ISSN: 0067-2904

Improved VSM Based Candidate Retrieval Model for Detecting External Textual Plagiarism

Mohannad T. Mohammed*[1], Nasreen J. Kadhim[2], Abdallah A. Ibrahim[2]
[1]College of Health And Medical Technology, Middle Technical University- Baghdad- Iraq
[2]College of Science University of Baghdad, Baghdad,Iraq

Abstract

A rapid growth has occurred for the act of plagiarism with the aid of Internet explosive growth wherein a massive volume of information offered with effortless use and access makes *plagiarism* − the process of taking someone else's work (*represented by ideas, or even words*) and representing it as other's own work − easy to be performed. For ensuring originality, detecting plagiarism has been massively necessitated in various areas so that the people who aim to plagiarize ought to offer considerable effort for introducing works centered on their research.

In this paper, work has been proposed for improving the detection of textual plagiarism through proposing a model for candidate retrieval phase. The model proposed for retrieving candidates has adopted the *vector space method* VSM as a retrieval model and centered on representing documents as vectors consisting of average term $tf - isf$ weights and considering them as queries for retrieval instead of representing them as vectors of term $tf - idf$ weight. The *detailed comparison* task comes as the second phase wherein *fuzzy semantic based string similarity* has been applied. Experiments have been conducted using PAN-PC-10 as an evaluation dataset for evaluating the proposed system. As the problem statement in this paper is restricted to detect extrinsic plagiarism and works on English documents, experiments have been performed on the portion dedicated to extrinsic detection and on documents in English language only. For evaluating performance of the proposed model for retrieving candidates, *Precision*, *Recall*, and *F-measure* have been used as an evaluation metrics. The overall performance of the proposed system has been assessed through the use of the five standard PAN measures *Precision*, *Recall*, *F-measure*, *Granularity* and $Plagdet$. The experimental results have clarified that the proposed model for retrieving candidates has a positive impact on the overall performance of the system and the system outperforms the other state-of-the-art methods. They clarified that the proposed model has detected about 80% of the plagiarism cases and about 90% of the detections were correct. The proposed model has the ability to detect literal plagiarism in addition to cases containing paraphrasing. Performance comparison has clarified that the proposed system is either comparable or outperforms the other baseline systems in terms of the five PAN evaluation metrics.

Keywords: External Plagiarism, vector space model, TF-IDF, TF-ISF, fuzzy similarity

نموذج استرجاع محسن مستند على VSM لاكتشاف الاستلال النصي الخارجي

مهند طه محمد[1]، نسرين جواد كاظم[2]، عبد الله عادل ابراهيم[3]
[1]كلية التقنيات الصحية والطبية، الجامعة التقنية الوسطى، بغداد، العراق
[2]كلية العلوم ,جامعة بغداد ,بغداد ,العراق

*Email: mohannad.tm@gmail.com

الخلاصة

مع التطور و الحجم الهائل من المعلومات المتوفرة على شبكة المعلومات(الانترنت) ، ساعد ذلك في تنامي ظاهرة الاستلال– عملية تبني اعمال شخص اخر سواء نصوص او حتى أفكار و اعتبارها عمله الخاص دون ذكر صاحب العمل الحقيقي– وجعلها سهلة التنفيذ. لذلك ولضمان الاصالة فإن عملية كشف الاستلال اصبحت ضرورية في العديد من المجالات ، حيث يتعين على الاشخاص الذين يرومون الاستلال بذل جهد أكبر لذلك. في هذه البحث ، تم اقتراح عمل لتحسين اكتشاف الاستلال النصي من خلال اقتراح نموذج لمرحلة استرجاع المرشحين. حيث اعتمد النموذج المقترح لاسترجاع المرشحين طريقة VSM للفضاء المتجه كنموذج استرجاع وركز على تمثيل المستندات كمتجهات تتكون من متوسط أوزان المصطلحات مقاسة باستخدام طريقة tf–isf واعتمادها كطلبات للاسترجاع بدلاً من تمثيلها كمتجهات تحوي أوزان المصطلحات مقاسة باستخدام طريقة tf–idf. وتأتي مهمة المقارنة التفصيلية في المرحلة الثانية، حيث تم تنفيذ مقياس التشابه الضبابي. و تم إجراء التجارب باستخدام PAN–PC–10 كمجموعة بيانات لتقييم النظام المقترح. ونظرًا لأن بيان المشكلة في هذا البحث مقصور على اكتشاف الاستلال الخارجي والعمل على المستندات الإنجليزية ، فقد تم إجراء التجارب على الجزء المخصص لكشف الاستلال الخارجي والوثائق باللغة الإنجليزية فقط. حيث تم استخدام Precision و Recall وF – measure كمقاييس لتقييم مرحلة استرجاع المرشحين. وقد تم تقييم الأداء العام للنظام المقترح من خلال استخدام خمسة معايير دقة PAN هي Precision, Recall, F–measure, Granularity and Plagdet.و أوضحت النتائج التجريبية أن النموذج المقترح لاسترجاع المرشحين له تأثير إيجابي على الأداء العام للنظام وأن النظام يتفوق على الطرق الحديثة الأخرى. وأوضحت نتائج المقارنة للنظام المقترح أن النظام المقترح إما متفوق أو قابل للمقارنة مع الانظمة الأخرى من حيث مقاييس PAN القياسية الخمسة

1 Background

With Internet explosive growth, the massive volume of information offered with effortless use and access makes the process of taking someone else's work and representing it as other's own work easy to be performed. Due to that, a rapid growing has occurred for the act of plagiarism. Plagiarism is defined as reusing someone else's work (*represented by ideas, or even words*) without citing the source [1]. At the present time, detecting plagiarism is massively necessitated in various areas for ensuring text, materials, and resources originality. Plagiarism detection tool can have crucial role for preventing people aiming to perform intentional plagiarism so that they should offer considerable effort for contributing novel thoughts or even techniques to the academic world centered on their research [2].

