Expert Systems with Applications 40 (2013) 1804–1813

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Determining and characterizing the reused text for plagiarism detection ☆

Fernando Sánchez-Vega [a,1], Esaú Villatoro-Tello [b,*], Manuel Montes-y-Gómez [a,*], Luis Villaseñor-Pineda [a], Paolo Rosso [c]

[a] Lab. de Tecnologías del Lenguaje, Coordinación de Ciencias Computacionales, Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Mexico.
[b] Information Technologies Department, Universidad Autónoma Metropolitana (UAM), Mexico
[c] Natural Language Engineering Lab., ELiRF, Universitat Politècnica de València, Spain

ARTICLE INFO

Keywords:
Plagiarism detection
Text reuse
Machine learning
Supervised classification

ABSTRACT

An important task in plagiarism detection is determining and measuring similar text portions between a given pair of documents. One of the main difficulties of this task resides on the fact that reused text is commonly modified with the aim of covering or camouflaging the plagiarism. Another difficulty is that not all similar text fragments are examples of plagiarism, since thematic coincidences also tend to produce portions of similar text. In order to tackle these problems, we propose a novel method for detecting likely portions of reused text. This method is able to detect common actions performed by plagiarists such as word deletion, insertion and transposition, allowing to obtain plausible portions of reused text. We also propose representing the identified reused text by means of a set of features that denote its degree of plagiarism, relevance and fragmentation. This new representation aims to facilitate the recognition of plagiarism by considering diverse characteristics of the reused text during the classification phase. Experimental results employing a supervised classification strategy showed that the proposed method is able to outperform traditionally used approaches.

1. Introduction

Plagiarism is known as intellectual theft: it consists in using words (ideas) of others and presenting them as your own. Nowadays, due to current technologies for creating and disseminating electronic information, it is very simple to compose a new document by copying sections from different sources extracted from the Web. This situation has caused the growing of the plagiarism phenomenon, and, at the same time, it has motivated the development of tools for its automatic detection.

Very recently, major publishers, namely Elsevier and Springer have showed their interest and concern to fight plagiarism (Butler, 2010). Hence, by using a software called CrossCheck, they scan submitted papers with the aim of finding verbatim or almost identical chunks of text that already appear in previously published papers. Several tests using the CrossCheck software over different journals showed that from 6% to 23% of the submitted articles had to be rejected because they contain a considerable degree of plagiarism. Although CrossCheck is able to uncover plagiarists, the software is susceptible to find false positives, since it determines the similarity between documents by considering only a percentage of single words overlap.

In this paper we focus on the problem of discriminating plagiarized from free-plagiarized suspicious documents by determining the reused text sections from an original document.[2] We assume that plagiarism is done by reusing some portions of text that can not be considered as common knowledge of the domain. In particular, we consider the task of finding similarities between a suspicious document and a given original document that are more than just a coincidence and more likely to be result of copying (Clough, 2003). This is a very complex task since reused text is commonly modified with the aim of covering or camouflaging the plagiarism. To date, most approaches have only partially addressed this issue by measuring lexical and structural similarity of documents by means of different kinds of features such as single words (Clough, Gaizauskas, Piao, & Wilks, 2002; Zechner, Muhr, Kern, & Granitzer, 2009), fixed length substrings (*i.e.*, word *n*-grams) (Barrón-Cedeño & Rosso, 2009; Clough et al., 2002), variable length substrings (Basile, Benedetto, Caglioti, Cristadoro, & Degli Esposti, 2009; Clough et al., 2002),

☆ This work was done under partial support of CONACyT project Grants: 134186, and Scholarships: 258345/224483. This work is the result of the collaboration in the framework of the WIQEI IRSES project (Grant No. 269180) within the FP 7 Marie Curie. The work of the last author was in the framework of the VLC/CAMPUS Microcluster on Multimodal Interaction in Intelligent Systems.

* Corresponding authors.
E-mail addresses: fer.callotl@inaoep.mx (F. Sánchez-Vega), evillatoro@correo.cua.uam.mx (E. Villatoro-Tello), mmontesg@inaoep.mx (M. Montes-y-Gómez), villasen@inaoep.mx (L. Villaseñor-Pineda), prosso@dsic.upv.es (P. Rosso).
[1] Principal corresponding author.

[2] Plagiarism is a case of text reuse where no explicitly acknowledging of the original author and source are given.

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.09.021

dependency relations or a combination of them (Chong, Specia, & Mitkov, 2010). The main drawback of these approaches is that they carry out the classification considering only information about the degree of overlap between the suspicious and source documents. Therefore, these strategies are affected by the thematic correspondence of the documents, which implies the existence of common domain-specific word sequences, and, as consequence causes an overestimation of their overlap (Clough, 2003).

In order to tackle the above problem we propose a novel approach for finding the portions of possible reused text. Our method, called the *Rewriting Index*, assigns a weight to each word contained in the suspicious document that describes its degree of membership to a possible portion of plagiarized text. This way, the proposed method is able to discover text that has suffered from some modifications such as word elimination, insertion, and transposition, allowing to perform a partial matching between documents. Additionally, we also consider more information during the classification process of the documents. Our idea is to characterize the portions of possible reused text by their relevance and fragmentation. In particular, we consider a set of features that denote the frequency of occurrence of portions of reused text as well as their length distribution. Our hypothesis is that the larger and the less frequent the portions of reused text, the greater the evidence of plagiarism. In other words, we consider that frequent portions of reused text tend to correspond to domain specific terminology, and that small portions of possible reused text may be co-incidental, and therefore, they are not a clear signal of plagiarism.

The experimental evaluation of the proposed approach was carried out on a subset of the *METER* corpus (Gaizauskas et al., 2001) and on the *Plagiarised Short Answers* corpus (Clough & Stevenson, 2010). In particular, we model the document plagiarism detection as a classification problem. Our goal was to show that using the portions of reused text obtained with the *Rewriting Index* method, and characterizing them by the proposed set of features, it is possible to achieve a greater discrimination performance between plagiarized and non-plagiarized documents than only considering their general degree of overlap.

The rest of the paper is organized as follows. Section 2 presents some recent work on plagiarism detection. Section 3 describes the proposed algorithm for finding portions of possible reused text as well as the formal definition of the proposed features. Section 4 presents the experimental configuration as well as the results achieved in the two test collections. Finally, Section 5 depicts our conclusions and formulates some directions for future work.

2. Related work

One of the main tasks in plagiarism detection consists in determining if the similarities between a suspicious and a source (original) document are more than just coincidence and more likely to be result of copying (Clough, 2003). Broadly speaking, this task includes two main phases: the searching of plagiarism evidence, and the classification of plagiarized documents based on the accumulated evidence.

