I.J. Information Technology and Computer Science, **2017, 12, 64-71**
Published Online December 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2017.12.08

Plagiarism Detection System for the Kurdish Language

Karzan Wakil
University of Human Development-Iraq
E-mail: karzanwakil@gmail.com, Karzan.wakil@uhd.edu.iq

Muhammad Ghafoor, Mehyeddin Abdulrahman and Shvan Tariq
University of Human Development-Iraq
E-mail: Muhammad@uhd.edu.iq, muhyialdeenclick@gmail.com, shvanta52@gmail.com

Received: 06 September 2017; Accepted: 22 September 2017; Published: 08 December 2017

***Abstract*—**One of the serious issues is plagiarism, especially in the education field. Detecting the plagiarism became a challenging task, particularly in natural language texts. In the past years, some plagiarism detection tools have been developed for diverse natural languages, mainly English. Language-independent tools exist as well but are considered as too restrictive as they usually do not consider specific language features. The problem is there is no plagiarism Detection system for the Kurdish language. In this paper, we introduce a new system for plagiarism detection for Kurdish Language, based on n-gram algorithm, our system can detect the word, phrases, and paragraphs. Moreover, our system effectiveness for detect plagiarist texts in localhost and online especially in Google search engine. This system is more useful for the academic organizations such as schools, institutes, and universities for finding copied texts from another document.

***Index Terms*—**Plagiarism Detection, Plagiarism Detection System, N-Gram, Kurdish Language, Theft.

I. Introduction

Plagiarism is described as the illegal use or close imitation of the language and thought of authors and their representation as one's original work [1]. It involves literary cheating, stealing, by copying the words or ideas from someone else and passing them off as one's own without recognizing the origin. Many people think of plagiarism as copying another's work or borrowing someone else's original ideas. However, terms like "copying" and "borrowing" can disguise the seriousness of the offense [2].

Plagiarism becomes one of the most critical issues for universities, schools, and researchers [3]. It is so accessible from the internet and due to using the advanced search engine to find documents or journals by scholars. Some of the researchers are just copying and pasting others works without mentioning the source of the owner's documents. Several kinds of plagiarism exist, including direct copying of phrases or passages from a published text without citing the references, plagiarism of concepts, references, and authorship. There are other types of plagiarism, such as translating content to another language, presenting the same content in the shape of other media like images, videos, and texts, and using program's source code without permission [4].

Plagiarized document detection performs important roles in many applications, such as file management, copyright protection, and plagiarism prevention [5]. Plagiarism can take one of the common types such as copying of the whole or some parts of the document, rewording the same content in different words, using others' ideas or referencing the work to mistaken or non-existing sources [6]. Other ways of plagiarism cover translated plagiarism wherein the content is translated and used without referencing the original work, artistic plagiarism in which different media such as pictures and videos are used to present other's work without proper citation [6].

Ibrahim et al., 2017 presented a good literature review about Arabic Script Languages (ASL). In their review, they introduced the plagiarism detection techniques per year. They reviewed all publications from 2009 to 2017, as their results plagiarism detection techniques widely used for that language. Moreover, Ibrahim et al. presented the techniques that used for ASLs based on their review, most techniques used for Arabic language, then Persia and Urdu languages, but there is no publication exist for the Kurdish language [7].

In the recent years the Kurdish language used for writing in academia and educations, the problem is, most of the works are in texts and documents, and plagiarism has been done for some languages such as English, Arabic and much more. Detecting plagiarism is excellent to judge on student's mark' and work, unusually for some students and scholars who are strictly prohibited from rewording, cheating, rephrasing, or restating without referencing. In this paper, we prepare a plagiarism system for the Kurdish language. There are many techniques used to finding plagiarism, but until now there are not any techniques used for Kurdish texts, in this paper, we are using n-gram for finding plagiarized texts.

The paper is organized as follows: The Section II explains the background work on plagiarism detection algorithms and implemented in different languages. Section III explained types of plagiarism. The Section IV prepared framework for designing the system.

In section V we demonstrate the designing of the system to detect Kurdish texts. Section VI presents concluding remarks and future works.

