

# Survey on Plagiarism Challenges

Volodymyr Taranukha

*International Research and Training Center for Information Technologies and Systems, 40, Acad. Glushkova av. Kyiv, 03187, Ukraine*

## Abstract

This paper describes the current state of plagiarism detection and the challenges that arise in modern society and are caused by plagiarism. There are several significant aspects that were highlighted: technological aspects caused by recent developments of modern NLP tools, social aspects caused by the ongoing COVID-19 pandemic, development of new content similarity detection methods, etc. All of them add new aspects to plagiarism challenges.

## Keywords <sup>1</sup>

Content similarity detection, plagiarism detection, text similarity, machine learning

## 1. Introduction

Modern society is more and more entrapped in the global communication environment. This ranges from TV to social networks, from science to advertisement, and from entertainment to propaganda. A significant part of these data sources deals with text in a variety of forms. This resulted in a surge of text generation techniques on top of old rewriting, appropriation, and plagiarism. Different areas of communication suffer differently from such malaise. Alongside malice text generation which plagues social media plagiarism stays one of the worst things that affects the infosphere. Text generation implemented as a component in bots creates a false image of grassroots support while hiding actual astroturfing and often creates an echo-chamber effect. This leads to politicians making wrong decisions with devastating effects. Plagiarism, especially machine-assisted plagiarism undermines the fundamentals of modern science both in scientific research and in university study since it's much easier to turn in autogenerated text instead of results of actual study.

There are some commonalities between content similarity detection, text rewriting, and text generation. Such commonalities lie in the aspect of text similarity and mathematical tools (measures) to measure such similarity. Text generation does not necessarily directly connect to this measure, but there are some links in there at least by usage. It is convenient to use some rewriting tool as one created by Grammarly [1] in combination with some tool enhanced by GPT-3 [2] to generate some elements of the text whole cloth. Additional coherence metrics tools [3] can be applied on top of it to make the whole text more appealing. And this raises the issue of fair use on one side and plagiarism detection on the other [4]. Search Engine Optimizers (SEO) content creators are free to use text generation tools since it technically is not plagiarism. However, this creates a significant amount of online accessible texts with similar stylistics, vocabulary, and such.

## 2. Background of current trends of plagiarism detection


As it was noted content similarity detection and plagiarism detection was among areas where the development never stopped. According to Google Scholar [5] the number of indexed articles on plagiarism detection was steadily raising for the last 5 years at rate over 3,000 a year (2022 has less, however it is not ended yet).

---


*Information technology and implementation (IT&I-2022), November 30 - December 2, 2021, Kyiv, Ukraine*

EMAIL: volodymyr.taranukha@gmail.com (A. 1)

ORCID: 0000-0002-9888-4144 (A. 1)

 © 2022 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

This avalanche of publications has several trends, social and educational being among them. The pandemic of COVID-19 forced educational institutions to shift online, with a significant part of the educational process turning paperless. It in turn caused a surge of plagiarized assignments to be delivered to the teachers [6]. This is even more pronounced for Ukraine [7] since on top need to reform the education and socio-economical effects of COVID-19 since February 2022 we have the Russian invasion which forced many students to relocate away from their schools and universities. This kind of situation in turn kept the development and integration of plagiarism detection tools continuing at a hastened pace.

## **2.1. Technical aspects of plagiarism**

In order to analyze the plagiarism one needs establish framework: what is plagiarism, what how it is dealt with, that including data sources and methods, how new technological developments look like, how it is connected to other issues which can help or muddle the waters more.

### **2.1.1. Defining plagiarism**

One of the main issues in academic circles is plagiarism. It has been studied for many years in an effort to reduce plagiarism, preserve the standard of writing, and safeguard the author's rights. A violation of an author's or writers' copyright is referred to as plagiarism. It refers to using someone else's ideas or works without giving them due credit. It is observed that there are several definitions and types for plagiarism.

Dictionaries define plagiarism in quite straight forward yet insufficient manner. Merriam-Webster dictionary defines plagiarism as "to steal and pass off (the ideas or words of another) as one's own and: to commit literary theft" [8]. Cambridge dictionary defines it as "the process or practice of using another person's ideas or work and pretending that it is your own" [9]. However, there is nothing there about the means to discern plagiarized text from not plagiarized one.