The purpose of the first phase is to find similar or reused text portions between the given two documents. Some works have searched for these similarities at the syntactic level by identifying common POS sequences (Chong et al., 2010; Hartrumpf, Brúck, & Eichhorn, 2010). On the other extreme, some works have searched for similarities at the lexical level, using common single words as the main evidence of plagiarism (Hoad & Zobel, 2003; Shivakumar & García-Molina, 1995; Zechner et al., 2009). Finally, in between these two approaches, there are works that consider word sequences. Some of them search for common fixed-length sequences known as word *n*-grams (Barrón-Cedeño, Eiselt, & Rosso, 2009; Barrón-Cedeño & Rosso, 2009; Chien-Ying, Jen-Yuan, & Hao-Ren, 2010; Grozea & Popescu, 2010, 2011; Oberreuter, L'Huillier, Ríos, & Velásquez, 2011; Rao, Gupta, Singhal, & Majumder, 2011; Seo & Croft, 2008), whereas others have used variable length sequences in order to preserve the integrity of the evidence (Basile et al., 2009; Chien-Ying et al., 2010; Clough et al., 2002; Nawab, Stevenson, & Clough, 2011).

In the second phase the collected evidence is transformed into a measure or set of measures that indicate the level of copy in the suspicious document. Particularly, most current methods use a representation based on the proportion of positive evidence in relation to the size of the suspicious document (Grozea & Popescu, 2011; HaCohen-Kerner et al., 2010; Oberreuter et al., 2011; Rao et al., 2011; Seo & Croft, 2008) or to the size of both documents (Barrón-Cedeño, Basile, Esposti, & Rosso, 2010; Chien-Ying et al., 2010; Stein & Eissen, 2006). This representation is used in the documents classification process; the common approach consists of applying a manually-defined threshold function on the computed measure (Basile et al., 2009; Barrón-Cedeño & Rosso, 2009; Rao et al., 2011; Suárez, González, & Villena-Román, 2011; Si, Leong, & Lau, 1997). On the contrary, when the plagiarism evidence is expressed by a set of measures, most methods apply machine learning techniques to automatically define the threshold function (Clough et al., 2002, 2010; Engles, Lakshmanan, & Craig, 2007; Hartrumpf et al., 2010).

In this paper we propose some ideas to enhance both phases of the plagiarism detection process. First, we propose a new method to find the portions of possible reused text. This method uses a fuzzy string matching automata that is able to detect common actions of plagiarism such as word deletion, insertion and transposition, and, therefore, that allows to collect evidence with a high degree of rewriting, which current methods tend to ignore. Second, we propose a new representation of the plagiarism evidence that helps to describe more appropriately its relevance and diversity and, consequently, allows taking further advantage of the capabilities of machine learning techniques to handle representations with multiple features.

3. Proposed method

As stated in previous sections, common word sequences between the suspicious and source documents are considered the primary evidence of plagiarism. Nevertheless, using their presence as unique indicator of plagiarism could be unreliable, since thematic coincidences also tend to produce sequences of common text (*i.e.*, false positives). In addition, even a minor modification to obfuscate the plagiarism will avoid the identification of the corresponding sequences, generating false negatives.

In order to handle the above problems, we propose a novel strategy for detecting plagiarised text called the *Rewriting Index* method. This method is able to identify portions of reused text even if they have suffered from some modifications. Additionally, we aim to facilitate the recognition of plagiarism by considering diverse characteristics of the portions of reused text during the classification phase.

In the following section we give a brief description of the Turing machine formalism, which will allow us to better describe, in Section 3.2, our proposed algorithm for identifying and extracting the possible reused text between the suspicious (D^S) and the original document (D^O). Then, in Section 3.3, we introduce the proposed set of features used to characterize the extracted portions of reused text.

3.1. Turing machine formalism

In order to explain the proposed method we are going to employ the Turing Machine (TM) notation. Formally a TM is defined as a 7-tuple with the form:

$$M = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle \quad (1)$$

where:

- Q is a finite, non-empty set of states.
- Σ is the set of input symbols.
- Γ is a finite, non-empty set of the tape alphabet (symbols).
- δ is the transition function which is defined as: $\delta(q_i,X) = (q_j,Y,S)$; where q_i represents the actual state and X is the symbol that the head of the TM is reading, q_j is the next state, Y is the symbol that is written in the cell pointed by the head of the TM, and S indicates the direction of the head shift, which could be either $\leftarrow$ (left shift), $\rightarrow$ (right shift) or N (no shift).
- q_0 is the initial state.
- B is the blank symbol.
- F is the set of final or accepting states.

Accordingly, we will employ the string $X_1X_2 \ldots X_{i-1}qX_iX_{i+1} \ldots X_n$ to refer at the configuration where:

- q is the actual state of the TM.
- X_i, the ith symbol from the left, is the symbol pointed by the head of the tape.
- $X_1X_2 \ldots X_n$ is the portion of the tape that is between the most left and most right blank symbols (*i.e.*, B)

Our TM will be capable of reading a null entry (*i.e.*, ε). Hence, a transition like $\delta(q_i,\varepsilon) = (q_j,Y,S)$ means that the TM will go from the state q_i to state q_j by reading ε, indicating to the head of the TM to write Y, and shifting into the S direction.[3]

Furthermore, our TM will handle a stack; *i.e.*, it is a *pushdown* TM. For our purposes, the main goal of the stack is to function as a counter, hence the alphabet of the stack corresponds to the set of the natural numbers $\mathbb{N}$.

Consequently, the transition function for our *pushdown* TM is defined as: $\delta(q_i,X,p) = (q_j,Y,p',S)$; where q_i is the actual state, X is the symbol that the head of the TM is reading and p is the topmost stack symbol, q_j is the next state, Y is the symbol that is written in the cell pointed by the head of the TM, p' is the symbol that is pushed to the stack (*i.e.*, pop p, replacing it by pushing p'), and S indicates the direction of the head shift.

There might be cases when it is not important to know which symbol is at the top of the stack. For denoting such situations we will use λ within the transition function: $\delta(q_i,X,\lambda) = (q_j,Y,p',S)$; indicating the TM to pop the topmost stack symbol and replacing it by pushing p'.

3.2. Identifying the reused text

The proposed *Rewriting Index* method assigns a weight to each word contained in the suspicious document describing its degree of membership to a possible portion of reused text. Hence, it is able to identify portions of text that although they do not represent an exact match, they indicate highly probable plagiarized sections. In other words, this method is able to obtain non-consecutive portions of reused text and, therefore, to capture the common actions of a plagiarist such as word elimination, insertion and transposition.

