdańsk, Poland
s176285@student.pg.edu.pl

Piotr Szczuko
Faculty of Electronics, Telecommunications and Informatics.
Gdańsk University of Technology
Gdańsk, Poland
szczuko@multimed.org

Text generation has been growing in popularity in recent years and can be used in order to for example generate summaries, generate fake news, disinformation by generating scientific, political, and medical texts with potentially high social harm. Being aware the seriousness of the problem, a deep neural network model that could be a component of an anti-plagiarism system is proposed in this article. Proposed solution is a text style classifier - natural or synthetic. The resulting model has good predictive ability - the distinction between natural and synthetic style is possible, suggesting that there are some differences between styles. The model's prediction accuracy of 80-90% is a satisfactory result. The results obtained are compared with those of interviewees who achieved an accuracy of 50%. The relatively short learning and inference times encourage the use of the model in practical anti-plagiarism type systems.

Keywords—Text generation, Deep Learning, Graph neural networks, Anti-plagiarism software

I. Introduction

Artificial Intelligence (AI) algorithms are perpetually evolving to keep on generating more convincing texts [1]. Average human can read full text and fail to notice that it is not written by another human. Evolving generators force classifiers to evolve with the same momentum.

Modern approaches to diverse language processing and generating problems include transformer-based language models like GPT-2 and BERT [2]. Recently text generation models like GPT-2, GPT-3 and chat-GPT were introduced. These models are neural network (NN) systems proposed in order to serve such tasks like text translation, autocompletion or summarization. They mostly rely on attention based NNs, more precisely transformer-based. Transformers are neural networks which transform one type of data like image, sequence, or text into same or other tasks.

More recently Graph Neural Networks (GNN) were introduced to analyse data that can be interpreted as graphs [3]. Graphs are treated mathematically as nodes and edges representing connections between them. GNN algorithm was successfully applied to analyse images, text and chemical compounds properties.

In this work GNN based system was used to classify style of the text is proposed.

II. Methods

Work on the project was divided into four main stages. Firstly, the dataset was prepared. In collected dataset both natural and synthetic (generated by GPT-like model) texts. Since the domain of the study is medical popular-scientific articles in press, GPT-2 was expected to be not well suited for the generation of text with specific domain knowledge [8]. Therefore, generator was fine-tuned in order to obtain better performance. Then classifier model to distinguish style of the text was developed. Finally, comparison of the results obtained with the effectiveness of distinguishing text style by the respondents in online survey was made.

Graphical description of the study is presented in the Figure 1.

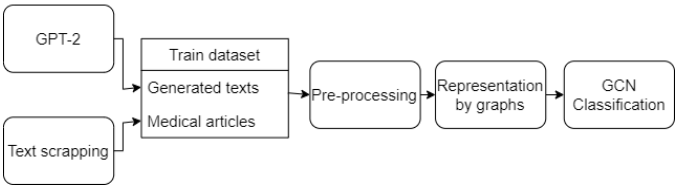

Fig. 1. Workflow of the study.

A. Text scrapping

To obtain natural texts it was decided to use articles taken from Harvard Health Publishing [9]. Python web scraping method as implemented in BeautifulSoup library [10] was used to automatically download articles. With a list of articles URLs, each of them was used to download HTML code of article. Data was extracted into JSON file with article title, authors, date of publishing, text category and link to original website in order to prepare proper documentation of the study.

B. Text generation

The key aim while utilizing this approach is to produce sentences depending on the initial words. The model's main goal is to continuously create new words based on input sequences of known text until it reaches a certain length or an end identifier.

As an incentive first sentence of the natural text was used. GPT-based model was then asked to continue the chapter. Since the model takes hyperparameters like temperature and top_k, their optimal values were determined.

C. Generator fine tuning

Since GPT-2 was originally proposed with chat-like solutions in mind, it was assumed that it might not be well suited for generation of text with specific domain knowledge in medical sciences. Therefore, model fine-tuning was proposed. To fine-tune the model natural text obtained from MedicalNewsToday [11] were used (in order to ensure that fine-tuned model would not benefit from text used to classification).

D. Data cleaning

Data cleaning required separate approach for natural and generated texts, which are described in detail below. Prepared dataset was randomly divided into train, validation and test sets with ratio 80:10:10.