There is no final agreement on what is plagiarism, yet categorization of plagiarism is quite developed and most of researchers agree at least on part of the basics.

There are such types of commonly recognized plagiarism with some variation; nevertheless the concepts behind names are mostly the same [10-12].

1. Copy and paste
2. Mix and paste
3. Sacrifice of unimportant part to make text look different
4. Structural rewriting plagiarism
5. Translation
6. Self-plagiarism

Some authors offer their own classification, such as [13]: secondary source, invalid source, duplication, paraphrasing, repetitive research, replication, misleading attribution, unethical collaboration, verbatim plagiarism, and complete plagiarism. However such classifications are often either very specialized or misleading.

Also, there are some research venues that try to tackle specific kind of plagiarism, for example, teachers plagiarism [14], yet there is little success there since sometimes one and the same term referring to different issues, not to mention the basic question: what is there in teachers plagiarism (or any other kind of specific plagiarism by the author of plagiarized text) that makes it either worth researching independently or at least different enough to guarantee own category?

### **2.1.2. Legal aspects and their influence on field of research**

From a legal standpoint, there are some documents describing what to do and how to treat the text with plagiarism in it. For example, in Ukraine, there is a hard 30% limit on not own text for scientific works and including citations. Direct plagiarism without pointing out the source is entirely prohibited. Any plagiarized (or suspicious of plagiarism) can be ground to strike down the paper entirely on any

level, that is from homework of student to PhD thesis [15]. In India, they use a notably different approach. UGC [16] stated in the draft policy that an academic misconduct panel should be constituted by higher educational institutions to investigate cases of plagiarism and submit the report to the panel. UGC has announced the Indian draft policy on plagiarism for academicians and researchers with levels of penalty. There is no penalty for 10 percent of similarity in articles, theses, projects, etc. At level 1 a paper must contain similarities above 10 percent to 40 percent. At level 2 a paper must contain similarities above 40 percent to 60 percent. At level 3 a paper must contain similarities above 60 percent. However, UGC declared that a zero-tolerance policy must be used in core areas of research. And if found then plagiarism disciplinary authority of the higher educational institutions must use the maximum penalty.

For the comparison, in the United States plagiarism is not a crime as is. However there is robust copyright protection along with notions of “breach of contract” (contract cheating [17]) and fraud that allows stopping and punishing plagiarizer in most cases.

Yet, as of the examples shown above neither legislative regulation offered clear measures when and how to discern plagiarized sentences, passages, or documents from non-plagiarized ones. So, I can conclude that the current state of the art in the legal sphere does not make any significant impact on actual development of scientific methods and commercial tools intended to combat plagiarism.

More so, as it was mentioned before, some things such as contract cheating muddle the waters even more. Contract cheating occurs when students turn in assignments they hired others to complete for them in order to receive academic credit: human or machine. Since the advent of internet services, this type of academic fraud has been more prevalent globally and it keeps growing. Many institutions switched to online exams during the worldwide COVID-19 [18] pandemic, and in Ukraine Russian invasion exacerbated this problem even more than before. From my own experience plagiarism ended as the most widely spread underlying aspect of contract cheating in Ukraine.

Very often automated paraphrasing tools [19] are used for this means which adds a new dimension to the problem. For example what about a hypothetical scenario in which a student uses such a tool to paraphrase content from file-sharing websites while citing the original source in a reference list? On one hand, citing the original source in a reference list suggests that the student did not intend to cheat and present somebody else’s research as own, yet by most definitions of plagiarism it is plagiarism. More so, if the assignment is given in non-native language. So, what if a student is writing in their native tongue, translates it into English, and then runs the text through a paraphrasing tool? It will result in text that will have some amount of stylistic clues pointing to plagiarism. And when one has a vague to no idea of what is inside this or that plagiarism analysis tool one will have some errors during evaluation. However, certain level of obscurity is important for any plagiarism detection tool with simple (or predictable) rules inside, since automatic paraphrasing tools will exploit any known weaknesses or data on the internal workings of plagiarism detection tools to make “better” plagiarized texts.

### **2.1.3. Plagiarism detection background**

There is very little research in the field on how much is enough to declare something plagiarism. For example, the work [20] uses MapLemon corpus to tackle this problem quantitatively. This corpus contains English language essays written by experimental online participants which were asked to write and submit essays on very specific topics. By having very restrictive guidelines the corpus gives good representation how one and the same thing can be represented as text. However, MapLemon is very limited in scale which greatly reduces its value.