In particular, the proposed method is an *ad hoc* search algorithm that uses a context window of size v, that contains v words from the original document D^O (*i.e.*, our search algorithm moves through the text of D^O). The position of this context window is defined by its middle word, which is, from a Turing machine perspective, the position where the head of the tape is pointing to. We will refer to the word positioned at middle of the context window as the *focus*.

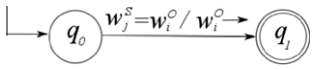

Fig. 1. TM capturing a verbatim copying case.

Therefore, we take for granted that the tape of our TM are the words contained in D^O (*i.e.*, the original document), represented by the string:

$$w_1^O w_2^O \ldots w_{i-1}^O q w_i^O w_{i+1}^O \ldots w_n^O \quad (2)$$

where the central word of the context window is the i-th word, which is the position where the head of the tape is pointing to; being q the actual state of the TM.[4] Notice that v has to be an odd number in order to have the same number of context words $\left(\frac{v-1}{2}\right)$ at the right and at the left of the *focus* word.[5]

The *Rewriting Index* algorithm will assign a *ReI* value to each word w_j^S (*i.e.*, the word at position j within the suspicious document D^S). To compute $ReI(w_j^S)$ we define five different TMs (Figs. 1–5). Each TM will assign a different *ReI* value (c_i) depending on: the position in D^O of the searched word w_j^S. That is, if the searched word appears at the *focus* the *ReI* is equal to c_1 indicating a verbatim case (Fig. 1); if the word appears at the right from *focus* it takes values c_2 or c_4 suggesting a moderate or large number of deletion/insertion operations respectively (Figs. 2 and 3); if it appears at the left of the *focus* it takes values c_3 or c_5 signifying a moderate or severe word transposition operation (Figs. 4 and 5); finally, if the searched word does not appears in D^O, its *ReI* value is equal to 0.

We assume that every TM acts over the same tape $(w_i^O \ldots w_n^S)$, and we will considerate only the changes (actions) made by the TM that reaches an accepting state. If more than one TM succeed, we will preserve those changes made from the one that obtains the higher *ReI* value. In general, the constants c_i fulfil the following condition: $c_1 > c_2 > c_3 > c_4 > c_5 > 0$. The following subsections describe in detail each one of the mentioned cases.

3.2.1. Capturing verbatim copies

The following automata (Fig. 1) is able to identify sequences of consecutive words that had been literally copied from the original document D^O. Notice that every time this TM reaches the final state q_1 the $ReI(w_j^S)$ will get the c_1 value.

The TM from Fig. 1 will reach an accepting state when the searched word w_j^S is equal to the word located at the *focus* (*i.e.*, w_i^O, the word pointed by the head of the tape). In this case, the TM leaves the same word on that cell of the tape and shifts one position to the right in order to search for another coincidence.

3.2.2. Capturing deletion/insertion operations

The TM described in Fig. 2 aims to identify moderate cases of word deletion and insertion operations. It is mainly able to identify if a few words, within the *local words* at the right of the *focus*, were deleted or inserted. If this situation occurs, the *focus* is moved to the symbol located after the position where $w_j^S = w_i^O$ was accomplished, the $ReI(w_j^S)$ is set to c_2, and the topmost stack symbol is set to 0 indicating that the position of the *focus* has changed. As we previously mentioned, our stack works as a counter and we assume that every time the TM is called, the initial stack symbol p is set to 0. Accordingly, every time the head of the TM is moved, p

[3] Notice that a null entry ε is different from the blank symbol B.

[4] TM notation assume that the word located at the head of the tape will always be w_i^O, *i.e.*, the *focus* word.

[5] From here we will refer to the words contained within the context window as *local words*, and to those outside the context window as *global words*.

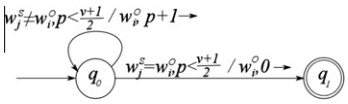

Fig. 2. TM capturing a moderate number of deletion and insertion operations.

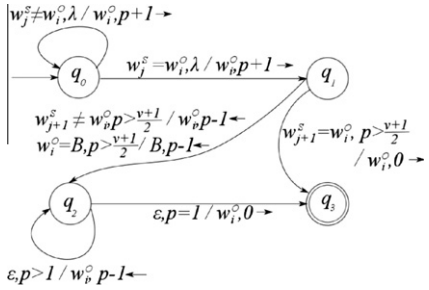

Fig. 3. TM capturing a severe number of deletion and insertion operations.

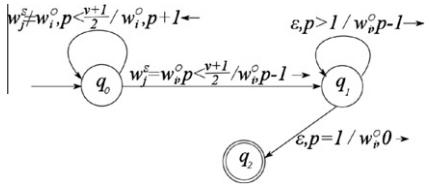

Fig. 4. TM capturing a moderate number of word transpositions.

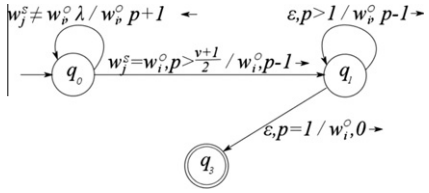

Fig. 5. TM capturing a severe number of word transpositions.

increases by 1 and the automata verifies if the head continues within the context window, *i.e.*, if $p < \frac{v+1}{2}$.

There are situations where plagiarists delete or insert several words between portions of plagiarized text, for such cases we define the TM shown in Fig. 3. Notice that the automata defined in this figure will search for w_j^S among the *global words* located at the right from the context window. If w_j^S is found, the TM verifies that the next entry $\left(w_{j+1}^S\right)$ also corresponds to a copied word (*i.e.*, verifies if this word is equal to the symbol pointed by the head of the tape), and if that is the case, it reaches the final state ($q_1 \rightarrow q_3$), updating the *focus* word by pushing 0 in the stack, and assigns the value c_4 to $RelI(w_j^S)$. If the later condition is not accomplished (*i.e.*, $w_{j+1}^S \neq w_i^O$), the TM returns the head of the tape to its initial position ($q_1 \rightarrow q_2 \rightarrow q_3$) by using the information provided by the word counter p. This step is performed since we consider that finding a single coincidence too far from the context window is not very relevant, but on the contrary, if two coincidences are found it is worth focusing on that section of the document.

3.2.3. Capturing word transpositions

Our method also considers plagiarism cases generated by word transposition operations, where the order of some words has been changed. In particular, the automata shown in Fig. 4 searches for w_j^S within the *local words* at the left of *focus*, whereas, the automata in Fig. 5 performs the same action but within the *global words* at the left of the context window. When these automata find the searched word w_j^S, they return the head of the tape to its initial position and assigns the value c_3 to $RelI(w_j^S)$ if the matching occurs within the *local words* (refer to Fig. 4), or a value of c_5 if it appears in the *global words* (refer to Fig. 5).