II. Related Work

There are many works done for finding plagiarism detection in different languages with different techniques; in this section, we explain essential works for Arabic script language. Moreover, we explain the recent works that used plagiarism detection algorithms especially N-gram algorithm.

Several state-of-the-art techniques for plagiarism detection have been extensively studied by Meuschke and Gipp [8], emphasizing the performance of each of them. On the other hand, Riad et al. [9] have investigated the different methods for plagiarism detection, considering their applicability and appropriateness for Arabic natural language text. Furthermore, in-depth investigation of cross-language plagiarism detection methods on a new paper published [10]. In these literature works some systems completed for different languages especially English, Arabic, Urdu and so on but there are no works for the Kurdish language [7]. The Kurdish language used Arabic letter after changing some letters. Therefore, we follow papers that used Arabic letter such as Arabic, Persia, Urdu.

A. S. Hussein, 2015 in an excellent work, presented an innovative similarity estimation method devoted to Arabic text documents. The method is based on modeling the relationship between documents, under consideration, and their n-gram phrases. PoS tagging is applied on the examined documents to support in resolving the morphological ambiguity during text normalization. A new NLP based method is used for text indexing and stop-words removal. Heuristic pair-wise phrase matching algorithm is introduced to build the documents TFIDF model, considering substitution of words with their synonyms. Finally, the hidden associations between documents and their n-gram phrases are investigated using the LSA, employing the SVD. The proposed method exhibited robust capabilities in discovering literal similarity. Also, it could be considered as a dangerous step towards detecting intelligent similarity [11].

W. Adouane and S. Dobnik, 2017 presented a system for identifying the language at the word and long sequence levels in multilingual documents in Algerian Arabic. They described the data and the different methods used to train the system that can identify the language of words in their context between Algerian Arabic, Berber, English, French, Modern Standard Arabic and mixed languages (borrowings). The system achieves an excellent performance, with an overall accuracy of 93.14% against a baseline of the majority class of 55.10% [12].

Although in another paper applies two n-gram based techniques of author attribution to Urdu poetry. A data corpus consisting of all the Urdu works of Iqbal was chosen, and the corpus was cleaned up using different string replacement rules and an existing collation software. Some poems were separated as the test data, and the rest of the data was used as input to train the system. He applied one existing and one modified method based on n-gram probabilities. Upon observing the results of the system on couplets of Iqbal and two other poets, it was found that the author profile matching based method performed poorly. The simpler method based on interpolation of n-gram probabilities proposed by us successfully distinguished between the works of Iqbal and other poets[13].

Another type of work on this field is preparing the tools and comparison between them for finding performance. Menai, 2012 presented a plagiarism detection tool for comparison of Arabic documents to identify potential similarities. The tool is based on a new comparison algorithm that uses heuristics to compare suspect documents at different hierarchical levels to avoid unnecessary comparisons. Then he made a comparison to evaluation tool's performance [14].

Another recent work focused on the accuracy of the Arabic text categorization by Hussein et al., 2016. They proposed a new approach to enhance the accuracy of the Arabic text. It is based on a new features representation technique that uses a mixture of a bag of words (BOW) and two adjacent words with different proportions. It also introduces a new features selection technique depends on Term Frequency (TF) and uses Frequency Ratio Accumulation Method (FRAM) as a classifier. In their work, they collected three data sets of different categories from online Arabic documents for evaluating the proposed approach. The highest accuracy obtained is 98.61% by the use of normalization [15].

Previous works on plagiarism detection system successfully exist for different languages, among them languages that used Arabic letters. However, some of tools and systems could not find texts especially in meaning but for succeeding to find an exact copy. However, there is no work for Kurdish texts, finding the exact copy is a start point to plagiarism detection in the Kurdish language.

III. Plagiarism Taxonomies

The taxonomy of plagiarism evolves out of the systematic identification of possible intra-textual manipulations. Alzahrani et al. offer an exemplary model that is mostly based upon a qualitative assessment of institutional findings during the review of student submissions. As evidenced in this visual model (Fig.1), there are two dominant cases of plagiarism, the literal and the intelligent. Literal plagiarism is tied to purposeful copying or manipulation of textual outputs either in whole or part without providing due credit to the originator. Within the framework of literal plagiarism, plagiarists are unlikely to spend significant time

attempting to 'hide their academic crime. Intelligent plagiarism, on the other hand, is much more difficult to detect and involves the manipulation of text, translation of foreign copy, or the adoption and ownership of others ideas, theories, or concepts. This form of plagiarism involves purposeful deception by students/academics and involves an attempt to obfuscate or change the original work in ways to prevent detection [16].