This way, most works concentrate on the tasks of creating some kind of machine learning-based infrastructure and learning some kind of model parameters without trying to measure where the line between plagiarism and not actually is, such as [21]. Yet, any machine-learning methodology at its best gets some weight coefficients to some explicitly or implicitly defined rules. And it is good if those rules are explicitly defined and analyzed.

There are some good datasets on plagiarism, including MSRP corpus [22] and the PAN plagiarism corpus 2011 (PAN-PC-11) [23] which covers how one text can be "creatively" rewritten into another. However, they do not provide enough diversity to show all necessary variations of plagiarism for

many tasks. So, many researchers use auto-generated and auto-obfuscated texts to somehow fix the issue and obtain enough data.

It is important to underline that I do not delve deeply into methods which use images and other graphical components of documents to discern fraud or plagiarism as in [24]. I assume that this task is too complex to be reliably automated right now at least until further development of image comparison tools which will be able to understand structure of image better.

There are tools and means to detect text similarity and plagiarism in source code [25]. They have their own niche since the student often need to turn in their source code and it's useful to check it for appropriations. Also such tools can provide means to improve programming performance as long as it comes to boilerplate code, since there is little meaningful difference between similar classes with very similar common behavior. More so, using auto-generated code, having same approach and maintaining same standards is actually beneficial to programming performance in general. Yet, I do not analyze tools such tools in this paper.

This paper deals with natural language tools since I assume the problem both complex enough to be worth researching and yet manageable. According with this and regardless of means to create plagiarized text plagiarism detection tasks are divided into such main categories.

1. Sub-Lexical
2. Lexical
3. Syntactic
4. Semantic
5. Stylometric
6. Structural
7. Citation
8. Cross language

The sub-lexical task deals with spam-derived [26] plagiarizing techniques when some symbols in the analyzed text were intentionally replaced with similarly looking symbols, for example "i.e." and "i.e." are actually two different strings of symbols from two different languages. As one can see in this case humans and automatic tools perceive different content in the same text: humans can potentially see coherent text while plagiarism detection system will see some (not)-coherent text with some inserts.

The lexical task focuses on a document's lexical structure (as in [20]). N-grams or some kind of dictionary fingerprinting (up to the means used in search engines to collapse all similar documents into a single entry), clustering methods, and longest common subsequence are among the most popular lexical methods. The system good at first two levels of tasks perform well on copy-paste and mix-and-paste type of plagiarism. This is the main task for most commercial systems especially if the system uses Internet to search for potential sources.

The syntactic task analyses and tracks positional syntactic changes [27] and can partially address minor paraphrases which are not semantic in nature. It is especially important in Ukrainian and other languages leaning on the synthetic side of the linguistic spectrum in contrast with English and other analytic leaning languages.

The semantic task analyses the meaning of a document by considering synonyms, antonyms, and semantic similarity/distance. Both SEO and plagiarizers often use simple synonym substitution to make semantically similar text with different appearance. So, embeddings (vector semantic-based methods) are common in the systems solving semantic task now. While Latent Semantic Analysis is still in use, different embeddings based on deep neural networks [28] are steadily overtaking everywhere in the field of Natural Language Processing and plagiarism detection is no exclusion to this process. For a language like Ukrainian, it is also quite important since we have significant room for verb-to-noun and noun-to-verb transformation (for example "будівництво" and "будують" in many contexts are interchangeable while they are different parts of speech and have different syntactic roles).

The stylometric task is an approach to the document as a single entity with a single style. It's a complex approach that extensively uses tools form lexical task in combination with syntactic distance measurement. It is a statistical method that analyses an author's style under the assumption that each author has a consistent style of writing. While it works fine for native speakers with consistent language habits yet for non-native speakers discrepancies observed in the produced text style often

depend on the amount of effort and time poured into editing certain parts of the text. That is why I consider this task as the least important of all.

The structural task analyses how structural features such as keywords, headers, paragraphs, and references are presented along with the distribution of the words in a document. Graph comparison approaches are widely used in structural task [29]. However, due to limitation of underlying mathematical problem of sub-graph isomorphism this is not used very often.