3.2.4. Complexity of the method

The *Rewriting Index* algorithm is able to provide a *RelI* value for each $w_j^S \in D^S$ in a time proportional to $O(m)$ in the best case, being m the number of words contained in D^S. In this case the suspicious document represents an exact copy of D^O. The worst case occurs when no word from the suspicious document occurs in the original document, which leads to a time proportional to $O(mn)$ being n the number of words contained in D^O.

3.3. Characterizing the reused text

Once evaluated each word w_j^S from the suspicious document as described in Section 3.2, we define a portion of reused text as a sequence of consecutive words ρ denoted by:

$$\rho = \langle w_k^S w_{k+1}^S \dots w_{l-1}^S w_l^S \rangle \tag{3}$$

where $k \leqslant j \leqslant l$, and satisfied: $RelI\left(w_j^S\right) > \tau$, $RelI(w_{k-1}^S) \leqslant \tau$ and $RelI(w_{l+1}^S) \leqslant \tau$, in order to consider only local words inside the portion of possible reused text.

Subsequently, we define P as the set of all the portions of reused text ρ contained in D^S. Then, in order to discriminate between plagiarized and non-plagiarized documents, we propose characterizing P by three main types of features, namely the *rewriting degree*, the *relevance* and the *fragmentation* features. The next expression shows the proposed representation of P.

$$\langle f^{RelI} | f_1^{rlv}, \dots, f_m^{rlv} | f_1^{frg}, \dots, f_{m'}^{frg} \rangle \tag{4}$$

We represent the set of portions of reused text by $1 + m + m'$ features, where f^{RelI} represents an agglomerative version of the *RelI* values computed with our proposed method (Section 3.2), and f_i^{rlv} and f_j^{frg} indicate the relevance and the fragmentation of the portions of reused text of length i and j respectively. Cases of particular interest are the f_m^{rlv} and $f_{m'}^{frg}$ features which indicate the values of all portions with length equal or greater than m and m' words. Their purpose is to deal with the data sparseness and to allow taking advantage of the occurrence of discriminative but very rare longer portions of reused text.

3.3.1. Rewriting degree feature

This feature aims to indicate the degree of plagiarized text contained in the suspicious document D^S; in other words, it represents how much the words from D^S were taken from D^O.[6] It is computed as an average of the *RelI* values from all the words contained in D^S as indicated in the following formula:

$$f^{RelI} = \frac{1}{|D^S|} \sum_{w_l^S \in D^S} RelI(w_l^S) \tag{5}$$

3.3.2. Relevance features

This group of features aims to quantify the portions of reused text by their words. That is, they aim to determine the relevance of the portions of reused text with respect to the thematic content

[6] This measure not only involves the number of shared words but also if they are in similar contexts.

of both documents. The idea behind these features is that frequent words or very small portions of reused text are related to the topic of the documents, and not necessarily are a clear signal of plagiarism. On the contrary, they are supported on the intuition that plagiarism is a planned action, and, therefore, that plagiarized sections are not used exhaustively.

In particular we measure the relevance of a given portion of reused text $\rho_k \in P$ by the formula:

$$rlvc(\rho_k) = \prod_{l=1}^{|\rho_k|} \frac{2}{occ\left(w_l^{\rho_k}, D^S\right) + occ\left(w_l^{\rho_k}, D^O\right)} \tag{6}$$

where $occ\left(w_l^{\rho_k}, D\right)$ indicates the times word w_l from fragment ρ_k occurs in D.

This measure of relevance castigates the portions of reused text formed by words that are frequent in both documents. The grater value (*i.e.*, $rlvc = 1$) occurs when the portion of reused text (and all its inner words) appear exclusively once in both documents, indicating that it has a great chance for being a deliberate copy.

Based on the definition of the relevance of a portion of reused text, relevance features are computed as follows:

$$f_i^{rlv} = \sum_{\{\rho_k : \rho_k \in P \wedge length(\rho_k) = i\}} rlvc(\rho_k) \tag{7}$$

The definition of the agglomerative feature f_m^{rel} is as follows:

$$f_m^{rlv} = \sum_{\{\rho_k : \rho_k \in P \wedge length(\rho_k) \geq m\}} rlvc(\rho_k) \tag{8}$$

Notice that the agglomerative feature represents the sum of the *rlvc* value for all the portions of reused text ρ_k with length equal or greater than m.

3.3.3. Fragmentation features

By means of these features we aim to find a relation between the length and quantity of portions of reused text and plagiarism. These features are based on two basic assumptions. On the one hand, we consider that the longer the portions of reused text, the greater the evidence of plagiarism. On the other hand, based on the fact that long portions of reused text are very rare, we consider that the more the portions of reused text, the greater the evidence of plagiarism.

According to these basic assumptions we compute the value of the f_j^{frg} feature by adding the lengths of all portions of reused text of length equal to j as described in the following formula:

$$f_j^{frg} = \sum_{\{\rho_k : \rho_k \in P \wedge length(\rho_k) = j\}} length(\rho_k) \tag{9}$$

Finally, the definition of the agglomerative feature $f_{m'}^{frg}$ is stated below:

$$f_{m'}^{frg} = \sum_{\{\rho_k : \rho_k \in P \wedge length(\rho_k) \geq m'\}} length(\rho_k) \tag{10}$$

4. Experiments and results

4.1. Datasets

For the experiments we used a subset of the METER corpus (Gaizauskas et al., 2001), a corpus specially designed to evaluate text reuse in the journalism domain. It consists of annotated examples of related newspaper texts collected from the British Press Association (PA) and nine British newspapers that subscribe to the PA newswire service. In particular, we only used the subset of news reports (suspicious documents) that have only one single related note (original document). This subset consists of 253 pairs of documents.

In this corpus each suspicious document (note from a newspaper) is manually annotated with one of three general classes indicating its derivation degree with respect to the corresponding PA news: *wholly-derived*, *partially-derived*, and *non-derived*. On the one hand, if a news report is tagged as *wholly–derived*, it means that all the information contained in the report can be extracted from the original PA newswire. On the other hand, if a news report is tagged as *partially–derived*, it means that there is some information contained in the report that was not extracted from the original PA newswire.

Notice that these labels do not provide any information about the type of plagiarism (*i.e,* complexity), since both the *wholly* and *partially* derived documents could be using a simple copy-paste strategy or a highly paraphrasing technique when using the information contained in the original PA newswire. Consequently, for our experiments we considered *wholly* and *partially* derived documents as examples of plagiarism and non-derived documents as examples of non-plagiarism, modelling in this way the plagiarism detection task as a two-class classification problem. In particular, the selected subset consists of 181 positive examples of plagiarism and 72 negative cases.