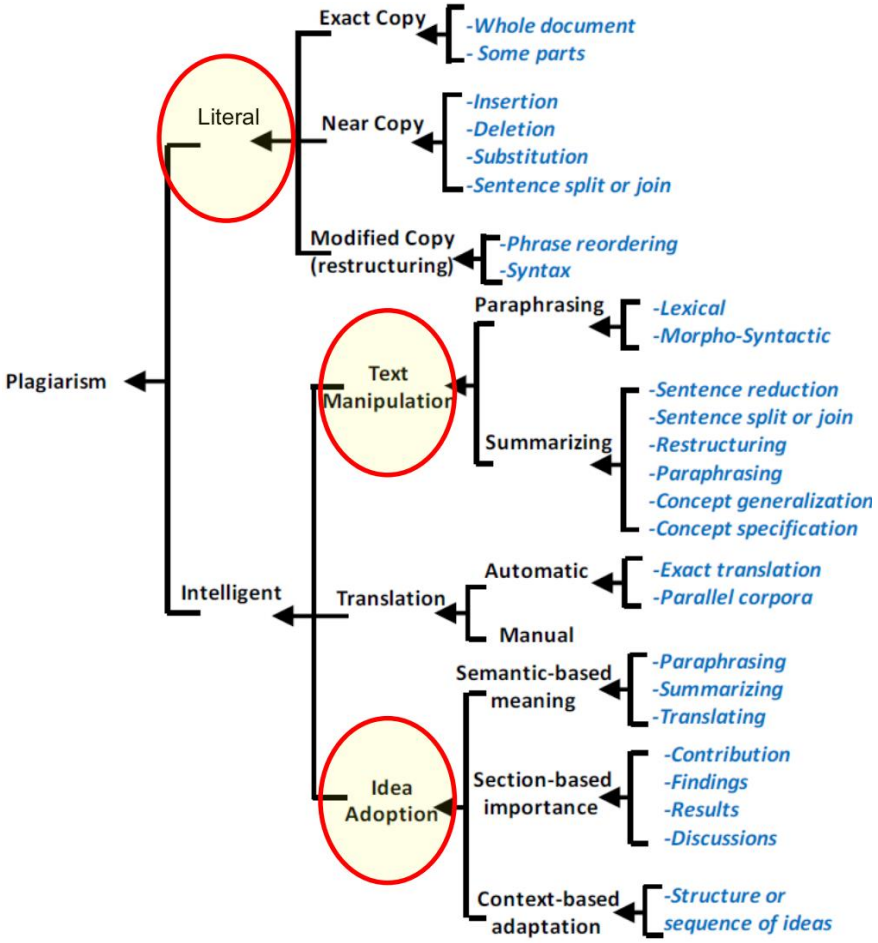

Fig.1. Taxonomy of plagiarism

The Kurdish Language is new language in the academic area. We cannot find many articles on the web compared with Arabic and English Languages. Also there is no plagiarism detection systems or tools for the Kurdish language. As the first step, we need to work on literal types, because those types are easy to copy and paste for everybody.

IV. Methodology and System Framework

In this section, we explain framework of our system. Furthermore, we prepare a methodology for plagiarism detection system for the Kurdish language and our method to enhancement the system.

The system framework consists of four steps as shown in Fig.2, in step 1 we explain plagiarism detection algorithms that used for finding similarity between texts, in this study we use n-gram algorithm. Step 2 describes the Kurdish Language structure; the Kurdish language Structure easer compared other languages. However, the Kurdish language morphology becomes to challenge to find texts after some paraphrasing. In step 3 we use PHP codes to compare the texts. Step 4 implement n-gram inside PHP codes to find similarity between texts based on n-gram.