The citation [30] task compares sets of source documents (and occasionally their order), it's a very fast and efficient tool for finding big-block text appropriations.

The cross-language task was among the hardest tasks in this list, partially covered by solving the citation task. However, the development of machine translation made doing it easier. Later development of cross-language deep neural networks provided efficient embedding methods [31] which allowed the efficient crossing of the gap between languages [32].

Development of Deep Neural Networks allowed setting a new ambitious task: image plagiarism detection. It is relatively new (first paper available at Google Scholar dates back to 2012) and extremely underdeveloped.

#### **2.1.4. Plagiarism analysis software types**

The tools are divided into standalone and online. WCopyFind[33] is an example of a standalone program while iThenticate[34] works online without the need to install any software.

There are three types of tools based on data source usage. Internal database tools such as CopyCatch [35] and WCopyFind detect plagiarism within a database. External database tools check the similarity of available external sources on Internet such as EVE2 and EduTie [36]. And some tools such as Turnitin [37] and iThenticate use both internal and external databases for plagiarism detection.

In respect to language, there are monolanguage, multilanguage (operating as monolanguage tools for the set of languages), and cress-language tools [32].

As for now, standalone internal database monolanguage systems are the most prolific systems both at the commercial and research side.

## **2.2. Neural networks at plagiarism detection**

Nowadays more and more plagiarism detection systems rely on ML-based subsystems to let them learn the rules which define if the passage under consideration is plagiarized or not implicitly. Powerful neural networks along with significant advantages, like an ability to solve cross-language plagiarism detection, also create the drawback: it is effectively impossible to dig out why this or passage was labeled as suspicious. Nevertheless, NN based ML is the best method available now for plagiarism detection.

Any plagiarism detection system must have a dataset. Since there is not enough good datasets on plagiarism many researchers use auto-generated texts either to pad the available human-generated datasets or use entirely machine-generated datasets. So, the issue of machine-generated examples was analyzed.

In [38] the effectiveness of six different word embedding models in combination with five classifiers for distinguishing human-written from machine-paraphrased text was evaluated. The most important part it that best performing automatic classification approach achieves an accuracy of 99.0% for documents and 83.4% for paragraphs. This is useful result showing that in spite of explosive development of machine-generated plagiarizer systems such systems still leave notable signs in the plagiarized texts enabling specific ML-training to combat the plagiarizers most often used by students.

In [39] machine-paraphrased plagiarism was analyzed. The effectiveness of five pre-trained word embedding models was evaluated, combined with machine learning classifiers and state-of-the-art Neural Networks language models. Preprints of research papers, graduation theses, and Wikipedia articles were paraphrased using different configurations of the tools SpinBot [40] and SpinnerChief [41]. The best performing technique in the paper, Longformer [42], achieved an average  $F_1$  score of

0.8099 ( $F_1=0.9968$  for SpinBot and  $F_1=0.7164$  for SpinnerChief cases), while human evaluators achieved  $F_1=0.784$  for SpinBot and  $F_1=0.656$  for SpinnerChief cases. The authors conclusively showed that this approach outperforms both humans and usual methods implemented in commercial systems such as Turnitin.

However, in [43] researchers reported that GloVe[44] can outperform BERT under certain conditions. It must be noted that the research is centered on the concept of “tortured phrases” that often appear out of misused translation. For example such phrases include “counterfeit consciousness” which is used instead of “artificial intelligence”. It becomes more prominent if the plagiarism was generated by circular translation such English-German-English. Usage of automatic translation tools also contributes to probability of appearance of such tortured phrases. The researchers used cosine score to prove that: they claim that GloVe embeddings produce cosine score of 0.12 for tortured phrases and 0.3 for normal phrases while BERT embedding gives 0.5 for tortured phrases and 0.55 for normal phrases. The researchers explained it as excessive influence of context in BERT model preventing the system from generalization unlike GloVe embeddings which are context-free. Also, it must be noted that such weak results were received when the final part of architecture that perform actual decision if the sentence is potentially plagiarized is simple enough. So, hunt for this kind of phrasing is a good kind hypothetical of evidence of plagiarism, yet it must not replace more general keys detectable by more general tools.