Plagiarism detection (PD) is one application of Natural Language Processing (NLP) that is connected with methods from associated fields, such as and soft computing (SC), data mining (DM), and information retrieval (IR). Discovering illegal copying of text patterns from other sources is the focus of PD research [3].

Detecting plagiarism can be performed manually or automatically. The manual technique for identifying plagiarism inside the text is a big challenge. As understanding of text is different from person to person, and when the amount of information increases, a reader is less probably to be able to discover the similarity among textual contents. Therefore automatic plagiarism detection began to gain attention as it can be capable of providing an effective and efficient solution at a lower economic cost than the use of human resources [4].

Generally, automatic plagiarism detection is classified into two standard detection approaches extrinsic plagiarism and intrinsic plagiarism detection approach. Within the case of the first method, a comparison is performed for a suspected document against a collection of sources (*corpus*) [5]. Whereas in the second method, a *suspicious document* is analyzed to discover parts that have not been written through author of this specified document (*author writing style*) devoid of carrying out comparisons with an extrinsic collection of sources [6].

Plagiarism can appear in lots of fields, such as written text (textual), source code in programming languages, design, image, video, and even music portions. In academic, types of plagiarism may be classified into two primary types, *source code* plagiarism and *textual* plagiarism [7].

Source code plagiarism can be arisen in different ways, such as code manipulation, reordering the code structure without modification and language replacing [8]. On the other hand, textual plagiarism can be categorized into two standard ways based on the plagiarist's behavior: literal and intelligent plagiarism. Within the literal plagiarism, plagiarists don't make any effort to hide the plagiarism they committed. They just copy and paste the text from a specific source, with or without citing this original source (*without clear quotation*). While in the second way, various intelligent methods may be used to hide the original work, which may include textual content manipulation or obfuscation. Mainly, obfuscation is performed through, text insertion, text shuffling, text deletion, and so on. Obfuscations range from simple to complex, including, replacement with synonyms, translation, summarization and idea adoption [9]. All the previously mentioned cases of textual plagiarism types are considered as mono-lingual (*plagiarized from text documents involving one language*) except for *text translation plagiarism* also known as cross-lingual plagiarism (*plagiarized from text documents involving more than one language*) [10].

2 Related works

For detecting external plagiarism in textual documents, most works consider three main stages: preprocessing source documents and the suspected document for retrieving a reduced set of candidates that may be sources for plagiarism. Next, a second stage begins that compares in details the suspicious document and each of the candidates generated from the retrieval stage wherein plagiarized sentences are detected. Finally, the consecutive sentences within a given distance are grouped into sections and all the extracted sections pairs are presented in a task called *heuristic post-processing* task. In what follows are some of the works for detecting extrinsic plagiarism in texts.

Alzahrani and Salim [11] introduced a semantic plagiarism detection technique that implemented string similarity based on fuzzy semantic. The scheme proposed in their work was established on: *preprocessing* that involves segmentation, tokenization, stop words exclusion and stemming, next, a list of candidate documents were retrieved by means of $Jaccard$ measurement and shingling algorithm in correspondence to every suspected document. Furthermore, a sentence-wise comparison was performed between the suspicious document and the related candidate documents. In this step, degree of fuzzy similarity was computed which have values range between 0 and 1: 0 for sentences that are wholly dissimilar and 1 for matching sentences. If a fuzzy similarity score exceeding a specific threshold was attached to a pair of sentences, they were marked as similar sentences. Lastly, post-processing was performed in which successive sentences were merged to form plagiarized sections.

In [4], a model for similarity based on fuzzy semantic for detecting obfuscated plagiarism was offered. A comparison of the proposed model was performed against five state of the art methods. The work focused on applying part-of-speech (POS) tags in addition to similarity measures based on WordNet for studying semantic relatedness between words. For assessing the semantic distance between suspicious and source documents of short lengths, fuzzy-based rules were hosted, which implemented as a membership function to a fuzzy set, the semantic relatedness between words. A learning method which combined a permission and a variation threshold was implemented for making a decision about true plagiarism cases for the sake of minimizing number of false negatives and false positives. The model proposed in their work and the baselines were assessed on ground-truth annotated cases taken out from diverse datasets. Extensive experimental verifications were conducted by the authors involving studying the impacts of diverse segmentation approaches and different settings for the parameters. When their approach compared against the baselines, it was shown to be statistically significant using paired t-tests, which revealed the proficiency of the proposed model for detecting cases of plagiarism beyond the verbatim plagiarism. Furthermore, using the variance analysis (ANOVA) statistical test clarified the effectiveness of diverse segmentation approaches applied to the proposed model.

In [12], combining different similarity metrics were investigated for the detection of extrinsic plagiarism and it was centered on clarifying the significance of combining similarity measures over the commonly used single metric usage in detecting plagiarism. Moreover, analyzing the effect of using POS tagging in the plagiarism detection model was performed. Different combinations of the four single metrics, Match coefficient, Dice coefficient, Cosine similarity, and Fuzzy-Semantic measure were used with and without POS tag information. PAN-2014 was used as an evaluation dataset and PAN measures were used as an evaluation metrics for analyzing and comparing results.

In [13], an approach constructed on the linguistic knowledge for detecting plagiarism was proposed by A. Abdi et al.. for calculating similarity between two sentences, they integrated three similarity measures: for pair of sentences, calculating semantic similarity, measuring word-order similarity, and for pair of words, computing semantic similarity. The impact of the three similarity measures was analyzed at their approach and as a result the best combination of them was selected. According to the evaluation carried out using PAN-PC dataset, the proposed method verified that it was easy to follow and required minimal cost for processing text. The experimental results clarified that the performance of the proposed approach was competitive when comparing it with other methods in PAN-PC-10 and PAN-PC-11 datasets.