In addition, we also performed experiments using the *Plagiarised Short Answers* (PSA) corpus (Clough & Stevenson, 2010). Different to the METER corpus, this collection represents an explicitly-designed corpus of plagiarized documents. In this corpus each suspicious document is annotated with one of four general classes indicating its plagiarism degree with respect to the original document: *near–copy*, *light–revision*, *heavy–revision* and *non–plagiarism*. For the experiments we considered the four classes, handling the task as a multi-class classification problem. This corpus consists of 95 pairs of documents having the following distribution: 19 near copies, 19 light revisions, 19 heavy revisions and 38 cases of non-plagiarism.

Recently, the PAN-PC corpus[7] has also been used to evaluate plagiarism detection. This corpus includes plagiarism examples generated by translation and automatic methods and is used to evaluate methods that search for reused-text portions from a very large reference collection (Potthast, Stein, Eiselt, Barrón-Cedeño, & Rosso, 2010). Although the relevance of this resource, we decided not to use it because we are mainly interested in modelling and detecting human generated text reuse.

4.2. Evaluation

For the evaluation of the proposed approach, as well as the baseline methods, we employed the Naïve Bayes classification algorithm as implemented by Weka, and applied a 10 times repeated random sub-sampling 10 cross-fold validation strategy. In all cases, we preprocessed the documents by substituting punctuation marks by a generic label, but we did not eliminate stop words nor apply any stemming procedure.

The evaluation of results was carried out mainly by means of the classification accuracy, which indicates the overall percentage of documents correctly classified as plagiarized and non-plagiarized. Additionally, due to the class imbalance, we also present the macro-averaged F_1 measure as used in Clough et al. (2002).

4.3. On the selection of the parameter values

As indicated by the expression (4), we propose representing the portions of reused text in the suspicious document (D^S) by a vector

[7] http://pan.webis.de/.

of $1 + m + m'$ features. In this vector, the first feature indicates the overall degree of plagiarized text, whereas the rest of the features indicate the relevance and fragmentation of the portions of reused text of a particular length, except for the m and m'-features which integrate information from all portions of reused text with length greater than m and m' respectively.

In order to automatically determine an appropriate value of m and m', our method, before the classification process; computes the information gain value (IG), on the training set, of each obtained feature. This automatic process is as follows; given a training set, we extract portions of reused text of lengths varying from 1 to 50 resulting a representation of 101 features. Then, we evaluate the IG value of those features that obtained a sore greater than 0, and compute their mean value. Finally, we decided preserving those features having an IG greater than the mean value. Following this procedure our method established for the experiments reported in this paper the following values: for the METER corpus $m = 4$ and $m' = 4$, and for the PSA corpus $m = 5$, and $m' = 1$.

Another important parameter of the proposed method is the size v of the context window. Similar to the definition of m and m', we determined the value of v by evaluating the IG of the f^{Rel} feature considering v equal to 9, 15, 19, 25, and 29. This process indicated that using a window of size equal to 19 contributes the best for the proposed method in both corpora.[8]

In addition, our method requires the definition of the constants c_i, which are the values that each automata assigns when it succeed. For the experiments reported here, these constants were defined as: $c_i = 1/i$. Notice that such definition results in the following c_i values: $1 > \frac{1}{2} > \frac{1}{3} > \frac{1}{4} > \frac{1}{5} > 0$. It is also important to notice that these values satisfy the conditions required by the TMs to reach their final states. Finally, for the performed experiments we defined the threshold τ (see Section 3.3) as $\tau = c_4$.

4.4. Results

4.4.1. Baseline definition

As we previously mentioned, most current methods discriminate plagiarized from non-plagiarized documents by evaluating their degree of overlap with the original document using three main kinds of features, namely, single words, fixed length substrings (*i.e.*, word n grams), and variable length substrings. In particular, we generated the baseline results describing the overlap between the suspicious and original documents by means of: (i) the percentage of common words, *i.e.*, the Bag of Words (BOW) representation (*Baseline 1*), (ii) the percentage of common words extracted from the consecutive common sequences, *i.e.*, word n-grams (*Baseline 2*), and (iii) the percentage of common variable length substrings, *i.e.*, Common Sequences of variable length (*Baseline 3*). It is worth mentioning that these techniques have proved being very competitive (Potthast, Eiselt, Barrón-Cedeño, Stein, & Rosso, 2011) and they are considered as *hard-baselines* within the plagiarism detection field (see Section 2).

In addition, for the PSA corpus, we also present the results by Chong et al. (2010), which are the best results reported elsewhere for this collection. They measured the overlap between the suspicious and original documents by combining all previous features with information about their common syntactic dependency relations.

4.4.2. Experiments on the METER corpus

Table 1 presents the results on the METER corpus. They indicate that the proposed method achieved a higher accuracy and F_1 measure than the other approaches, outperforming the best baseline configuration (*i.e.*, BOW) by 5.24 % in terms of accuracy.

Table 1 shows that baseline results are very high (above 54% in terms of accuracy), demonstrating the relevance of the word intersection as main criterion for plagiarism detection. However, notice that our method considering 9 features <f^{Rel} (1), f^{rlv} (4), f^{frg} (4)>, which were automatically defined (Section 4.3) is able to perform a better classification process, indicating that there are in fact some actions that single word(s) overlap methods are unable to capture.

4.4.3. Experiments on the PSA corpus

Similar to the previous section, Table 2 compares the results from our method against the defined baselines, including, in this case, the best results reported in Chong et al. (2010). It is worth mentioning that Chong et al. (2010) used seven features that combine information at lexical and syntactic level: *Trigram Containment Measure (as baseline)*, *Baseline + Lem*, *Baseline + Stop + Pun + Num*, *Language Model – Bigram Perplexity*, *Language Model – Trigram Perplexity*, *Longest Common Subsequence and Dependency Relations*. Obtained results indicate that the proposed method clearly outperformed the best reported configuration (*Chong*) in accuracy and F_1 measure by 7.1% and 8.7% respectively.

It is important to notice that the *best* baseline configurations obtained in this experiment were very different from those generated with the METER corpus. These variations took place because of the different characteristics of the two corpora (Section 4.1); they mainly consisted in a better evaluation when the similarity between the suspicious and original documents is obtained using larger word n-grams and common sequences.

In order to carry out a deeper understanding of the proposed method, Table 3 shows the obtained performance by our method when different subsets of the proposed TMs are employed during the plagiarism detection task. The main goal of the experiments reported in Table 3 was to provide a detailed view of the behaviour of the proposed automata when detecting different types of plagiarism.

As it is possible to observe, using only the TM that identifies verbatim sequences allows to correctly classified the *near copy* and *non-plagiarism* cases, however the *heavy revision* class is commonly confused as *non-plagiarism*. Accordingly, using only the TM that detects transposition actions did not show an important improvement, nonetheless this automaton detects more accurately the *heavy revision* cases than the verbatim automaton. Finally, the automaton that detects deletion/insertion actions showed to be the more accurate across all the plagiarism classes. Nevertheless, using all the TM's results in better performance, particularly for the *heavy revision* cases, that are the most difficult to detect even for state-of-the-art methods, such as the one reported by Chong et al. (2010).

