Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 103 (2017) 421 – 425

Procedia Computer Science

XIIth International Symposium «Intelligent Systems», INTELS'16, 5-7 October 2016, Moscow, Russia

Detecting near-duplicates in russian documents through using fingerprint algorithm Simhash

N. Rezaeian, G.M. Novikova*

RUDN University, 6 Miklukho-Maklaya str., Moscowf 117198, Russia

Abstract

Plagiarism is one of the major problems in the age of communication. In many languages such as English, this issue is seriously of high importance and many powerful devices have been invented to prevent this problem from occurring. This article aims at discovering plagiarism in Russian texts based on fingerprint algorithm. The fingerprint algorithms have high speeds in finding out the plagiarism due to the compact features it creates and purely because of the comparison of these properties between original documents and dubious documents. Increasing the power and accuracy of plagiarism discovery, there must be elimination of general words and word rooting before pre-processing applications such as words separation, numbers replacement, and homogenization. In this article, four Simhash algorithms have been used. The implementation of these algorithms confirmed on 800 articles with the scientific topics was found to have satisfactory results.

Peer-review under responsibility of the scientific committee of the XIIth International Symposium "Intelligent Systems"

Keywords: plagiarism; fingerprint algorithm; Simhash.

1. Introduction

With an increase in the accessibility easiness to the data in the network, the plagiarism has been one serious problem. This problem has caused the authors and publishers to have less confidence to the internet. Plagiarism in the texts is divided into two types, the source code plagiarism and free text plagiarism. According to the limitations and the keywords of programming languages, the investigation of this type of plagiarism is easier than a text.

The text plagiarism has different forms that Maurer categorizes different forms of plagiarism as following[1]:

- Copy and paste or the plagiarism of word to word, or in a way that the content of copied text is from one or several sources. The copied content can be changed a little.

* Corresponding author.
E-mail address: novikova_gm@mail.ru

Peer-review under responsibility of the scientific committee of the XIIth International Symposium "Intelligent Systems"
doi:10.1016/j.procs.2017.01.006

- Change in grammar through using synonyms, movement of lines of the original test or the expression of sentences differently
- Using exact phrase without putting the text in quotation mark
- Adding untrue sources or not stating them
- Translating a text without stating the original reference

According to aforementioned categorization, the plagiarism discovery devices are divided into three fundamental types[1,2]:

- Recognition of the writing style of a writer and the discovery of every incompatible change of the style
- Comparison of several documents and the discovery of their common and similar parts, mostly used method.
- Reception of a document as the entrance and to find the documents copied on the web pages. The following figure shows this categorization.

In this article, the second group is focused. The posed method compares one document with a set of documents based on syntax. The semantic methods are sensitive to the sentence changes, the use of synonyms instead of the words themselves, generally rephrasing the sentences. Therefore, these methods are smarter in the recognition of plagiarism, but the discovery of plagiarism through utilizing the method of this group requires a lexical database of the language word such as Wordnet in English language. In the following, we will explain the important required stages of fingerprint algorithms.

2. Preprocessing

Preprocessing includes the performance of some actions on the text, improving the outcomes of similarity detection algorithms. These actions increase the accuracy and decrease the time of investigation.

2.1. Tokenization

This part must have the ability of sentence recognition in the input text regarding the sentence divider characters in Russian language[3]. To create this device, first all symbols, characters, especially syntactic rules which break the sentences must be identified. Since the sentence is basic in many language processes, the accurate outcome of this section is of high importance.

2.2. Replacement numbers

To replace and eliminate the numbers, a method[4] so called Token-making, is employed. This method is an appropriate method to recognize the similarities especially, in the computer programs. The Token-making algorithm replaces the elements of program with the unit tokens. For example, every ID is replaced by the token $< ID >$. Or every numerical value with $< value >$. Now if a program has a statement in the form of $a = b + 4$, it will be replaced by the line or string of $< ID >=< ID > + < value >$. Therefore, if we change a variable name, there is no change in the translation.

2.3. Remove Stop words

In this stage, the general words will be omitted[3]. This makes the investigation operation faster. The general words are the ones whose importance in the sentences are trivial. Some think that the general words ("O","C","CO","B",...) are as the same as high frequency and repeated words while the repeated words do not include all of general words.