For the work in this research paper, the contribution is proposing a candidate retrieval model that affects the process of detecting plagiarism positively through taking in consideration how the way of representing documents can affect improving the detection of plagiarism. As a result, a retrieval model based on the commonly used VSM method has been proposed wherein documents have been represented as vectors of average term weights through using $tf - isf$ as weighting scheme and considering them as queries for retrieval instead of the representation as vectors of term $tf - idf$ weights. The rest of this paper is organized as follows: Section 3 introduces the statement of the external plagiarism detection problem together with the preliminary concepts for its main stages. Next, the description of the work proposed in this paper has been introduced in section 4. Performance evaluation and the performance comparison of the proposed system have been introduced in section 5. In addition, analysis and discussion of the proposed work have been introduced in this section. Finally, conclusions and some future works directions have been introduced in section 6.

3 Preliminary concepts

3.1 Problem statement and formulation

For the problem of external textual plagiarism detection, a suspected document represented by d^{sus} and a massive collection of sources represented by $corp^{src}$ are given. For detecting plagiarism in the proposed system, three tasks have been performed in the sequence illustrated in what follows: Firstly, a smaller set of candidates $cand^{src}$ form $corp^{src}$ that are the most similar to d^{sus} and may be the source of the plagiarized contents is retrieved. The *detailed comparison* task comes in the second stage in which d^{sus} is compared in detail against each document d^{csrc} contained in $cand^{src}$, then an extraction of pair of sentences belonging to the documents under comparison d^{sus} and d^{csrc} is performed and the d^{sus} the sentence is considered plagiarized if it's similarity with the sentence of the candidate d^{csrc} is within or higher than a given threshold. This similarity is measured by measuring the *fuzzy semantic based string similarity*. Finally, the consecutive sentences within a given distance are grouped into sections and all the extracted sections pairs are presented in a task called *heuristic post-processing* task. Preliminary concepts together with the implemented and the proposed models and algorithms for the stages of the plagiarism detection are illustrated in what follows:

3.2 Candidate retrieval stage

The preprocessing steps in this stage include: Tokenization, punctuation elimination, lowercasing, removing duplicate tokens, stop-words removal, stemming and removal of duplicate stems. The result from preprocessing is the set of all distinct terms exist at the suspicious document d^{sus} and the collection of source documents $corp^{src}$; $T = \{t_1, t_2, t_{3,\dots,} t_m\}$.

The objective of candidate retrieval stage is to retrieve a reduced set of sources from the corpus $corp^{src}$ that are relevant and satisfy a global similarity to the suspected document d^{sus} and will be determined as candidates to be sources of plagiarism. This preliminary filtering is a significant task for reducing the number of possible pairs of documents before the exhaustive analysis phase. The source documents within an equal or a higher similarity score than a particular threshold are considered to be the candidates for the detailed comparison stage. Formally speaking:

$d^{csrc} = \{d^{src} \in corp^{src} | \, sim(d^{src}, d^{sus}) \geq thr\}$

The Candidates list will be defined then as:

$cand^{src} = \{d_1^{csrc}, d_2^{csrc}, d_3^{csrc}, \dots, d_N^{csrc}\}$

Where:

N: The top N sources d_i^{src} that have attained similarity to $d^{sus} \geq thr$

Implementing the VSM retrieval model $CR_{\varphi 1}$

In this stage, the commonly used information retrieval model VSM has been adopted as a representation model for representing the suspicious document d^{sus} and the collection of sources comprise $corp^{src}$. Elements comprising these vectors have been then weighted according to this model. Next, the $cosine\ similarity$ measure has been used for measuring the similarity between each pair of documents that have been represented as vectors. Finally, the lists involving the pair of documents and their related similarity resulted from this retrieval models are then ranked in a descending order based on $cosine\ similarity$ score and the $top\ N$ sources within or greater than a given threshold thr have been retrieved and considered as candidates for the detailed comparison stage. Value of N has been tuned and tried for several values to investigate the suitable value that the system performs well using it for retrieving candidates.

In this model, the suspicious d^{sus} and all sources d_i^{src} in the corpus $corp^{src}$ are tokenized, duplicate tokens are excluded, stop words are removed, the resulted terms are stemmed and the duplicate stems are removed to generate the set T involving m distinct terms occurred at d^{sus} and all d_i^{src} in $corp^{src}$ wherein $T = \{t_1, t_2, t_{3,\dots,} t_m\}$. Next, d^{sus} all d_i^{src} in $corp^{src}$ are represented as vectors of length m comprising the m distinct terms in T weighted by means of the $tf - idf$ whighting scheme. Algorithm 1 illustrates the steps of implementing the commonly used VSM model:

Algorithm 1: candidate retrieval through implementing $CR_{\varphi 1}$
Input : d^{sus}, $corp^{src} = \{d_1^{src}, d_2^{src}, d_3^{src}, \dots, d_{nsrc}^{src}\}$, thr
Output: $cand^{src} = \{d_1^{csrc}, d_2^{csrc}, d_3^{csrc}, \dots, d_N^{csrc}\}$
Step one: Start
Step two: $T = \{t_1, t_2, t_{3,\dots,} t_m\}$ ← Preprocess d^{sus} & all d_i^{src} in $corp^{src}$
Step three: According to VSM, represent d^{sus} & all d_i^{src} in $corp^{src}$ as m size vectors containing $t_{k,1\le k\le m}$ weight through weighting it by $tf - idf$ scheme
Step four: Compute Cosine similarity between d^{sus} & all d_i^{src} in $corp^{src}$
$for\ i\ =\ 1\ to\ nsrc$
- $simscore_i$ ← compute $CosSim(d^{sus}, d_i^{src})$
- $listSim[i]$ ← Add$(d^{sus}, d_i^{src}, simscore_i)$

$endfor$
Step five: Sort $listSim$ in descending order
Step six: $cand^{src}$ ← Extract $Top\ N$ (d_i^{src}) wherein $simscore_i \ge thr$
Step seven: Stop

3.3 Detailed comparison stage

The detailed comparison has been implemented through measuring the fuzzy similarity between the sentences comprising the pair of documents under comparison. Firstly, the suspicious d^{sus} and the set $cand^{src}$ of the candidate documents d^{csrc_i} retrieved from the *candidate retrieval* stage are preprocessed. Next, d^{sus} is compared in detail with each d^{csrc_i} in $cand^{src}$ through measuring fuzzy semantic-based string similarity. The specified pair of documents are compared using sentence level comparison. A sentence is considered as a plagiarized sentence if it recorded a similarity score equal to or larger than a given threshold value.