Natural texts preparation required analysing and detecting remaining artifacts of HTML code, which was done by detecting HTML brackets. That search also allowed to eliminate tables. Secondly parts of Unicode were replaced with their ASCII counterparts. Readable text was divided into sentences, using tokenize package form NLTK library [12]. Tokenization allowed to count exact number of sentences in a text and dividing them into five-sentences short paragraphs. Each short text was tested for percentage of numerical digits in it. It was estimated that for this dataset numbers should not make more than 5% of whole text. This method allowed to find texts with more reference's numbers then text or unusual enumerating. Texts overstepping this limit were excluded from database. Lastly texts were scanned on words "References" and "Sources" to minimalize the risk of getting unnatural texts, like citations and references and sources were only allowed in the last sentence of texts (since this is quite common practice in press to provide source at the end of the article).

Generated texts required less additional preparation. Firstly, generated text required replacing Unicode signs with their ASCII counterparts. Then they were divided into sentences, using tokenize package form NLTK library and cutting text after first five sentences. Some analysis of texts on their percentage of numerical digits was performed. For this set it was estimated that the numbers should not make over 30% of the texts. It was noticed that high percentage of numerical digits is caused by many links and references, but in general texts had more statistical data in them.

E. Tokenization and graph representation

As an intermediate step between the preparation of dataset (pure text form) and the classification, a few transformations have been applied. The five-sentence texts were split into tokens with BERT large cased tokenizer [13]. Tokens are single stems carrying information e.g., the word "sleeping" consists of "sleep" and the suffix "ing". In the next step, the tokens prepared in this way are transformed into embedding vectors representing their meaning throughout the text. The BERT large cased [13] model has been used.

Each text was then represented using a graph. Figure 2 presents an example representation of the relationship of the individual embeddings by graph. The values of the individual embeddings are included as features of the nodes. Edges, on the other hand, define the relationship between consecutive nodes. In the study, this way of linking tokens was used because it imitates the human perception of looking at a text (the human eye can notice the previous current word and the next one). Edges were connected according to the following scheme: each node was connected to the next node and the second node in the sequence (e.g. 2nd with 3rd and 2nd with 4th). Connections were given as weight depending on the distance between the nodes, 1 for the nearest neighbour, 2 for the further neighbour. The exception is the character "[SEP]" denoting a separator between sentences, the connection of nodes with this character was assigned a weight of 0.

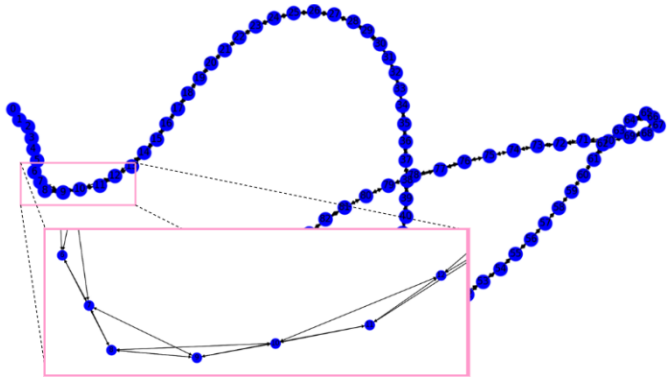

Fig. 2. Graph representation of example sentence.

F. GNN model building

To graph dataset classification Graph Convolution Neural Network (GCN) as implemented in PyTorch-geometric library [14] has been used. Based architecture contains three convolutional layers with 128 neurons. Input layer contains 1024 neurons because of embedding vector size. Output fully connect layer with 128 input neurons and 2 output neurons. There are 2 output neurons, because one-hot encoding has been used. Dropout 0.2 was applied at last layer. Adam optimizer was used with learning rate equal to 0.005. Loss criterion used was multiclass cross entropy. Every epoch loss and accuracy has been measured on training and validation set as well.

III. Discussion and Results

A. Generator hyperparameters finding

Two most important hyperparameters for text generation were found to be temperature and top_k. Temperature value 0.7 is not sufficient for creative text generation - neural network often repeats itself and it leads to no novelty in generated text (like in 2137 where net proposed just personal data of random doctor and nothing more). In case of fine-tuned model this value of the parameter is enough – in that case there is no lack of novelty in a text (like in 2137 text about covering majority of health issues was generated). For higher temperature value like 0.9 neural network often changes the subject, sometimes till the

end of text - from cancer to menopause (3315) or from ask doctor to cheese production (1104). Example of text 278 shows that for temperature 0.8 with top_k=20 narrative is moving slowly, with top_k=40 is more similar to natural text and with top_k=60 subject is quite often changed starting with sound than brain response, chemicals in nervous system and vaccines (in real life it is quite hard to read article with that many topics covered). Optimal value of temperature of about 0.8 with top_k equal to 40 is found to be most reliable.