4.5. Further analysis

As we mentioned in Section 4.3 our method depends on the definition of three individual parameters, namely, m which is the number of the relevance features, m' that corresponds to the number of the fragmentation features and finally, v that represents the size of the context window.[9] In the following sections we present an analysis of the proposed plagiarism detection method when these parameters are manually defined for both the METER and the PSA corpus.

4.5.1. Additional experiments on the METER corpus

As we mentioned in Section 4.3, the process that automatically selects the parameter values in the METER corpus established

[8] Remind that the automatic process for defining the parameter values is repeated for each fold during the experiments; however, the reported values represent the statistical mode of the parameter values.

[9] Remember that m and m' also indicate the length of the portions of reused text that will be considered during the f^{rlv}_{m} and the $f^{frg}_{m'}$ computation (Section 3.3).

Table 1
Comparison of the proposed method against baseline approaches on the METER corpus.

Method	Features	Num. of features	Acc.	F_1 measure
Proposed	f^{Rel}, f^{lv}, f^{frg}	9	**77.2%**	**0.683**
Baseline 1	BOW	1	**73.1**%	**0.655**
Baseline 2	2-grams	1	**71.1**%	**0.674**
	3-grams	1	66.7%	0.644
	4-grams	1	66.0%	0.645
	5-grams	1	64.0%	0.630
	6-grams	1	62.8%	0.620
	7-grams	1	60.4%	0.597
	8-grams	1	58.1%	0.576
	9-grams	1	56.5%	0.563
	10-grams	1	54.1%	0.540
Baseline 3	CommSeqs (length $\geqslant 1$)	1	69.1%	0.592
	CommSeqs (length $\geqslant 2$)	1	**72.7**%	**0.677**
	CommSeqs (length $\geqslant 3$)	1	72.7%	0.676
	CommSeqs (length $\geqslant 4$)	1	69.1%	0.665
	CommSeqs (length $\geqslant 5$)	1	66.7%	0.651
	CommSeqs (length $\geqslant 6$)	1	66.7%	0.654
	CommSeqs (length $\geqslant 7$)	1	65.6%	0.644
	CommSeqs (length $\geqslant 8$)	1	63.6%	0.627
	CommSeqs (length $\geqslant 9$)	1	62.4%	0.616
	CommSeqs (length $\geqslant 10$)	1	60.0%	0.593

Table 2
Comparison of the proposed method against baseline approaches on the PSA corpus.

Method	Features	Num. of features	Acc.	F_1measure
Proposed	f^{Rel}, f^{lv}, f^{frg}	7	**75.9%**	**0.701**
Baseline 1	BOW	1	**61.0**%	**0.516**
Baseline 2	2-grams	1	65.2%	0.572
	3-grams	1	66.3%	0.589
	4-grams	1	65.2%	0.577
	5-grams	1	**67.3**%	**0.597**
	6-grams	1	67.3%	0.585
	7-grams	1	65,2%	0.569
	8-grams	1	66.3%	0.562
	9-grams	1	63.1%	0.517
	10-grams	1	62.1%	0.492
Baseline 3	CommSeqs (length $\geqslant 1$)	1	62.1%	0.522
	CommSeqs (length $\geqslant 2$)	1	63.1%	0.540
	CommSeqs (length $\geqslant 3$)	1	65.2%	0.574
	CommSeqs (length $\geqslant 4$)	1	63.1%	0.545
	CommSeqs (length $\geqslant 5$)	1	64.2%	0.566
	CommSeqs (length $\geqslant 6$)	1	67.3%	0.596
	CommSeqs (length $\geqslant 7$)	1	68.4%	0.603
	CommSeqs (length $\geqslant 8$)	1	**69.4**%	**0.614**
	CommSeqs (length $\geqslant 9$)	1	68.4%	0.599
	CommSeqs (length $\geqslant 10$)	1	65.2%	0.556
Chong	Combination	7	**70.53%**	**0.640**

Table 3
Performance comparison of the different TM's capturing different rewriting actions.

Captured rewriting actions	F_1 measure			
	Non plagiarism	Heavy revision	Light Revision	Nearcopy
Verbatim	0.923	0.464	0.169	0.660
Transpositions	0.935	0.586	0.403	0.644
Deletion/Insertion	0.918	0.549	0.459	0.418
All actions	**0.952**	**0.639**	0.483	**0.729**
Chong	0.925	0.564	**0.486**	0.588

using $m = 4$, $m' = 4$ and $v = 19$. Using these values our method to achieve a F_1 score of 0.683.

Accordingly, Fig. 6 depicts the performance of the proposed method, in terms of the F_1 measure, when varying the size of the relevance and fragmentation features (*i.e.*, m and m') as well as the size of the context window v. Notice that for these experiments we consider $m = m'$ (the same situation suggested by the automatic process), *i.e.*, the number of relevance and fragmentation features are always the same.

Notice that as we increase the size of $m = m'$ the performance of the proposed method declines, this means that, considering the relevance and fragmentation features of portions of reused text

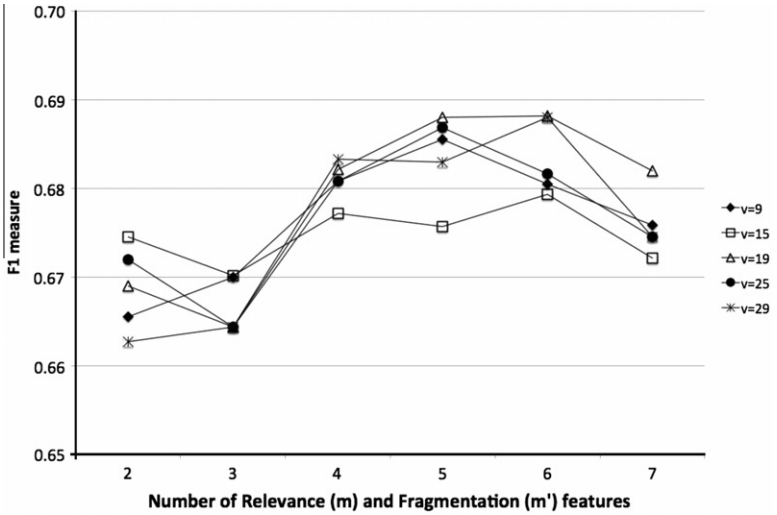

Fig. 6. Behaviour of the proposed method when varying the size of the context window v, and the number of the relevance and fragmentation features m and m' for the METER corpus.