It can be concluded that automatic generation of plagiarized text produces better detectable texts no matter the method of generation: rephrasing or translation. So, it is better to use human made or at worst machine assisted and human controlled datasets.

It can be assumed that the most important part of Neural Networks usage in plagiarism detection is embedding that represents vector semantics.

In [45] an experiment with different embeddings was described. The results show that the BERT pre-trained model offers the best results and outperforms GloVe and RoBERTa in monolanguage task. This is half-expected since BERT usually outperforms simpler (or simplified) embedding methods. The authors used indexing ranking as metrics with BERT and RoBERTa offered ranking 0.76 and 0.72 while GloVe+TF-IDF offered 0.57. It has to be noted that a more robust RoBERTa does not allow getting enough collateral drift to improve plagiarism detection specifically like it was expected after the results shown in [43].

In [46] cross-language plagiarism detection with contextualized word embeddings was analyzed. The evaluation experiments show that contextualized word embeddings is an appropriate approach that improves performance greatly. SBERT[47] is used to make embeddings of the whole sentences in contrast with [39] where Longformer was used. It must be mentioned that the method designed in [46] does not use any translation system. The tests performed have demonstrated that it works for different language pairs such as English-French, English-Spanish, English-Portuguese, and English-Russian. For PAN-PC-12 Spanish partition  $F_1 = 0.7938$  and for PAN-PC-12 German partition  $F_1 = 0.778$ . The English-Russian comparison is very important since the languages of English-Russian pair is almost at the different ends of the synthetic-analytic language scale, with drastically different syntactic structures on top of notably different semantics.

In [48] another cross-language research was performed. The authors claimed accuracy of 0.9701. While not so impressive like [46] nevertheless, Arabic-English experimental results showed that using deep neural networks with rich semantic features achieves encouraging results.

In [49] attempt at cross-language plagiarism detection research for English-Persian pair was performed. Unlike most papers on cross-language plagiarism in this one significant effort was spent on combating post-translation obfuscation. The percentage of text with different obfuscation methods are: no obfuscation 29%, mechanical obfuscation 29%, human paraphrasing 10%, summarization 8%, circular translation 10%, split 9%, merge 5%. For Persian language the most efficient method among analyzed was translation plus monolanguage plagiarism detection (the suspicious documents were translated with Google translate API from Persian into English). Using this approach they managed to achieve score of  $F_1=0.713$ .

The performance of BERT in plagiarism detection task shows that some kind high-end context-sensitive embedding and some kind of complex final resolution mechanism are both absolutely necessary to achieve good result in plagiarism detection.

Longformer and SBERT are not the only methods to process long springs.

In [50] architecture based on a Long Short Term Memory (LSTM) and attention mechanism called LSTM-AM-ABC boosted by a population-based approach for parameter initialization was proposed. The paper employs a population-based metaheuristic algorithm (Artificial Bee Colony) to solve the problem. The algorithm can find the initial values for model learning in all LSTM, attention mechanism, and feed-forward neural network, simultaneously. On MSRP dataset and compared to several other methods including Siamese CNN+LSTM[51] and CETE [52] the method showed the best performance with average score of  $F_1=0.857$ .

In [53] to model the “partial matching” between documents, a Partial Matching Convolutional Neural Network (PMCNN) was proposed for source retrieval. PMCNN exploits a sequential convolution neural network to extract the plagiarism patterns of contiguous text segments. The experimental results on PAN 2013[54] and PAN 2014[55] plagiarism source retrieval corpus has shown that PMCNN can boost the performance of source retrieval significantly compared to ranking SVM-based approach [56]. General performance of NN on PAN 2013 and PAN 2014 corpora gives  $F_1=0.6171$  and  $F_1=0.5474$  respectively. The paper one more time confirms that neural networks outperform other methods. The important contribution however consists in usage of CNN, since this type of feed-forward networks is one that has the best chance to be analyzed for the purpose of extracting knowledge unlike LSTMs.

In [57] very ambitious task of plagiarism detection in the image-based medium was undertaken. The reasoning behind the research is sound: it takes much more effort to plagiarize images to the same degree of being unrecognizable compared to text thus making such a system a valuable addition to any large-scale plagiarism detection system, especially if it is a commercial one. It is important to notice that there are very few papers on the subject. I was able to find only 52 papers in Google Scholar 32 of them were published since 2018.