B. Generator fine tuning impact on style of generated text

Since authors of dataset did not provide descriptions of topics covered in articles, short keyword analysis performed. In the dataset articles covering topics like asthma (article no. 3), smoking (no. 12), drugs in treatment of infections (no. 15), neuroscience research (no. 99), teletherapy (no. 430), blood tests (no. 1754) and health insurance issues (no. 1911) to name few of them. Keyword searching for whole dataset using TF-IDF algorithm reveals that words like "people", "symptoms", "cancer", "doctor" and "pain" happens to occur most often and not specific to one disease are observed. Using Yake algorithm similar findings can be observed - now extracted keywords are "people", "symptoms", "doctor", "treatment", "risk", "body" and "condition". Both of the keyword extractors showed that for the whole dataset keywords are rather generic and not specific to one disease or health condition. Therefore, usage of the dataset to fine-tune GPT-2 for medical articles generation is seen as justified.

Some important changes related to the style of the text was observed after model was fine-tuned.

Dialogues are missing in fine-tuned model - all Q&A type articles generated by pure GPT-2 from time to time are not generated anymore after fine-tuning. Instead of "What is your current health insurance?", sentences like "The doctor will often ask about any changes in the symptoms of a person with Crohn's disease or celiac disease, including: any changes in the bowel or how hard the stool is" (2137). There is a different style of two text - second one is more formal, informative and more field-related vocabulary is involved. It is clearly observed that after asking net to generate text about "Ask the doctor" pure model generates dialogue while fine-tuned model generates description.

Model after fine-tuning better understands given prompt. For example, in 5132 text incentive was: "Massage can be a helpful add-onto conventional medical care for back pain". After fine-tuning generating model proposed text covering information like: which kind of pain might be treated that way, which methods are used and some basic doctor's recommendation. Pure GPT2 model concentrated more on last words of incentive and generated text about back pain and possible method to decrease it and forgot about context of message at all.

What is more, style of text is more similar to news in magazine (popular-science for example). For example, in 2971 original GPT2 propose sentences like: "If you're injured, do not panic. We'll take a deep dive into your symptoms and techniques." while in fine-tuned net corresponding phrases are: "If you do it correctly, you may be able to avoid some of the potential complications. The following are some potential side effects from sitting for long periods of time: a headache that lasts longer than a few minutes, trouble sleeping, a loss of appetite, fatigue, nausea, vomiting, dizziness, low blood pressure."

Passive voice is used more often in fine-tuned model. For example, in 3315 "The findings have been published in the journal Nature Communications. In their study paper, the scientists point out that the current approach could be more effective in the future if it was applied in a large number of patients. In the future, they say, it could be possible to eliminate cancer in its tracks and control its spread as rapidly as possible". Passive voice was used several times and citation (rather common for pure GPT-2) was replaced by the phrase "they said". It is also interesting to point out that real scientific journal with high impact factor was mentioned in that case. Comparable fragment generated by pure model "But when the first cells died, the researchers decided to ignore them. "We had to look at all the data that had been collected; the data that had been collected. We wanted to have an accurate picture of the risks of radiation exposure," says Dr. Jost." is written in less formal and less scientific style.

It can be also seen that pure GPT2 sometimes fail to generate medical text even though text about medicine was provided. For example, in 520 provided to neural network sentence was: "Particularly in the legs, it's the muscles surrounding the veins that provide the pumping power that drains the vessels near the surface of the skin and then push the blood up through the "deeper" vessels that travel toward the heart". Pure GPT2 completed text with: "What's going on?" I asked." That's why I said it's a mystery and how can we explain it, because we can't really explain it, it's too complex and just too hard to believe." He laughed". On the other hand, proposed text with strong medical background: "The body uses many different factors and sensations to regulate blood flow to the legs, including heart rate, breathing rate, and respiration rate. The following sections will look at some of these factors and their effects on the thigh and arms".