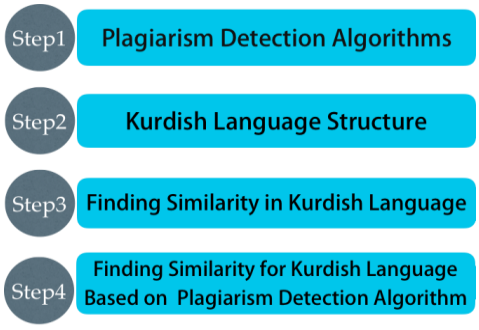

Fig.2. System Framework

Kurdish language constructed from Arabic letter after changing some letters. The number of letters in the Kurdish language are 33 letter, so creating word in the Kurdish language too similar that is why is not that easy to detect. For example; (گوڵ،گول), (بڵاو، پلاو), (کۆن، کون), as we see there is a lot of similarity between of them based

on letters, but they are such a dissimilarity in meaning.

In [17] explained n-gram class as a number from 0 to m−1 such that the class labeled 0 involves the least frequent n-grams and the class labeled m−1 contains the most frequent n-grams in a document. If m > 2, classes between 0 and m−1 will contain n-grams with intermediate frequency levels. Concretely, to assign the n-grams of a given document to m classes, first, the document is represented by a 2 × l matrix (l is the total number of n-grams), where the first row contains the n-grams ngi (i =1..l) and the second one contains their number of occurrences freqi (raw frequency).

Let max_freq denotes the maximum frequency, so:

$$\text{max\_freq} = \text{argmax freqi ; i=1..l} \quad (1)$$

Then, the class of a n-gram ngi is computed as follows:

$$\text{Class ngi} = \text{Log base (freq i);} \quad (2)$$

Given that:

$$\text{base} = max\_freq\ m - 1\ . \quad (3)$$

By computing the base of the logarithm as shown in the equation (3). The most frequent n-grams (i.e. the n-grams with the maximum number of occurrences) will be in the class m−1, and the least frequent n-grams will be in the class 0, and the n-grams with intermediate levels of frequency will be in the classes between 0 and m−1.

Our model is: N-gram + similarity + separation sentences

Separation sentences the brackets are converted into spaces and the commas are converted to points, and the points remain as they are.

V. Result and Discussion

In this Section we analysis the Kurdish language sentences and documentations, analyzing plagiarism detection algorithms especially N-gram algorithms for the Kurdish language, then we apply N-gram on the Kurdish language. Finally, we implement the web and evaluation the result.

The code that detects plagiarism is by reading the first text and the second text then compares them through the similar sentences found in the two texts, the result is shown in the form where the result shows the full text on the two colors of the text. The red color indicates that the sentences are intact. The text whose color is yellow indicates that these sentences are present in both texts and that it is taken and indicates the complete similarity between the two sentences as shown in Fig.3.