For a detailed comparison, firstly, the preprocessing steps necessitated for measuring fuzzy similarity between the pair of documents under comparison are performed. For preprocessing the pair of documents under comparison through fuzzy similarity, the specified pair of documents, d^{sus} and d_i^{csrc} are preprocessed wherein the segmentation process is applied to the documents for segmenting them into individual sentences. Then, sentences that comprise three words or less are discarded and the duplicate sentences are excluded. After that, tokenization, punctuation elimination, lowercasing, removal of duplicate tokens, stop words removal, lemmatization and exclusion of duplicate lemmas are applied in sequence. Lemmatization has been used in this proposed work instead of stemming process for the reason that lemmatization yields dictionary base forms that are suitable for comparing semantics. As a result of preprocessing, the specified pair of documents under comparison will be stated formally as: $d^{sus} = \{s_k^{sus} | 1 \le k \le n_{sus}\}$ and $d_i^{csrc} = \left\{s_j^{csrc_i} \middle| 1 \le j \le n_{csrc_i}\right\}$ wherein n_{sus} denotes the total number of d^{sus} sentences whereas, n_{csrc_i} denotes the total number of sentences comprising d_i^{csrc}.

Afterwards, the fuzzy semantic similarity is calculated between sentence s_k^{sus} in d^{sus} with sentence $s_j^{csrc_i}$ exist at the candidate d_i^{csrc}.

A fuzzy similarity between two words $word_{ii}$ and $word_{jj}$ can be computed as in Eq. (2) [11].

$$Fs_{ii,jj} = \begin{cases} 1 & \text{if} \quad word_{ii} = word_{jj} \\ 0.5 & \text{if} \quad word_{ii} = \text{is in the } synset \text{ of } (word_{jj}) \\ 0 & otherwise \end{cases} \tag{2}$$

However, to obtain the degree of similarity between two sentences $(s_k^{sus}, s_j^{csrc_i})$, a term-to-sentence correlation factor for each term w_1 in s_1 and the sentence s_2, is computed as in equation (3).

$$M_{s_k^{sus}, s_j^{csrc_i}} = 1 - \Pi\, w_k \in s_j^{csrc_i} (1 - Fs_{ii,jj}) \tag{3}$$

Where w_k are words in $s_j^{csrc_i}$ and $Fs_{ii,jj}$ is a fuzzy similarity between $word_{ii}$ and $word_{jj}$.
According to the $M - value$ of every word in a sentence s_k^{sus}, which is computed against sentence $s_j^{csrc_i}$, the similarity between s_k^{sus} and $s_j^{csrc_i}$ can be defined as in equation (4)

$$Fs\,(s_k^{sus}, s_j^{csrc_i}) = \frac{(M_{1,s_k^{sus}} + M_{2,s_k^{sus}} + \cdots + M_{n,s_k^{sus}})}{n} \tag{4}$$

Where n is the total number of words in s_k^{sus}.

However, if the two sentences s_k^{sus} and $s_j^{csrc_i}$ sentences have an unequal number of word $Sim(s_k^{sus}, s_j^{csrc_i}) \neq Sim(s_j^{csrc_i}, s_k^{sus})$, in this case, the minimum similarity score must be computed as in equation (5).

$$EQ\left(s_k^{sus}, s_j^{csrc_i}\right) = \begin{cases} 1 & if \;\; Min\left(Fs\left(s_k^{sus}, s_j^{csrc_i}\right), Fs\left(s_j^{csrc_i}, s_k^{sus}\right)\right) \geq thr_{fuzzy} \\ 0 & otherwise \end{cases} \tag{5}$$

Where thr_{fuzzy}, is a permission threshold value, which is the minimal similarity between pair of sentences s_k^{sus} *and* $s_j^{csrc_i}$.

Finally, the suspicious sentence s_k^{sus} with similarity score against $s_j^{csrc_i}$ that is within or exceeds the threshold values th_{fuzzy} for fuzzy semantic based similarity is considered as a plagiarized sentence. As a result, the sentence pair $(s_k^{sus}, s_j^{csrc_i})$ *where* $s_j^{sus} \in d^{sus}$ *and* $s_j^{csrc_i} \in d_i^{csrc}$ are marked as plagiarized and included in an output list and sorted in descending order according to their similarity score attached with their corresponding suspicious and candidate documents.

3.4 Post processing stage

Regarding the list of sentences pairs $(s_k^{sus}, s_j^{csrc_i})$ together with the documents comprising them where $s_k^{sus} \in d^{sus}$ and $s_j^{csrc_i} \in d_i^{csrc}$ resulted from the detailed comparison stage, the successive sentences that are within a given distance are merged to constitute the plagiarized passages. A distance of 100 characters is considered for the proposed work. Finally, the plagiarized passage p^{sus} and the source passage $p_t^{csrc_i}$ from the document that has been verified to be the source of plagiarism are presented to the user as the pair of passages $(p^{sus}, p_t^{csrc_i})$ together with the documents involving them, where $p^{sus} \in d^{sus}$ and $p_t^{csrc_i} \in d_i^{csrc}$.