C. Qualitative analysis of artifacts in scrapped and generated texts

In table 1 statistical characteristics of the dataset is presented. It can be seen that sets are quite well balanced. On average generated texts consists of more words. Especially texts generated with base GPT-2 model. As all texts consist the same amount of sentences, it indicates that generated sentences are longer than natural ones. It however does not transfer to lexical diversity [15], which is average number of unique words divided by average number of words. Texts generated by tuned model have lower number of unique words. Which can mean our tuned model is often repeating words, still the difference is acceptable margin that allows to further work with tuned model. Interestingly average number of unique words in a sentence does not uphold this trend, as both used models have on average higher number than natural texts. From this we can presume that models create more complex sentences, but over the course of whole text they stay strict to the prompt topic.

TABLE I TRAIN DATASET PARAMETERS

	Natural texts	Generated texts	Generated texts – tuned model
Number of texts	5957	5009	5430
Average word count	108.409	126.626	114.946
Average number of unique words	75.089	74.864	71.968
Average number of words in a sentence	21.678	25.324	22.989
Average number of unique words in a sentence	19.511	21.502	19.941
Lexical diversity	0.705	0.606	0.639

D. Classification results

Training has been executed on computer with 16 threads CPU and NVIDIA GeForce RTX 3090Ti GPU supported with 128GB of RAM memory. Training 100 epochs took an average of 14 minutes 30 seconds.

After training for 100 epoch the classification model with accuracy of 92% on training and 85% on validation set. Loss after training was 0.32 and 0.39 on training and validation sets accordingly. Figure 3 presents the learning curves; on the left-hand side has been presented the change of the loss function at each epoch and on the right-hand side the change of the accuracy. The blue color indicates the values obtained on the training set, while the orange color indicates the values obtained on the validation set.

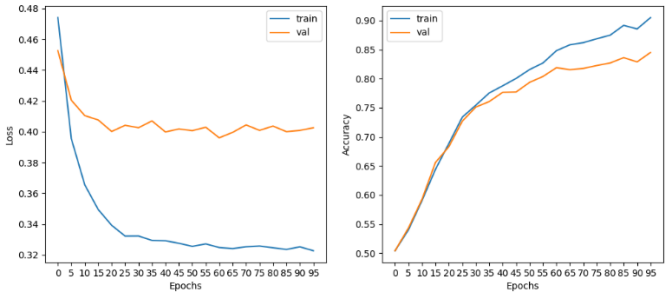

Fig. 3. Learning curves, GPT-2 model.

The model was tested on a test set containing 518 synthetic texts and 579 natural texts (a total of 1097 texts). Accuracy of about 83% was obtained with F1 score, precision and recall at 0.875, 0.880 and 0.879 respectively. A confusion matrix was generated, a graphical interpretation of which is shown in Figure 4, where 1 denotes natural texts, 0 denotes synthetic tests. The confusion matrix illustrates the ability of the classifier to verify the naturalness of the texts. Significantly fewer confusions were observed for natural texts classified correctly.

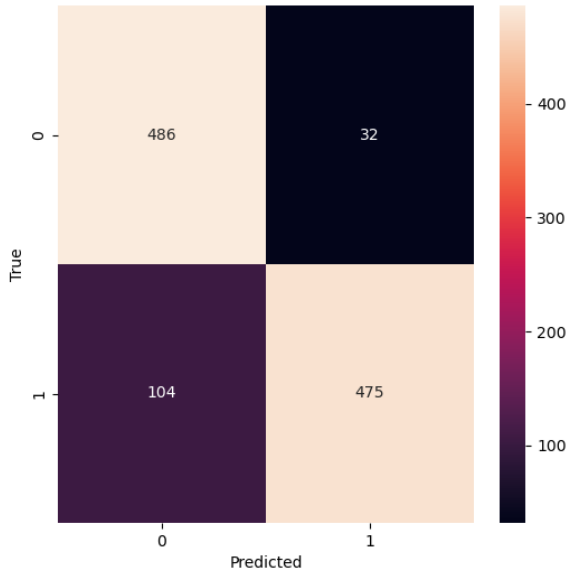

Fig. 4. Confusion matrix, GPT-2 model.

The same experiment was performed for a dataset consisting of natural texts and texts generated using a trained model to imitate the style and vocabulary of medical texts. In that scenario similar metrics were obtained as in previously described classifier – classification accuracy also was at 92% and 85% with slightly lower loss values at 0.32 and 0.37 for training and validation splits. Figure 5 presents the learning curves during this learning scenario.