The paper [57] proposed a system that can potentially cover the usual flaws of image plagiarism detection systems. The research is focused on flowcharts, since it is the most vulnerable image type, however it is suitable for any kind of images, as it was shown in experimental section. Alas, while the system can detect unedited and flipped images with high accuracy, yet the accuracy goes down drastically if operations such as rotation, greyscaling, and cropping were applied. Rotated image can be detected with 80% accuracy while greyscaling reduces the accuracy to 20%. The most telling problem is a drop in accuracy for cropped images to 60%. It defeats implementation of idea to convert the flowchart into a directed graph. Hypothetically such approach must enable the system to detect the shape of the flowcharts under any positioning changes, as long as the graph stays the same. Yet for some reason it failed which indicates that one needs another approach to the task.

In my opinion, any such system can be used as an auxiliary to a text-based one but will neither gain the same efficiency nor can serve as the main tool. The problems with any image plagiarism detection system are exacerbated by the rapid development of image-generation Neural Network-based systems such as DALL-E [58], which can create image whole cloth in any style out of text description. Any plagiarizer can describe any image (especially one such as flowchart) and feed the description into image generation engine in order to receive originally looking yet totally plagiarized image.

### 3. Conclusions

The field of plagiarism detection is undergoing rapid development in some areas while staying stagnant in others. The most complex task in direct text plagiarism detection was a cross-language task. And now powerful cross-language tools to solve the task are already developed and will continue development in the foreseeable future. For now the most complex task is image plagiarism detection.

However, the legislative part of the issue is lacking and most probably will stay lacking as long as researchers are unable to make some tools able to explain why and how this or that passage was labeled as suspicious. It is a task as hard as any task of extracting knowledge from the neural network. And with the development of Deep Neural Networks, the problem was only exacerbated. Also, I do not expect problem of interaction between scientist and legislators to be resolved by agreement

between designers of commercial plagiarism detection software by creating de-facto industry standard which can serve as agreeable common ground.

More so, there is no golden standard for what is plagiarism and what is not neither on the national level nor on the international, so the practice of relying on automated tools to evaluate any paper or student assignment will still produce a significant amount of friction between students and teachers. At least with research papers there is tried and tested peer review which while being slow and not perfect solves the issues of plagiarism detection in most cases.

## 4. Acknowledgements

The author would like to acknowledge the following people for their contributions to the research: prof. Anisimov A.V. from faculty of Computer Sciences and Cybernetics, Taras Shevchenko National University of Kyiv for useful suggestions on the nature of natural language texts and general support; staff members of dpt. 165 of IRTC IT &S, Kyiv for libraries and support provided during this research.

## 5. References

- [1] Introducing Grammarly's New Tone Rewrite Suggestions URL: <https://www.grammarly.com/blog/tone-rewrite-suggestions>
- [2] T. Brown, et al., Language models are few-shot learners. *Advances in neural information processing systems*, (2020), 33: 1877-1901.
- [3] O. O. Marchenko, O. S. Radyvonenko, T. S. Ignatova, et al., Improving Text Generation Through Introducing Coherence Metrics, *Cybernetics and Systems Analysis* 56 (2020) 13–21. doi:10.1007/s10559-020-00216-x
- [4] T. Bretag, S. Mahmud, A model for determining student plagiarism: Electronic detection and academic judgment. *Journal of University Teaching & Learning Practice*. 6. (2009) 10.53761/1.6.1.6.
- [5] Google Scholar URL: <https://scholar.google.com/>
- [6] K.A. Gamage, E.K.D. Silva, N. Gunawardhana, Online delivery and assessment during COVID-19: Safeguarding academic integrity. *Education Sciences*, 10(11), (2020) 301.
- [7] V. Luniachek, et al., "Academic integrity in higher education of Ukraine: current state and call for action." *Education Research International* (2020): 1-8.
- [8] Merriam-Webster, "Dictionary by Merriam-Webster: America's most-trusted online dictionary," URL: <https://www.merriamwebster.com/>.
- [9] Cambridge Dictionary (online), "PLAGIARISM | meaning in the Cambridge English Dictionary," URL: <https://dictionary.cambridge.org/dictionary/english/plagiarism>.
- [10] L. Bornmann, "Research misconduct—definitions, manifestations and extent." *Publications* 1.3 (2013): 87-98.
- [11] H. Sharma, S. Verma, Insight into modern-day plagiarism: The science of pseudo research. *Tzu-Chi Medical Journal*, 32(3), (2020), 240.
- [12] D. Weber-Wulff, *False feathers: A perspective on academic plagiarism*, Springer-Verlag, Berlin, 2014.
- [13] Roig, Miguel. "Plagiarism and self-plagiarism: What every author should know." *Biochemia Medica* 20.3 (2010): 295-300.
- [14] R. M. Ghițău, L. Măță, University Teachers Plagiarism-A Preliminary Review of Research. *BRAIN. Broad Research in Artificial Intelligence and Neuroscience*, 10, (2019) 22-32.
- [15] Ministry of Education and Sciences of Ukraine Order 40 12.01.2017 URL: <https://zakon.rada.gov.ua/laws/show/z0155-17#Text>
- [16] UGC Policy on Plagiarism. 2017. [https://www.ugc.ac.in/pdfnews/8864815\\_UGC-Public-Notice-on-Draft-UGC-Regulations,-2017.pdf](https://www.ugc.ac.in/pdfnews/8864815_UGC-Public-Notice-on-Draft-UGC-Regulations,-2017.pdf)
- [17] K. Ahsan, S. Akbar, B. Kam, Contract cheating in higher education: a systematic literature review and future research agenda. *Assessment & Evaluation in Higher Education*. 2022, 47(4) 523-39.