4 The proposed method

A retrieval model named as $CR_{\varphi 2}$ has been proposed in this paper. In this model, d^{sus} and all d_i^{src} in $corp^{src}$ are segmented into individual sentences and the duplicate sentences and the sentences with three words or less are removed. After that, the documents are preprocessed following all the steps applied to the first model $CR_{\varphi 1}$. Next, for each pair involving the suspicious document d^{sus} and each one of the sources d_i^{src}, given $T^{pair} = \{t_1, t_2, t_{3,\ldots,} t_{srcsus}\}$ where T^{pair} represents the set of all distinct terms exist in the specified pair d^{sus} and d_i^{src} and $srcsus$ denotes the total number of distinct terms exist in the considered pair of documents, the center vectors

$C^{sus} = \{c_1, c_2, c_{3,\dots,} c_{srcsus}\}$ *and* $C_i^{src} = \{c_1, c_2, c_{3,\dots,} c_{srcsus}\}$ for d^{sus} and d_i^{src} respectively are calculated. In this model, the documents are represented as vectors of average term weight and used $tf - isf$ as a weighting scheme for terms weighting. The k^{th} coordinate, c_k of the center, vector C is calculated as in Eq. (1) illustrated below:

$$c_k = \frac{1}{n}\sum_{i=1}^{n} w_{ik} \qquad k = 1, 2, 3, \dots, srcsus \tag{1}$$

Wherein for the suspicious document, n denotes the total sentences number within it whereas for the source document, it denotes the total sentences number within this specified source. Finally, the $cosine\ similarity$ between d^{sus} and d_i^{src} represented as center vectors C^{sus} and C_i^{src}, respectively, are calculated. Steps for the proposed model $CR_{\varphi 2}$ are illustrated in Algorithm 2. Also, the steps for implementing detailed comparison using fuzzy semantic based string similarity is clarified in Algorithm 3 in what follows:

Algorithm 2: Proposed candidate retrieval model $CR_{\varphi 2}$
Input : d^{sus}, $corp^{src} = \{d_1^{src}, d_2^{src}, d_3^{src}, \dots, d_{nsrc}^{src}\}$, thr
Output: $cand^{src} = \{d_1^{csrc}, d_2^{csrc}, d_3^{csrc}, \dots, d_N^{csrc}\}$
Step one: Start
Step two: For all sources d_i^{src}, perform the following processes:
$for\ i\ =\ 1\ to\ nsrc$
- $T = \{t_1, t_2, t_{3,\dots,} t_{srcsus}\} \leftarrow$ Preprocess d^{sus} & d_i^{src}
- According to VSM, represent d^{sus} & d_i^{src} as vectors containing t_k weight through weighting it by $tf - isf$ scheme
- $C^{sus} \leftarrow$ compute c_k of center vector of d^{sus} using $Eq.(1)$
- $C_i^{src} \leftarrow$ compute c_k of center vector of d_i^{src} using $Eq.(1)$
- $simscore_i \leftarrow$ compute $CosSim(C^{sus}, C_i^{src})$
- $listSim[i] \leftarrow$ Add$(d^{sus}, d_i^{src}, simscore_i)$

$endfor$
Step three: Sort $listSim$ in descending order
Step four: $cand^{src} \leftarrow$ Extract $Top\ N\ d_i^{src}$ wherein $simscore_i \geq thr$
Step five: Stop

Algorithm 3: Detailed comparison using fuzzy semantic based string similarity

Input:
$\boldsymbol{d^{sus}}$: Suspicious document:
$\boldsymbol{cand^{src}} = \{\boldsymbol{d_i^{csrc}} |\ \boldsymbol{1 \le i \le N}\}$: Candidate list
$\boldsymbol{WordNet}$
Output:
$\boldsymbol{PlagSentList}$: List containing plagiarized sentences together with their source sentences, source documents and their similarity scores
start
$for\ i = 1\ to\ N$
Preprocess the pair of documents under comparison: d^{sus} and d_i^{csrc}
$for\ k = 1\ to\ n_{sus}$
$for\ j = 1\ to\ n_{csrci}$
Compute Fuzzy similarity between s_k^{sus} & $s_j^{csrc_i}$
$F_s = 0$; $F_{s1} = 0$; $F_{s2} = 0$;
If (length of (s_k^{sus}) == length of ($s_j^{csrc_i}$))
Begin
$Foreach\ word\ in\ s_k^{sus}$
Extract $Synset\ (word)$ from $WordNet$
If $s_j^{csrc_i}$ Contains $(word)$ then $F_s = F_s + 1$
$Elseif\ s_j^{csrc_i}$ Contains $(synset\ of\ (word))$ then $F_s = F_s + 0.5$
$Else\ F_s = F_s + 0$
$EndForeach$
$FuzzySimilarityScore = \frac{F_s}{NumberOfWords\ in\ s_j^{csrc_i}}$
$Endif$
Else
Begin
$Foreach\ word\ in\ s_k^{sus}$
Extract $Synset\ (word)$ from $WordNet$

If $s_j^{csrc_i}$ Contains $(word)$ then $F_{s1} = F_{s1} + 1$
$Elseif\ s_j^{csrc_i}$ Contains $(synset\ of\ (word))$ then $F_{s1} = F_{s1} + 0.5$
$Else\ F_{s1} = F_{s1} + 0$
$EndForeach$
$F_{s1} = \frac{F_{s1}}{NumberOfWords\ in\ s_j^{csrc_i}}$
$Foreach$ word in $s_j^{csrc_i}$
Extract $Synset\ (word)$ from $WordNet$
If s_k^{sus} Contains $(word)$ then $F_{s2} = F_{s2} + 1$
$Elseif$ if s_k^{sus} Contains $(synset\ of\ (word))$ then $F_{s2} = F_{s2} + 0.5$
$Else\ F_{s2} = F_{s2} + 0$
$EndForeach$
$F_{s2} = \frac{F_{s2}}{NumberOfWords\ in\ s_k^{sus}}$
$FuzzySimilarityScore = GetMinimumValue\ (F_{s1}, F_{s2})$
$EndElse$
Add sentences pair that recorded a similarity score within or exceeds the given threshold value to the list of plagiarized sentences
if $FuzzySimilarityScore \geq thr_{fuzzy}$
$PlagSentList = (s_k^{sus}, s_j^{csrc_i}, d^{sus}, d_i^{csrc}, SimilarityScore_{kji})$
$endif$
$EndforEachSrcSent$
$EndforEachSusSent$
$EndforEachCand$
Stop

5 Experimental results

5.1 Requirements

For excluding stop words, the English stop words list (*http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11smart−stop−list/english.stop*) has been used. Also, lemmatization has been used instead of stemming for generating lemmas. For semantic-based analysis, WorldNet v3.0 using MySQL has been used for querying the $Synset$ table and extracting word synonyms.