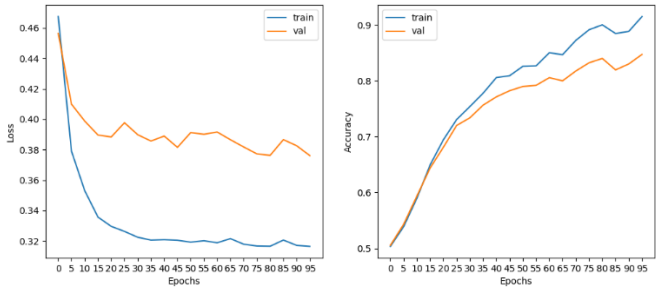

Fig. 5. Learning curves, GPT-2 fine-tuned model.

The trained model was tested on a test set containing 579 synthetic texts and 560 natural texts (1139 texts in total). Obtained accuracy on test set was about 87% with F1 score, precision and recall at 0.9376, 0.9378 and 0.9375 respectively.

The trained model was tested on a test set containing 579 synthetic texts and 560 natural texts (1139 texts in total). Obtained accuracy on test set was about 87% with F1 score, precision and recall at 0.9376, 0.9378 and 0.9375 respectively. A confusion matrix was generated, a graphical interpretation of which is shown in Figure 6.

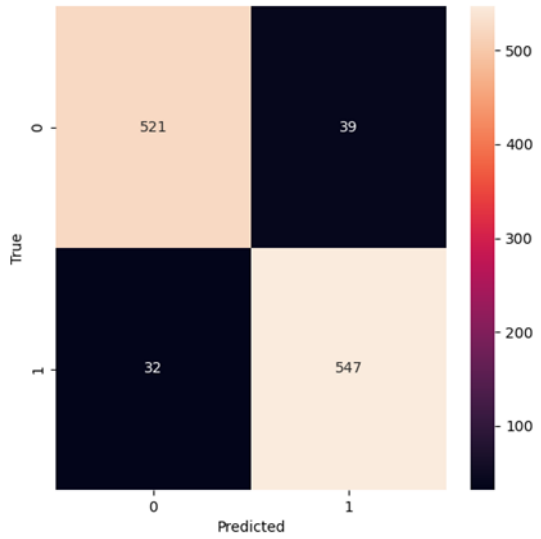

Fig. 6.Confusion matrix, GPT-2 fine-tuned model.

The following conclusions can be drawn from the analysis of the results presented above:

• The obtained model has a good prediction accuracy - the distinction between natural and synthetic styles is possible, that may suggest the existence of some differences between the styles.

• For the tuned model, a higher value of precision than sensitivity indicates a particular ability of the model to recognise synthetic texts - relatively few natural texts will be misclassified by the model. The potential use of the model in anti-plagiarism-like systems aimed at verifying whether a text was written by a human (e.g. a student writing an essay on health sciences or a journalist reporting on new scientific reports in medical disciplines) is postulated.

• A comparison of the metrics of the tuned model with the use of data generated with the pure GPT-2 model shows that a much more effective classifier was obtained when the tuned GPT-2 was used to generate. The results presented here do not allow a clear interpretation of this fact. It is only hypothesised that this is a consequence of the fact that, in the texts generated by pure GPT-2, some texts had little natural artefacts strongly differentiating them from natural texts such as frequent change of topic, high factuality of description such as contact details of the doctor, etc. When the classifier was taught on texts with a style closer to journalistic, it is likely that the distinction was already focused solely on the details distinguishing the two styles.

• The analysis of the search for the optimal network architecture clearly shows that, due to the relatively small data set, there is a risk of overfitting the model for large architectures with multiple convolutional layers. The resulting model has comparable predictive capacity on the learning, validation and test sets, and the changes of the loss function values during training is characterised by similar monotonicity for the learning and validation sets - these facts indicate that model overfitting has not occurred.

• The relatively short learning and inference times favour the use of the model in practical anti-plagiarism-like systems.

E. Comparison of the results achieved with the survey result

To accompany this paper a survey was conducted to examine if people can distinguish the difference between generated and natural texts. Survey consisted of 14 texts form our database. To reduce probability of respondents mistakes, all questions were the same – "Is this text generated?". First group of questions had two text, respondents had to point which text is generated (or both, or none). Second group consisted of six questions, each with one text. Respondents were asked to answer in a scale from 1 to 5, where 1 is strongly disagree, and 5 is strongly agree. For only three questions we received over 50% correct answers. After gathering all answers metrics [6] were calculated, obtained accuracy was about 53% with F1 score, precision and recall at 0.4297, 0.3409, 0.5811 respectively. A confusion matrix was generated, a graphical interpretation of which is shown in Figure 7.