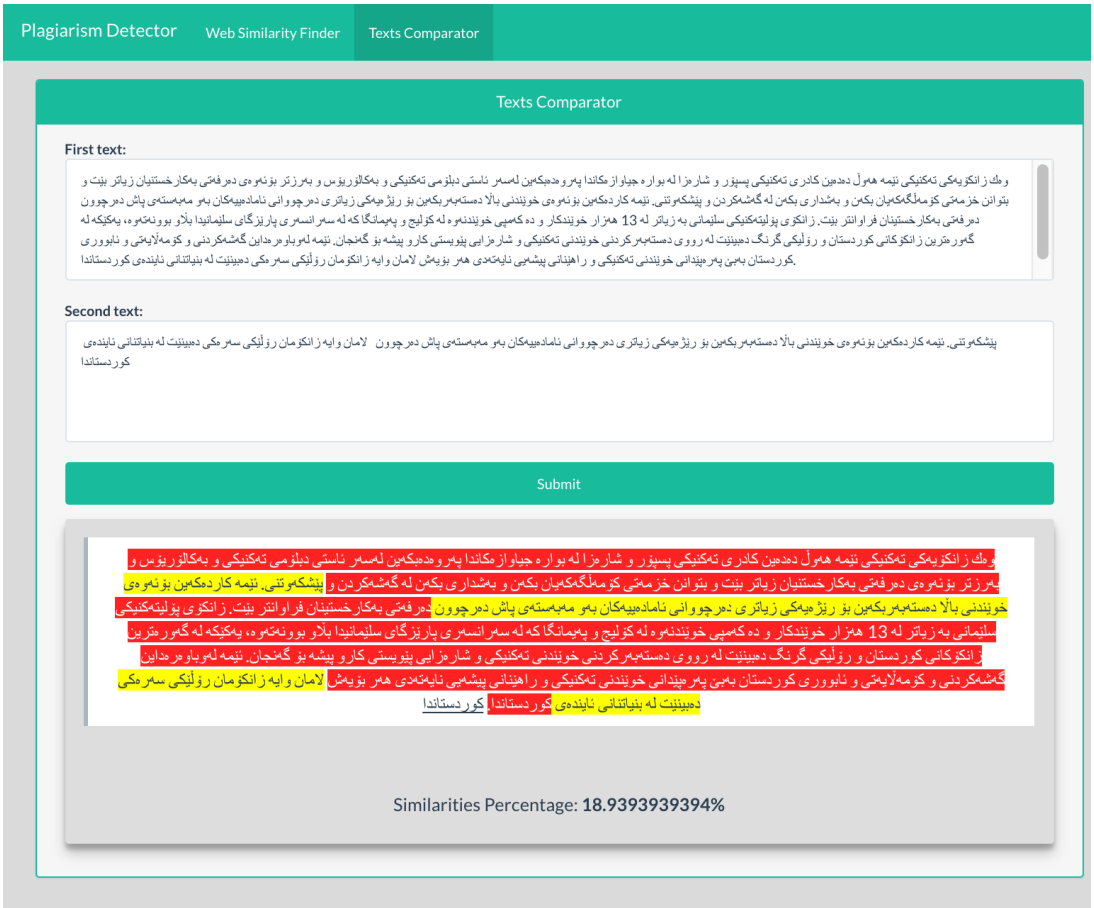

Fig.3. Finding similarity between two texts in localhost

The main interface of the website contains a menu bar consisting of web similarity finder and text comparator. The website supports the design of the bootstrap. It fits the measurements of multiple screens without showing a defect, in the design from the web similarity finder section. We upload a text file And then click on the submit and the site do Request to search for each sentence on Google and bring the links found in Google as shown in Fig.4. after submitting the text, our software search in the google for finding original text and present the link to the original text. Sometimes the target text published in the more websites, our software can find and present all links as shown in Fig.4.