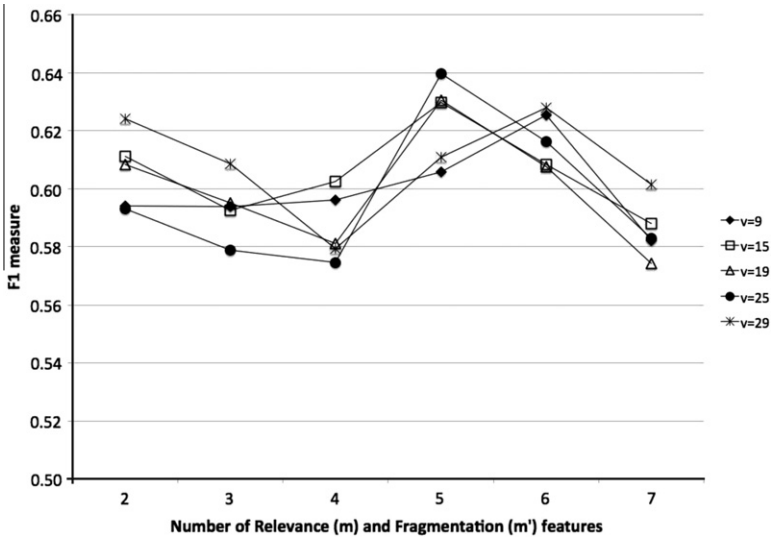

Fig. 7. Behaviour of the proposed method when varying the size of the context window v, and the number of the relevance and fragmentation features m and m' for the PSA corpus.

with length equal or greater than 6 is not very useful for the proposed method in the METER corpus. Furthermore, it is also possible to observe that the size for the context window v that allows to obtain higher values for the F_1 measure is, in most of the cases, $v = 19$. Particularly when m and m' are equal to 5, the method achieved a $F_1 = 0.688$.

As final conclusion, we can claim that the proposed heuristic for the automatic definition of the parameter values made a very good approximation of the optimal values, allowing to obtain a result that is only 0.72% below the best performance.

4.5.2. Additional experiments on the PSA corpus

Similarly to the previous section, Fig. 7 depicts the performance of our proposed algorithm when the three main parameters are manually fixed.

Notice that, similar to the METER corpus, for the PSA considering portions of reused text with length equal or greater than 6 results in a bad performance. Consequently, most of the higher F_1 scores are obtained when m and m' are equal to 5.

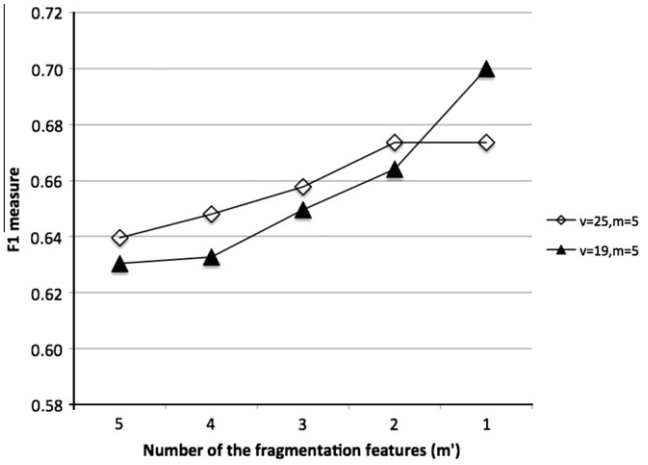

Fig. 8. Behaviour of the proposed method when varying the number of the fragmentation features m' for the PSA corpus.

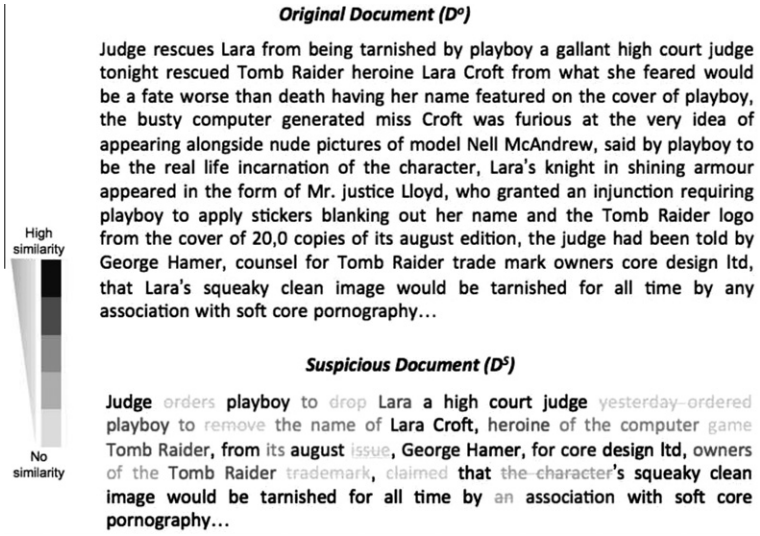

Fig. 9. An example of a possible plagiarized document D^S with its corresponding source D^O.

An important difference that we observed when performing these experiments is that apparently the best context window size was $v = 25$, allowing to obtain a F_1 score of 0.639. However, remember that the automatic process for defining the parameter values suggested that for the PSA corpus $m = 5$, $m' = 1$ and $v = 19$, resulting in $F_1 = 0.701$ (Table 2).

In order to provide a better understanding of this behaviour, we performed an additional set of experiments that consisted in fixing the values of the context window in 19 and 25, and also fixing $m = 5$; the only variation across experiments is the value of m'. Obtained results are shown in Fig. 8. Accordingly, our automatic process for defining the parameter values (▲) produces the configuration that allows to obtain the best performance.

4.6. A practical example

This section illustrates the *Rewriting Index* behaviour in a real scenario. Fig. 9 shows a pair of text fragments obtained from a couple of documents from the METER corpus.

As it is possible to observe, the first paragraph corresponds to the source document D^O, while the second one is the suspicious document D^S. In order to exemplify how the *ReI* method works, we use a greyscale code to mark those portions of reused text that were identified by our method. Accordingly, the most dark (black) words represent cases with a high similarity, whereas the most brighter words represent cases of no similarity (such words are marked by a middle line as:).

Table 4 shows the *fragmentation* and *relevance* values obtained for some of the portions of reused text identified by the *ReI* method. As mentioned in Section 3.3 the *fragmentation* features aim to find a relation between the length and quantity of portions of reused text and plagiarism; whereas the *relevance* features aim to determine the relevance of the portions of reused text with respect to the thematic content of both documents.[10]

According with these definitions, on the one hand, the first three portions of reused text (*playboy, heroine*, and*Raider*) although they are words that appear in both documents (D^O and D^S) they are not a clear evidence of a deliberate copy, since they are isolated words (*i.e.*, fragmentation equal 1) and very related to the main thematic (*i.e.*, low relevance values). On the other hand, the last two portions of reused text provide a greater evidence of plagiarism, since they have a greater number of words (*i.e.*, high fragmentation values) as well as they are formed by words that appear once in both documents and not thematic related (*i.e.*, high relevancy values).