5.2 Evaluation metrics

A plagiarism detection system is typically assessed through the use of the standard evaluation metrics which includes $recall$, $precision$, and $F-measure$. Furthermore, in the context of the PAN competitions, $plagdet$ and $granularity$ metrics have been proposed. $Granularity$ measures the method accuracy at discovering the right segmentation for cases of plagiarism, whereas, $Plagdet$ characterizes the total score of combining $granularity$ and $F-measure$ [14].

For further explanation, let d_q be a plagiarized document; d_q defines a characters sequence each of which is considered as plagiarized or non-plagiarized. A plagiarized section s forms an adjacent arrangement of plagiarized characters in d_q. The set of all plagiarized sections in d_q is denoted by S. Also, the set of all sections $r \subseteq d_q$ found through a plagiarism detection algorithm is denoted through R. If the characters in d_q are considered as basic retrieval units, precision and recall for a given (d_q, S, R). Computing $macro-average\ precision$ and $recall$ is illustrated in $Eqs.$ (6) and (7) respectively:

$$Precision\ (S,R) = \frac{1}{|R|}\sum_{r\in R}\frac{|\cup_{s\in S}(s \sqcap r)|}{|r|} \tag{6}$$

$$Recall\ (S,R) = \frac{1}{|S|}\sum_{r\in R}\frac{|\cup_{r\in R}(s \sqcap r)|}{|s|} \tag{7}$$

Where ⊓ computes the positionally overlapping characters.

The two measures are sometimes used together in the $F-Score$ ($f-measure$) to provide a single measurement for a system which is calculated as in Eq. (8).

$$F-measure = \frac{2*\text{Precision}*\text{Recall}}{\text{Precision}+\text{Recall}} \tag{8}$$

In addition to $precision$ and $recall$, another evaluation metric, $granularity$ is used for evaluating plagiarism detection system which is defined as the ratio of the number of recognized plagiarized source sections to a given plagiarized source sections as illustrated in Eq. (9) :

$$Granularity(S,R) = \frac{1}{|S_R|} \sum_{s \in S_R} |R_s| \quad (9)$$

Where $S_R \subseteq S$ cases are recognized through detecting s in R and $R_S \subseteq R$ are the detections of a given s. The domain of granularity (S,R) is $[1,|R|]$. The minimum and ideal granularity value is 1 and $|R|$ indicates the worst case.

The measures are joined into a single score $Plagdet$ for making a unique ranking among methods of detection as in Eq. (10).

$$Plagdet = \frac{F-\text{measure}}{log_2(1+\text{granularity}\ (S,R))} \quad (10)$$

For the candidate retrieval stage, the evaluation metrics which are used in the IR field, $Recall, Precision$ and $F-measure$ were used of the proposed method.

A $recall$ is described as the number of relevant documents retrieved through an algorithm divided by the total number of existing relevant documents, while $precision$ is defined as the number of relevant documents retrieved through a search divided by the total number of documents retrieved via that algorithm. $F-measure$ which combines the two metrics precision and recall in the harmonic mean as in Eq. (8), see $Eqs.$ (11) and (12) respectively [15]:

$$\text{Recall} = \frac{Retrieved\ relevant}{Total\ relevant} \quad (11)$$

$$\text{Precision} = \frac{Retrieved\ relevant}{Total\ retrieved} \quad (12)$$

5.3 Parameters setting and Performance evaluation

In the present section, the main focus is on optimizing parameters for the candidate retrieval proposed model. For being more specific, its parameters have been tried to be optimized through running the proposed model on the training dataset and then, selecting the values that the proposed model performs well using them for evaluating the proposed model using the testing dataset. Next, the proposed model has been evaluated using $precision, recall\ and\ f-measure$ evaluation metrics. After that, a comparison has been performed between the results of the proposed model $CR_{\varphi 2}$ against the existing VSM model $CR_{\varphi 1}$. In order to discover the suitable number of candidates ($top\ N$) to be regarded at the next stage, experiments have been carried out with different values for N ($N = 8,9,10,11,12,15$) for the existing and the proposed model. On considering the results, it is shown that the best performance has been attained with ($N = 8$).

Table 1-Performance evaluation of implementing the existing $CR_{\varphi 1}$ retrieval model and the proposed candidate retrieval model $CR_{\varphi 2}$ using various values for N in terms of $Precision, Recall\ and\ F-measure$ evaluation metrics.

Retrieval model	$Top\ N$	$Precision$	$Recall$	$F-measure$
$CR_{\varphi 1}$	**8**	**0.458**	**0.678**	**0.546**
	9	0.407	0.678	0.508
	10	0.367	0.678	0.475
	11	0.333	0.678	0.446
	12	0.306	0.678	0.420
	15	0.267	0.800	0.397
$CR_{\varphi 2}$	**8**	**0.500**	**0.744**	**0.597**
	9	0.444	0.744	0.556
	10	0.433	0.811	0.564
	11	0.394	0.811	0.529
	12	0.361	0.811	0.499
	15	0.311	0.867	0.457