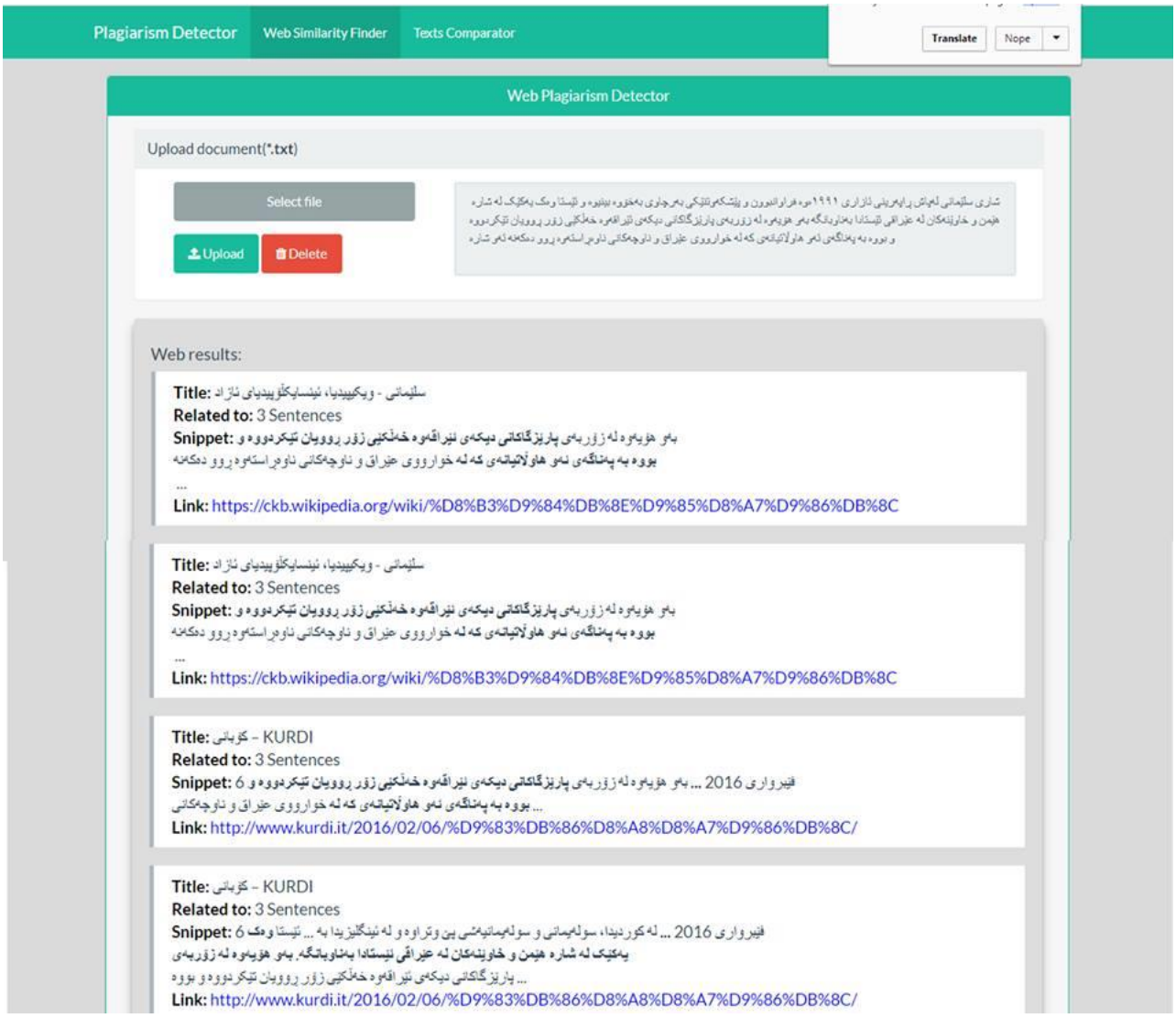

Fig.4. Finding similarity between a text and web

The results that we have reached are finding similarity in two ways; text to text in local and text to Google web engine online. The first one is work with two text file and compares them then finding similarity between them. The second one is uploading a file and search in Google engine for finding similarity between text and original text that published online.

In the following we present essential codes that used for creating our system, we used PHP codes for finding similarity between the texts as shown in Fig.5, Fig.6, Fig.7, and Fig.8.

In Fig.5, the brackets are converted into spaces, and the commas are converted to points.

As shown in the codes, we used (str_replace) function for the following problems. The brackets are converted into spaces, and the commas are converted to points, and the points remain as they are.The explode function allows us to break a string into smaller text with each break occurring at the same symbol. This symbol is known as the delimiter.


```
// Declaring Text1
        $text1 = test_input($_POST["text1"]);
        $text1 = str_replace("(", " ", $text1);
        $text1 = str_replace(")", " ", $text1);
        $text1 = str_replace(",", ".", $text1);
        $sentences1 = explode('.', $text1);

// Declaring Text2
        $text2 = test_input($_POST["text2"]);
        $text2 = str_replace("(", " ", $text2);
        $text2 = str_replace(")", " ", $text2);
        $text2 = str_replace(",", ".", $text2);
        $sentences2 = explode('.', $text2);
```

Fig.5. Code for Convert

After finding spaces and points by using above codes (Fig.5). We add all sentences to a matrix, each element in the matric presents a new sentence. Then our system makes a comparison between matrix's elements. When an element from array1 equal to array2 meaning this sentence was copied from text2, as presented in Fig.6.

```
function diff($old, $new)
        {
            $matrix = array();
            $maxlen = 0;
            foreach ($old as $oindex => $ovalue)
           {
                $nkeys = array_keys($new, $ovalue);
                foreach ($nkeys as $nindex)
            {
                    $matrix[$oindex][$nindex] =
isset($matrix[$oindex - 1][$nindex - 1]) ?
            $matrix[$oindex - 1][$nindex - 1] + 1 : 1;
            if ($matrix[$oindex][$nindex] >
$maxlen)
            {
                $maxlen = $matrix[$oindex][$nindex];
                $omax = $oindex + 1 - $maxlen;
                $nmax = $nindex + 1 - $maxlen;
                    }
                }
            }
 if ($maxlen == 0) return array(array('d' => $old, 'i'
=> $new));
            return array_merge(
                diff(array_slice($old, 0, $omax),
array_slice($new, 0, $nmax)),
                array_slice($new, $nmax, $maxlen),
                diff(array_slice($old, $omax +
$maxlen), array_slice($new, $nmax + $maxlen)));
        }
```