- [18] S. E. Eaton, K. L. Turner, Exploring academic integrity and mental health during COVID-19: Rapid review. *Journal of Contemporary Education Theory & Research (JCETR)*, 2020, 4(2), 35-41.
- [19] J. Roe, M. Perkins, What are Automated Paraphrasing Tools and how do we address them? A review of a growing threat to academic integrity. *Int J Educ Integr* 18, 15 (2022). <https://doi.org/10.1007/s40979-022-00109-w>
- [20] P. Juola, "How much overlap means plagiarism? A controlled test corpus." *Concurr. Sess* 12 (2022): 13-14.
- [21] V. Vrublevskiy, O. Marchenko, "Development and Analysis of a Sentence Semantics Representation Model." *Cybernetics and Systems Analysis* 58.1 (2022): 16-23.
- [22] B. Dolan, C. Quirk, C. Brockett Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. *Proc. 20th International Conference on Computational Linguistics (COLING 2004)*. (23–27 August 2004, Geneva, Switzerland). Geneva, 2004. P. 350–356. URL: <https://aclanthology.org/C04-1051>
- [23] The PAN plagiarism corpus 2011 URL: <https://webis.de/data/pan-pc-11.html>
- [24] M.A.G. van der Heyden, The 1-h fraud detection challenge. *Naunyn-Schmiedeberg's Arch Pharmacol* 394, (2021) 1633–1640. <https://doi.org/10.1007/s00210-021-02120-3>
- [25] M. H. Ismail, M. M. Lakulu, A Critical Review on Recent Proposed Automated Programming Assessment Tool. *Turk. J. Comput. Math. Educ*, 12, (2021), 884-894.
- [26] S. A. Rojas-Galeano, "Revealing non-alphabetical guises of spam-trigger vocables." *Dyna* 80.182 (2013): 50-57.
- [27] K. Vani, G. Deepa, "Text plagiarism classification using syntax based linguistic features." *Expert Systems with Applications* 88 (2017): 448-464.
- [28] Y. Wang, et al., "A comparison of word embeddings for the biomedical natural language processing." *Journal of biomedical informatics* 87 (2018): 12-20.
- [29] M. Franco-Salvador, et al., "Cross-language plagiarism detection over continuous-space-and knowledge graph-based representations of language." *Knowledge-based systems* 111 (2016), 87-99.
- [30] B. Gipp, J. BEEL, Citation based plagiarism detection: a new approach to identify plagiarized work language independently. In: *Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*. (2010), 273-274.
- [31] J. Devlin, et al., "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018). et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).
- [32] T. Pires, E. Schlinger, D. Garrette, "How multilingual is multilingual BERT?." *arXiv preprint arXiv:1906.01502* (2019)
- [33] Wcopyfind URL: <https://plagiarism.bloomfieldmedia.com/software/wcopyfind/>
- [34] iThenticate URL: <https://www.ithenticate.com/>
- [35] CopyCatch URL: <https://www.elute.io/copycatch>
- [36] T. Lancaster and F. Culwin, "Classifications of plagiarism detection engines," *Innov. Teach. Learn. Inf. Comput. Sci.*, vol. 4, no. 2, (2005) 1–16
- [37] TurnItIn URL: <https://www.turnitin.com/>
- [38] Foltýnek, Tomáš, et al., "Detecting machine-obfuscated plagiarism." *International Conference on Information*. Springer, Cham, (2020) 816-827
- [39] J. P. Wahle, et al. Identifying machine-paraphrased plagiarism. In: *International Conference on Information*. Springer, Cham, (2022). 393-413
- [40] K. Dey, R. Shrivastava, S. Kaushik, A Paraphrase and Semantic Similarity Detection System for User Generated Short-Text Content on Microblogs. In: *Proceedings International Conference on Computational Linguistics (Coling)*, (2016), 2880–2890
- [41] T. Foltýnek, N. Meuschke, B. Gipp, Academic Plagiarism Detection: A Systematic Literature Review. *ACM Computing Surveys* 52(6), (2019), 112:1–112:42 <https://doi.org/10.1145/3345317F>
- [42] I. Beltagy, M.E. Peters, and A. Cohan Longformer: The Long-Document Transformer. *arXiv:2004.05150* (2020)