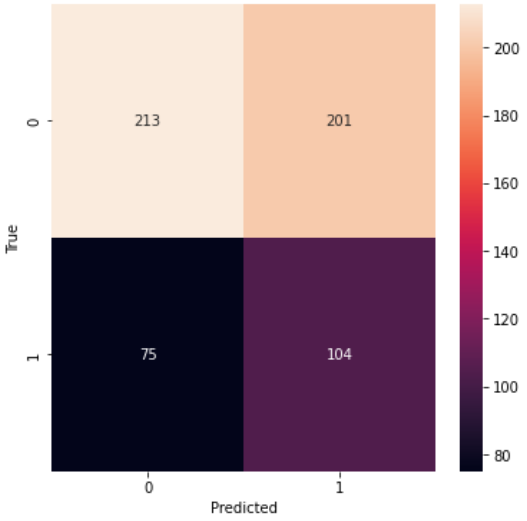

Fig. 7. Confusion matrix, human classification.

Results achieved were significantly lower than those obtained by both used models. Most respondents couldn't notice differences in styles, results show their answers were close to random.

IV. SUMMARY

In the present work classifier of the naturalness of text was proposed based on GNN architecture. The work was dedicated to area of medical popular-science articles. The obtained model's accuracy of 80-90% is found to be a satisfying score. These results are better than those obtained by survey respondents. Utilization of graph neural networks was found interesting approach imitating human perception of style of text. Major advantages of the propoes solution are relatively short training and inference times. Proposed methodology was found to be

universal and could be adopted to be used with any generating model. In future works, attention should be paid to analyse other graph structures since it is possible that the use of semantic connections between tokens would improve the performance of the classifier. Need to perform similar study for recognizing generated texts in other languages than English is highly recommended.

REFERENCES

[1] ZHANG, Qiuyun, et al. AI-powered text generation for harmonious human-machine interaction: current state and future directions. 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2019, 859-864.

[2] GRUETZEMACHER, Ross; PARADICE, David. Deep Transfer Learning & Beyond: Transformer Language Models in Information Systems Research. ACM Computing Surveys (CSUR), 2022, 54.10s: 1-35.

[3] WU, Zonghan, et al. A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems, 2020, 32.1: 4-24.

[4] XIONG, Jiping; HUANG, Tao. An effective method to identify machine automatically generated paper. In: 2009 Pacific-Asia Conference on Knowledge Engineering and Software Engineering. IEEE, 2009. p. 101-102.

[5] AMANCIO, Diego Raphael. Comparing the topological properties of real and artificially generated scientific manuscripts. Scientometrics, 2015, 105: 1763-1779.

[6] WILLIAMS, Kyle; GILES, C. Lee. On the use of similarity search to detect fake scientific papers. In: Similarity Search and Applications: 8th International Conference, SISAP 2015, Glasgow, UK, October 12-14, 2015, Proceedings 8. Springer International Publishing, 2015. p. 332-338.

[7] NGUYEN, Minh Tien; LABBÉ, Cyril. Engineering a tool to detect automatically generated papers. In: BIR 2016 Bibliometric-enhanced Information Retrieval. 2016.

[8] BUDZIANOWSKI, Paweł; VULIĆ, Ivan. Hello, it's GPT-2--how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. arXiv preprint arXiv:1907.05774, 2019.

[9] Harvard Health, https://www.health.harvard.edu/

[10] Richardson L. Beautiful soup documentation. April. 2007;

[11] ALAA TRIKI, 2k clean medical articles (MedicalNewsToday), Kaggle, https://www.kaggle.com/datasets/trikialaaa/2k-clean-medical-articles-medicalnewstoday

[12] NLTK library, https://www.nltk.org/ NLTK library;

[13] BERT large model (cased) https://huggingface.co/bert-large-cased.

[14] FEY, Matthias; LENSSEN, Jan Eric. Fast graph representation learning with PyTorch Geometric. arXiv preprint arXiv:1903.02428, 2019.

[15] LAWSON, Rebecca. Fine-tuning Daria: Exploring the Implications of Temperature, Epochs, & Corpus Size on GPT-2 Screenplay Generation. 2021.

[16] CABITZA, Federico; CAMPAGNER, Andrea. Who wants accurate models? arguing for a different metrics to take classification models seriously. arXiv preprint arXiv:1910.09246, 2019..