Fig.6. Code for finding similarity between two texts

One of the important points in these systems is showing the percentage of similarity. Some of systems count the percentage based on words, but others used numbers of similar sentences for counting percentage. In our system, we count the similarity percentage based on words, because counting words can better detect plagiarism percentage. In our code after counting the words and divided by original words then multiply 100. As presented in Fig.7.

```
Function htmlDiff($old, $new)
        {
            $ret = '';
            $diff = diff(preg_split("/[\s]+/", $old),
preg_split("/[\s]+/", $new));
            $all_words = 0;
            $differences = 0;
            foreach ($diff as $k) {
                $all_words++;
                if (is_array($k)) {
                 if (!empty($k['d'])) $differences +=
count($k['d']);
                    $ret .= (!empty($k['d']) ? "<del>" .
implode(' ', $k['d']) . "</del> " : '') .
                        (!empty($k['i']) ? "<corrected>" .
implode(' ', $k['i']) . "</corrected> " : '');
                }
            else {
                    $ret .= '<ins>' . $k . ' </ins>';
                };
            }
            return [$ret, '<br><br><h4>Similarities
Percentage: <strong>' . (((count(preg_split("/[\s]+/",
$old)) - $differences) / count(preg_split("/[\s]+/",
$old))) * 100) . '%</strong></h4>'];
        }
        ?>
      </div>
      </div>
   </div>
</div>
```

Fig.7. Code for finding similarity based on the algorithm

Most of students and scholars copying the texts from websites on the internet, so we need to compare the similarity of the texts with online texts through google search engine. In Fig.8 we compare our texts with online search engine texts. Fig.4 shows the implementation PHP codes that present in Fig.8. The codes search for similar texts; sometimes the search engine can find similar texts in several sources.


```
Class Google_Item
{
   public $title, $content, $link, $sentence_key,
$repeated;
}
$google_items = [];
$i = 0;
foreach ($res as $item) {
   $_items = google_search($item, $i);
   foreach ($_items as $_item) {
      array_push($google_items, $_item);
   }
   $i++;
}
foreach ($google_items as $item) {
   $__item = $item;
   $item->repeated =
count(array_filter($google_items, function ($n) {
      global $__item;
      return ($__item->link == $n->link) ? true :
false;
   }));
}
```

Fig.8. Code for finding texts in Google

VI. Conclusion and Future Work

In this paper, an innovative similarity estimation method devoted to Kurdish text documents is presented and studied in detail. The method is based on sentences and words, under consideration, and their n-gram phrases. However, our project can find copy text In the Kurdish Language also used for finding text in different languages such as; Arabic, Persian, and English. However, our focus extremely for the Kurdish language because currently there is no tool or any websites can provide plagiarism detection for the Kurdish language.

In this project, we made a website for finding similarity between two texts in the Kurdish language as the first step. Furthermore, our website can compare a local text from a computer with search engine websites such as Google meaning that can locate the similar text in google. The website consists of a system for finding similarity by using PHP code and JavaScript; we improve our system by using N-gram technique for detecting the sentences. Our system separates the paragraphs for sentences after each full stop, if we forget full stop the system count every ten spaces as one sentence and remove every special character in the sentences. We recommend for the researcher to enhance our system by; using different techniques such as Tree gram and RST, and improve our system based on morphology sentences.

References

[1] plagiarism.com, "glatt plagiarism services," 2017.

[2] UKessays, "A Survey Of Plagiarism Detection Methods Information Technology Essay," 2015.

[3] Plagiarism.org, "What is Plagiarism?" 2015.

[4] A. Jadalla and A. Elnagar, "A plagiarism detection system for Arabic text-based documents," in *Pacific-Asia Workshop on Intelligence and Security Informatics*, 2012, pp. 145-153: Springer.