- [43] P. Lay, M. Lentschat, C. Labbé, Investigating the detection of Tortured Phrases in Scientific Literature. In: Proceedings of the Third Workshop on Scholarly Document Processing. (2022) 32-36.
- [44] J. Pennington, R. Socher, and C. Manning. "Stanford glove: Global vectors for word representation." (2017).
- [45] R. Rosu et al., "NLP based Deep Learning Approach for Plagiarism Detection." RoCHI-International Conference on Human-Computer Interaction, Romania. 2021
- [46] D. D. A. Vaz, Cross language plagiarism detection with contextualized word embeddings, (2021)
- [47] N. Reimers, I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks." Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2019) 671-688
- [48] S. Alzahrani, H. Aljuaid, Identifying cross-lingual plagiarism using rich semantic features and deep neural networks: A study on Arabic-English plagiarism cases. Journal of King Saud University-Computer and Information Sciences, (2020), <https://doi.org/10.1016/j.jksuci.2020.04.009>
- [49] H. Asghari, et al., On the use of word embedding for cross language plagiarism detection. Intelligent Data Analysis. 23(3) (2019). 661-680. <https://doi.org/10.3233/IDA-183985>
- [50] S.V. Moravvej et al., An LSTM-based plagiarism detection via attention mechanism and a population-based approach for pre-training parameters with imbalanced classes. In: International Conference on Neural Information Processing. Springer, Cham, (2021). 690-701.
- [51] M.T.R.Laskar, X. Huang, and E. Hoque. Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task. in Proceedings of The 12th Language Resources and Evaluation Conference. (2020).
- [52] E. L. Pontes, et al., "Predicting the semantic textual similarity with siamese CNN and LSTM." arXiv preprint arXiv:1810.10641 (2018).
- [53] L. Kong, et al., "A Partial Matching Convolution Neural Network for Source Retrieval of Plagiarism Detection." IEICE TRANSACTIONS on Information and Systems 104.6 (2021): 915-918.
- [54] M. Potthast, et al., "Overview of the 5th international competition on plagiarism detection," Proc. CLEF 2013 Evaluation Labs and Workshop, Valencia, Spain, (2013) 301–331
- [55] M. Potthast, et al., "Overview of the 6th international competition on plagiarism detection," Proc. CLEF 2014 Evaluation Labs and Workshop, Sheffield, United Kingdom, (2014) 845–876
- [56] L. Kong, et al., "A ranking approach to source retrieval of plagiarism detection." IEICE TRANSACTIONS on Information and Systems 100.1 (2017): 203-205
- [57] A. S. B. Ibrahim, O. O. Khalifa, D. E. M. Ahmed, Plagiarism Detection of Images. In 2020 IEEE Student Conference on Research and Development (SCOREd) IEEE, (2020) 183-188
- [58] DALL-E: Creating Images from Text URL: <https://openai.com/blog/dall-e/>