UDK 004.052.42

THE SHINGLE ALGORITHM

K.O. Nekhaienko[1], *O.M. Kryvonos[2]*

[1] the student of English Language and Applied Linguistics Department, Zhytomyr Ivan Franko State University, Zhytomyr, Ukraine

[2] the associate professor, PhD of Computer Science and Information Technology Department, Zhytomyr Ivan Franko State University, Zhytomyr, Ukraine

There is an important condition for any text posted on the internet - no plagiarism. The higher the percentage of uniqueness, the better, of course. Anyone who writes articles should understand what shingles are and how the algorithm for finding duplicates works. By understanding how programmes and services determine uniqueness, it is easier to improve text originality and pass the check for duplicates.

The idea of finding and checking for duplicate texts was invented by Israeli scientist Udi Manber in 1994. One of Google's most distinguished scientists, Andrei Zari Broder, took Udi Manber's idea of checking text uniqueness to its logical conclusion in 1997 and coined the name "shingle algorithm".

So how does it work? Let's compare two similar texts to see how this algorithm works. Each text will be subjected to the following operations:

1. Canonisation of the text. Before any text is checked, it is reduced to a so-called "single normal form". Conjunctions, prepositions, various html tags and any other markup and punctuation are removed from the text to be checked. In some cases adjectives are removed. When determining originality, adjectives are not given importance. All nouns are reduced to singular and nominative case. Some programs leave only the roots from the nouns. After canonicalization we get a rubbish free text, ready to search for duplicates.

2. Dividing text into sequences. Sequences of consecutive words in the prepared text are sequenced. The sequences in the text are not separated sequentially, but overlap each other. Cutting the whole prepared text, we obtain a number of sequences equal to the number of words in the prepared text minus the length of the shingle + one.

3. Selection of values for comparison. The algorithm for determining the uniqueness of the text selects a certain number of shings randomly. There is a technical problem with random selection of shings. For better comparison of texts, it is necessary to increase the number of sequences for comparison. This in turn exponentially increases the number of operations and is reflected in performance[4].

4. Shingle comparison and result determination. Randomly chosen sequences are compared and all matches are counted. The ratio of all matches is the result of the check[1].

Systems for determining uniqueness (texru, advego, anti-plagiarism) uses a somewhat simplified version of the algorithm. The algorithm for finding duplicates is based on a certain tiling length.

Let's get to the root of what a 'shingle' is. A shingle is a sequence of text fragments of a given length, by which services and programs determine the uniqueness of documents[3]. For more detailed understanding of the algorithm's principle, it is necessary to understand such a notion as the shingle length.

Shingle pitch or length is the number or order of words used by algorithms to determine uniqueness. The number of words can range from 2 to 10. The smaller the number of words in the shingle, the more accurately the originality of the text.

For example, if a text uniqueness service uses a shingle length of 3, it means that will be checked every third word in the text.

Having understood how the algorithm works, this knowledge has to be put into practice somehow. The text can be made more unique simply by changing the shingle length. For example, if it is known that the algorithm uses a shingle length of 2, then it is necessary to

change every second word in the text. Verification service will find new elements in the text and increase the percentage of originality[2].

But this method of writing text for the site should not be used, as it may not coincide with the policy of the site, as will not be taken into account other important conditions of the site.

For writing a text for the site it is necessary to have a complete terms of reference, which includes the collection of key phrases, words defining the subject of the text, the definition of the length of the text, etc.

Reference list:

1. Broder A. Algorithms for duplicate documents [Електронний ресурс] / Andrei Broder. – 2005. – Режим доступу до ресурсу: http://www.cs.princeton.edu/courses/archive/spr05/cos598E/bib/Princeton.pdf

2. Chowdhury A. Collection statistics for fast duplicate document detection. ACM Transactions on Information Systems (TOIS) [Електронний ресурс] / A.Chowdhury, O. Frieder, D. Grossman, M. Grossman // Vol. 20, Issue 2. – 2002. – Режим доступу до ресурсу: http://ir.iit.edu/~dagr/2002collectionstatisticsfor.pdf.

3. Uwamahoro G. Efficient Algorithm for Near Duplicate Documents Detection [Електронний ресурс] / G. Uwamahoro, Z. Zuping // JCSI International Journal of Computer Science Issues, Vol. 10, Issue 2, No 2. – 2013. – Режим доступу до ресурсу: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1071.9616&rep=rep1&type=pdf .

4. Valls, E. & Rosso, P., “Detection of near-duplicate user generated contents: the SMS spam collection”, in Proceedings of the 3rd international workshop on search and mining user-generated contents, 2011, pp. 27–34