[5] C. Lyon, R. Barrett, and J. Malcolm, *Plagiarism is easy, but also easy to detect*. Ann Arbor, MI: Scholarly Publishing Office, University of Michigan Library, 2006.

[6] R. Lukashenko, V. Graudina, and J. Grundspenkis, "Computer-based plagiarism detection methods and tools: an overview," in *Proceedings of the 2007 international conference on Computer systems and technologies*, 2007, p. 40: ACM.

[7] R. Ibrahim, S. Saeed, and K. Wakil, "Plagiarism Detection Techniques for Arabic Script Languages: A Literature Review," *Kurdistan Journal of Applied Research*, vol. 2, no. 3, 2017.

[8] N. Meuschke and B. Gipp, "State-of-the-art in detecting academic plagiarism," *International Journal for Educational Integrity*, vol. 9, no. 1, 2013.

[9] A. Riad, F. Farahat, A. Asem, and M. Zaher, "Studying different methods for plagiarism detection," *International Journal of Computer Science*, vol. 2, no. 5, pp. 147-154, 2013.

[10] J. Ferrero, L. Besacier, D. Schwab, and F. Agnes, "Deep Investigation of Cross-Language Plagiarism Detection Methods," *arXiv preprint arXiv:1705.08828*, 2017.

[11] A. S. Hussein, "Arabic document similarity analysis using n-grams and singular value decomposition," in *Research Challenges in Information Science (RCIS), 2015 IEEE 9th International Conference on*, 2015, pp. 445-455: IEEE.

[12] W. Adouane and S. Dobnik, "Identification of Languages in Algerian Arabic Multilingual Documents," *WANLP 2017 (co-located with EACL 2017)*, p. 1, 2017.

[13] A. A. Raza, A. Athar, and S. Nadeem, "N-Gram Based Authorship Attribution in Urdu Poetry," in *Proceedings of the Conference on Language & Technology*, 2009, pp. 88-93.

[14] M. E. B. Menai, "Detection of plagiarism in Arabic documents," *International journal of information technology and computer science (IJITCS)*, vol. 4, no. 10, p. 80, 2012.

[15] M. Hussein, H. M. Mousa, and R. M. Sallam, "Arabic Text Categorization Using Mixed Words," 2016.

[16] S. M. Alzahrani, N. Salim, and A. Abraham, "Understanding plagiarism linguistic patterns, textual features, and detection methods," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 2, pp. 133-149, 2012.

[17] I. Bensalem, P. Rosso, and S. Chikhi, "Intrinsic Plagiarism Detection using N-gram Classes," in *EMNLP*, 2014, pp. 1459-1464.

Authors' Profiles

Karzan Wakil: Lecturer at the University of Human Development-Iraq and Sulaimani Polytechnique University-Iraq. Received BSc. Degree in Computer Science from Salahaddin University-Iraq-2006 and M.Sc. in Computer Science from University Technology Malaysia (UTM), Malaysia, 2013. Currently, he is PhD

student at UTM, Malaysia. His research areas are; Web Engineering, Software Engineering, Web Development, Model-Driven, Metamodel. Email: karzanwakil@gmail.com.

Muhammad Ghafoor: BSc Student at Computer Science in the University of Human Development (UHD)-Iraq, His research areas are Information Retrieval, Artificial Intelligence, Database, web programming, and Information System. Email:Muhammad@uhd.edu.iq.

Mehyeddin Abdulrahman: BSc Student at Computer Science in University of Human Development (UHD)-Iraq, His research areas are Information Retrieval, Artificial Intelligence, Database, web programming, and Information System. muhyialdeenclick@gmail.com.

Shvan Tariq BSc Student at Computer Science in University of Human Development (UHD)-Iraq, His research areas are Information Retrieval, Artificial Intelligence, Database, web programming, and Information System shvanta52@gmail.com.

How to cite this paper: Karzan Wakil, Muhammad Ghafoor, Mehyeddin Abdulrahman, Shvan Tariq, "Plagiarism Detection System for the Kurdish Language", International Journal of Information Technology and Computer Science(IJITCS), Vol.9, No.12, pp.64-71, 2017. DOI: 10.5815/ijitcs.2017.12.08