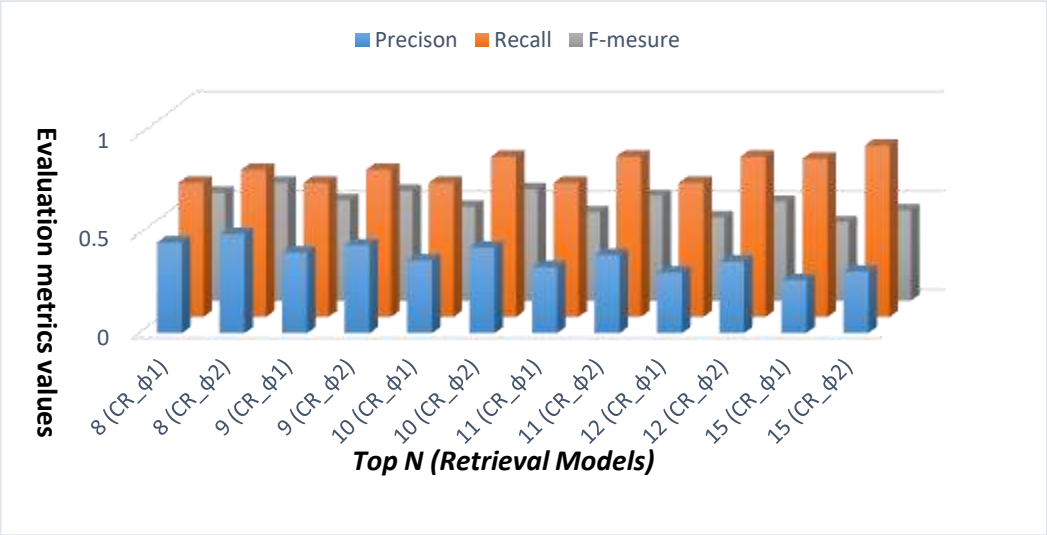

Figure 1- Performance evaluation of implementing the existing $CR_{\varphi 1}$ retrieval model and the proposed candidate retrieval model $CR_{\varphi 2}$

In Table-1, 2,3 and Figure 1,2,3 the performance evaluation of the proposed candidate retrieval model has been introduced. Firstly, for the implementation of the model named $CR_{\varphi 1}$ that bases on utilizing VSM for representing documents under comparison as vectors whose elements are weights of their terms through using $tf - idf$ weighting scheme, the performance evaluation has been achieved. Secondly, performance evaluation of the proposed model $CR_{\varphi 2}$ has been achieved which considers elements constituting the vectors as average term weights instead of term weights wherein $tf - isf$ weighting scheme has been used. It is observed that $CR_{\varphi 2}$ performs better than $CR_{\varphi 1}$.

Table 2-Performance comparison of implementing the detailed comparison using fuzzy semantic based string similarity and applying the proposed retrieval model $CR_{\varphi 2}$ and the work of $sys_{[11]}$ and $sys_{[13]}$ in terms of $Precision$, $Recall$, $F-measure$, granularity and $plagdet$ evaluation metrics.

Retrieval Model	Precision	Recall	F-Measure	Granularity	$Plagdet$
$CR_{\varphi 2}$	**0.9061**	**0.8098**	**0.85366**	1.47037	0.65429
$sys_{[11]}$	0.575	0.154	0.242	3.591	0.1109
$sys_{[13]}$	0.802	0.685	0.739	**1.01**	**0.733**

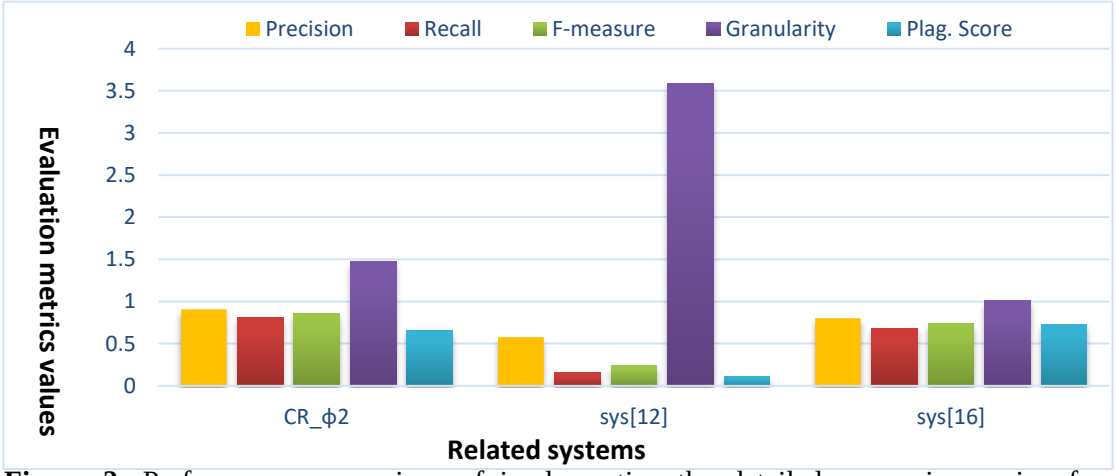

Figure 2: Performance comparison of implementing the detailed comparison using fuzzy semantic based string similarity and applying the proposed retrieval model $CR_{\varphi 2}$ and the other works

Table 3-Relative improvement percentage of the proposed system against other systems

Related systems	Precision	Recall	F-measure	$Plagdet$
$sys_{[11]}$	+57.583	+425.844	+252.752	+489.982
$sys_{[13]}$	+12.980	+18.219	+15.516	-10.738

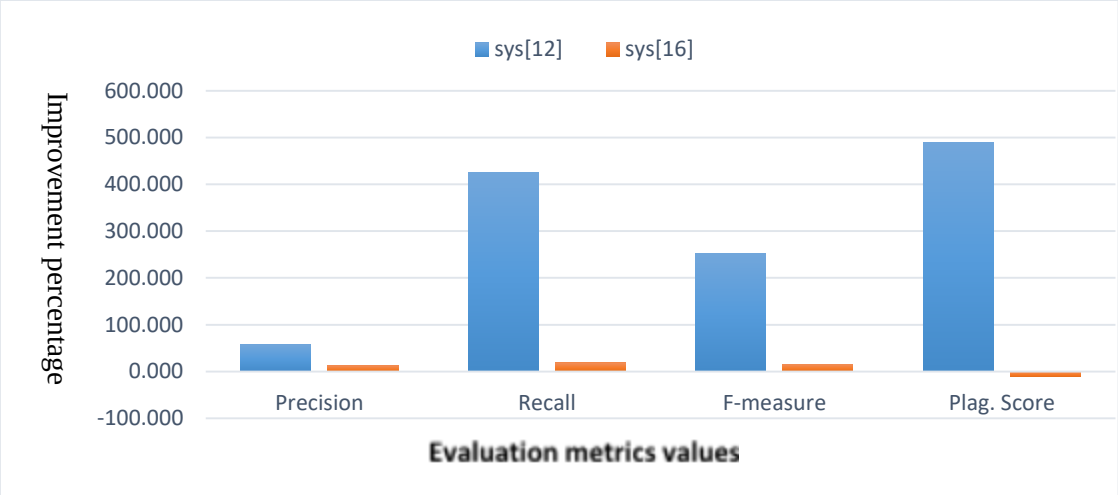

Figure 3-Relative improvement percentage of the proposed system against other systems

