УДК 004.6

ANTI-PLAGIARISM SYSTEMS

Tishuk A. G., student
Narkovich A. M., student
Scientific supervisor –Lapko O. A., senior lecturer
English language department №1
Belarusian National University of Technology
Minsk, Republic of Belarus

The modern world is saturated with countless digital and analog systems, and almost all of them make life much easier for a person, but for most people it is mystery how these systems work. This article discusses the principle of operation of relevant systems in the field of science and education: anti-plagiarism systems.

In simple terms, the anti-plagiarism algorithm is as follows: the program compares a given text with the available database using the shingle method (where a shingle is a fragment of a text, a collection of words in the amount determined by the system) and reflects the result of the revision in the report. For example, the sentence: “I am a student of the Belarusian Technical University” anti-plagiarism will check with shingles in 3 words, and then will compare each with the available libraries of information. The first shingle is “I am a Belarusian student”, the second is “a student of the Belarusian Technical University”, and etc. If any of them exists in the anti-plagiarism database, then the text is considered unoriginal and is marked in red in the report.

The prototype of such systems was created in 2005 by domestic programmers independently from foreign companies. Its operation was based on Microsoft SQL server (a database Management System), but it worked very slowly, so processing documents in excess of several million was problematic. The programmers solved the problem in this way: the development of their own search engine began. Now the anti-plagiarism program is based on two elements: a software package written in C# and Python, and a database that this system will use to search for matches with the document being checked [1].

In fact, to start working with text, you need to perform an equally important operation: extract text from a file of a certain data format for subsequent processing. Along with the text from the document, the program

extracts the code of the location of words on the page, since the system is able to identify and add non-unique elements to the report.

Russian anti-plagiarism can extract text from almost all known formats, however, it is worth clarifying that the paid versions of the program have significantly more features than their free counterparts: only two txt and pdf formats are available for customers of free services, and this list is much broader for users of paid accounts. This feature can be traced to other functional components, and the difference between the resources of the systems is considered on the Russian software products anti-plagiarism Ru (free alternative) and anti–plagiarism HEI (paid). This way, anti-plagiarism Ru can compare texts and search for borrowings only from open sources - the Internet. The Internet is accepted as a separate search module (Databases and libraries are also called modules).

While the anti-plagiarism University has access to other modules: *The ring of Universities*, It allows you to analyze information from all student papers entered by an individual university across the country in this database; *Elibrary*, This collection contains about 13 million scientific articles and abstracts; *Wiley (Rein)*, The publishing house includes tens of thousands of encyclopedias and reference books; *The State Library*, All known scientific research materials are available: abstracts, doctoral dissertations, patents and even medical ones.

After extraction, the next stage of the Anti-Plagiarism work is determined by technical circumventions (Technical circumventions are called intentional operations aimed at unreasonably increasing the originality of the text) of the Anti-Plagiarism. Let's look at how anti-plagiarism deals with them.

One of the easiest ways to work around this is to replace characters that are similar in spelling to the original ones, otherwise they are called homoglyphs (Russian a, English a, α, à, å, å). To prevent this circumvention, the language of the suspected text and a possible list of homoglyphs for the character are determined, and then, as a result of certain manipulations, the system will restore the correct characters in the correct language and notify about the circumvention attempt. The software package simply destroys other inconspicuous characters during verification (•; `; °; ™). The procedure for detecting invisible text is a bit more complicated. First, the location of words and symbols on the page is set, they differ programmatically by a special code value, and then the variance of text and background colors is mathematically calculated (where the variance

is the difference between their numeric values). If the variance value does not match the set range, there is a uniform color in the analyzed text, that is, a crawl attempt. In case of text substitution with an image, the system is equipped with optical text recognition, since only a text is extracted at the previous stage, however, a symbolic pass must be filled in for the complete content. There are still a lot of technical workarounds, some related to file conversion, and some more complex ones related to changing the structure of the page code [2].

Now we will create a complete algorithm for the search engine (a system that analyzes billions of modules and compares the text with the source). The idea implemented in the search procedure was proposed by Ilya Segalovich and Yuri Zelenkov. First, sentences are divided into words, numbers and punctuation marks are ignored, then these words are lemmatized (where lemmatization is the transformation of a word into its initial grammatical form, different parts of speech have their own lemma defined, for verbs it is the infinitive, for nouns it is the nominative singular, for adjectives it is the nominative singular of the masculine gender). After that, the words are digitized and given certain integer values. This is done by hashing (a method of converting string values to numeric). Next, arrays of cached data in the size of several words are analyzed, and they are called shingles. Now the found loans are entered into a special huge hash table. It is stored on an ssd, which significantly reduces the processing time. Matched arrays of numbers are sorted in descending order: by the number of shingles found in a single source or document, and sources with which there are several matches are included in the report.

The search engine needs to work fast because it needs to browse billions of documents. Using hashes makes it possible to perform comparison operations more efficiently, saves memory, and stores digital images of source documents rather than texts, without violating copyrights.

References

1. Технические обходы системы антиплагиат: – URL: https://habr.com/ru/companies/antiplagiat/articles/480580/ (date of access: 08.03.2025).

2. Так устроен поиск заимствований в Антиплагиат: – URL: https://habr.com/ru/companies/antiplagiat/articles/429634/ (date of access: 09.03.2025).