5.4 Results analysis and discussion

PAN10 has been used as an evaluation dataset for evaluating the proposed model. Two types of plagiarism are included in the corpus: extrinsic and intrinsic plagiarism. As the problem statement in this work is restricted to detect extrinsic plagiarism and to work on English documents, our experiments have been performed on the portion dedicated for extrinsic detection which involves 70% of the documents in the collection and on documents in English language only. These documents have been randomly separated into training and testing dataset. The training data have been used for parameters tuning whereas evaluating the performance of the proposed system and comparing it against the existing methods have been performed using testing dataset. For evaluating the performance of each of the proposed models, five folds with an equal number of plagiarism case types (high obfuscation, low obfuscation and none obfuscation) have been evaluated and their average has been considered. The models proposed for solving candidate retrieval problem have been evaluated using $Precision,\ Recall$ and $f-measure$ as an evaluation metrics. The overall performance of the proposed system has been assessed through the use of the five PAN standard measures $Precision$, $Recall$, $f-measure$, $Granularity$ and $Plagdet$. Experimental results clarified that the proposed model has detected about 80% of the plagiarism cases and about 90% of the detections were correct. In the proposed model, the reasons for recording low recall in the work [11] belongs to the use of stems instead of lemmas which has been overcome in the proposed system. The other reason that has been taken into consideration is focusing on improving the stage of candidate retrieval in order to improve the recall of detection stage.

6 Conclusions and future works

Based on the commonly used VSM retrieval model, a model for retrieving candidates and necessitated for the detailed comparison stage has been proposed. This proposed retrieval model that represents documents as vectors constituting average weights of their terms instead of term weights and then measuring the similarity between the centers of the documents has improved the performance of retrieval problem and the overall performance of the plagiarism detection system. Experimental results demonstrated that the proposed model has the ability to capture the relevant document and passing them as candidates for the detailed comparison. They clarified that the proposed method has detected about 80% of the plagiarism cases and about 90% of the detections were correct. The proposed model has the ability to detect literal plagiarism in addition to cases containing paraphrasing. Performance comparison has been illustrated that the proposed system either outperforms or comparable with other baseline systems. As future work, we aim to improve the performance of the

system to detect intelligent plagiarism cases by means of discovering the different preprocessing methods constructed on NLP techniques. Also, in order to get higher recall scores, we aim at improving the proposed model for detecting and retrieving more sources for detailed comparison stage.

References

1. Abdi, A., Shamsuddin, S., Idris, N., Rasim, M. and Ramiz M**. 2017.** *A linguistic treatment for automatic external plagiarism detection.* Knowledge-Based Systems, p.135-146.
2. Lukashenko, R., Graudina, V. and Grundspenkis, J. **2007.** Computer-based plagiarism detection methods and tools: an overview. in Proceedings of the 2007 international conference on Computer systems and technologies. 2007. ACM.
3. Sarkar, A., Marjit, U. and Biswas, U. **2014.** A conceptual model to develop an advanced plagiarism checking tool based on semantic matching. in 2014 2nd International Conference on Business and Information Management (ICBIM). 2014. IEEE.
4. Alzahrani, S., Naomie, S. and Vasile, p**. 2015.** Uncovering highly obfuscated plagiarism cases using fuzzy semantic-based similarity model. Journal of King Saud University-Computer Information Sciences. **27**(3): 248-268.
5. Oberreuter, G. and VeláSquez, J.D.J.S.W.A. **2013.** Text mining applied to plagiarism detection: The use of words for detecting deviations in the writing style. 2013. **40**(9): 3756-3763.
6. Wang S., Haoliang Q., Leilei K. and Cuixia N. **2013.** Combination of VSM and Jaccard coefficient for external plagiarism detection. in International Conference on Machine Learning and Cybernetics. 2013. IEEE.
7. Rao, S., Gupta, P., Singhal, K. and Majumder, P**. 2011.** External & Intrinsic Plagiarism Detection: VSM & Discourse Markers based Approach Notebook for PAN at CLEF 2011. 2011.
8. Alzahrani, S., Naomie, S. and Ajith, A**. 2012.** Understanding plagiarism linguistic patterns, textual features, and detection methods. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews. **42**(2): 133-149.
9. Prechelt, L., Malpohl, G. and Philippsen, M.J.J.U**. 2002.** Finding plagiarisms among a set of programs with JPlag. *Journal of Universal Computer Science* **8**(11): 1016–1038 (2002).
10. Roig, M**. 2006.** *Avoiding plagiarism, self-plagiarism, and other questionable writing practices: A guide to ethical writing*. NewYork: St. Johns Univ.Press.
11. Alzahrani, S. and Salim, N. **2010.** Fuzzy Semantic-Based String Similarity for Extrinsic Plagiarism Detection Lab Report for PAN at CLEF *2010.* 2010.
12. Vani, K. and Gupta, D. **2015.** Investigating the impact of combined similarity metrics and POS tagging in extrinsic text plagiarism detection system. in 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2015. IEEE.
13. Abdi A., Shamsuddin S., Idris N., Rasim M., and Ramiz M**.** PDLK. **2015.** Plagiarism detection using linguistic knowledge. *Expert Syst. Appl*., 2015, **42**: 8936–894.
14. Potthast, M. **2011.** Cross-language plagiarism detection. 2011. **45**(1): 45-62.
15. Tie-Yan, L. **2009.** "Learning to Rank for Information Retrieval", Foundations and Trends in Information" *Retrieval*, **3**(3): 225-331.