Table 4
Fragmentation and relevance values of some of the identified portions of reused text.

Portion of reused text	Fragmentation	Relevance
Playboy	1	0.004
Heroine	1	0.027
Raider	1	0.006
Clean image would be tarnished for all time by	9	0.156
Association with soft core pornography	6	0.113

5. Conclusions and future work

In this paper we have proposed a new method for detecting document plagiarism. Its main contribution focuses on the identification of similar and – possible – reused word strings between a original and a suspicious document that are not necessary an exact copy. This method, called the *Rewriting Index*, assigns a weight to each word from the suspicious document in order to describe its degree of membership to a portion of plagiarized text. This way, it is able to discover text that has suffered from some modifications such as word elimination, insertion, and transposition, allowing to perform a partial matching between documents.

Another important contribution of this paper is the proposal of a richer representation of the portions of reused text. This new representation helps the classification algorithms to better discriminate between plagiarized and non-plagiarized documents by including features that describe not only the number of reused text portions but also their relevance and fragmentation. Additionally, we have proposed a simple methodology that allows for automatically select the best configuration of its three main parameter values.

Experimental results on the METER and PSA corpora are encouraging since they showed the appropriateness of the proposed method for the task at hand. Particularly, they outperformed the accuracy results from current methods by 5.2% and 7.1% on the METER and PSA corpora, respectively.

As future work we plan to improve the *Rewriting Index* method by considering synonyms and applying some morphological nor-

[10] Values showed in Table 4 were computed using the entire documents (*i.e.*, D^O and D^S).

malizations in order to capture the paraphrasing actions, which is a limitation of the current method. Furthermore, we are considering a non-lineal definition for the constants c_i; by doing so it will be possible to remark the importance of some particular plagiarism actions. In addition, we plan to explore the use of the *ReI* feature as a document similarity measure in other related tasks such as document categorization.

References

Barrón-Cedeño, A., Basile, C., Esposti, M. D., & Rosso, P. (2010). Word length *n*-grams for text re-use detection. In *11th International conference on intelligent text processing and computational linguistics (CICLing '10). LNCS* (Vol. 6008, pp. 687–699). Springer Verlag.

Barrón-Cedeño, A., Eiselt, A., & Rosso, P. (2009). Monolingual text similarity measures: A comparison of models over wikipedia articles revisions. In *Proceedings of ICON-2009: 7th international conference on natural language processing* (pp. 29–38). Hyderabad, India.

Barrón-Cedeño, A., & Rosso, P. (2009). On automatic plagiarism detection based on *n*-grams comparison. In *Proceedings of the 31th European conference on IR research on advances in information retrieval (ECIR). LNCS* (Vol. 5478, pp. 696–700). Berlin, Heidelberg: Springer-Verlag.

Basile, C., Benedetto, D., Caglioti, E., Cristadoro, G., & Degli Esposti, M. (2009). A plagiarism detection procedure in three steps: Selection, matches and "squares". In *Proceedings of the SEPLN 2009 workshop on uncovering plagiarism, authorship and social software misuse (PAN 2009)* CEUR-WS (Vol. 502). Donostia-San Sebastian, Spain.

Butler, M. (2010). Journals step up plagiarism policing. *Nature, 466*(7303).

Chien-Ying, C., Jen-Yuan, Y., & Hao-Ren, K. (2010). Plagiarism detection using ROUGE and WordNet. *Journal of Computing, 2*(3), 34–44.

Chong, B. M., Specia, L., & Mitkov, R. (2010). Using natural language processing for automatic detection of plagiarism. In *Proceedings of the 4th international plagiarism conference*. Newcastle-upon-Tyne, UK.

Clough, P. (2003). Old a new challenges in automatic plagiarism detection. *National Plagiarism Advisory Service*, 391–407.

Clough, P., Gaizauskas, R., Piao, S., & Wilks, Y. (2002). METER: Measuring text reuse. In *Proceedings of the 40th annual meeting of the association for computational linguistics (ACL)*. Philadelphia.

Clough, P., & Stevenson, M. (2010). Developing a corpus of plagiarised short answers. *Language Resources and Evaluation: Special Issue on Plagiarism and Authorship Analysis, 45*(1), 5–24.

Engles, S., Lakshmanan, V., & Craig, M. (2007). Plagiarism detection using feature-based neural networks. In *Proceedings of the 38th SIGCSE technical symposium on computer science education (SIGCSE '07)* (pp. 34–38).

Gaizauskas, R., Foster, J., Wilks, Y., Arundel, J., Clough, P., & Piao, S. (2001). The METER corpus: A corpus for analysing journalistic text reuse. In *Proceedings of corpus linguistics 2001* (pp. 214–223). Lancaster, UK.

Grozea, G. C., & Popescu, M. (2011). The encoplot similarity measure for automatic detection of plagiarism. In *Notebook for PAN at CLEF 2011*.

Grozea, G. C., & Popescu, M. (2010). Who's the thief? automatic detection of the direction of plagiarism. In *11th International conference on intelligent text processing and computational linguistics (CICLing '10). LNCS* (Vol. 6008, pp. 700–710). Iasi, Romania: Springer-Verlag.

HaCohen-Kerner, Y., Tayeb, A., & Ben-Dror. (2010). Detection of simple plagiarism in computer science papers. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)* (pp. 421–429). Beijing, China.

Hartrumpf, A. S., Brúck, T., & Eichhorn, C. (2010). Semantic duplicate identification with parsing and machine learning. In *Eleventh international conference on text speech and dialogue (TSD 2010). LNAI* (Vol. 6231, pp. 84–92). Brno, Czech Republic: Springer-Verlag.

Hoad, T. C., & Zobel, J. (2003). Methods for identifying versioned and plagiarized documents. *Journal of the American Society for Information Science and Technology, 54*(3), 203–215.

Nawab, R. M. A., Stevenson, M., & Clough, P. (2011). External plagiarism detection using information retrieval and sequence alignment. In *Notebook for PAN at CLEF 2011*. Amsterdam, Netherlands.

Oberreuter, G., L'Huillier, G., Ríos, S. A., & Velásquez, J. D. (2011). Approaches for intrinsic and external plagiarism detection. In *Notebook for PAN at CLEF 2011*. Amsterdam, Netherlands.

Potthast, M., Stein, B., Eiselt, A., Barrón-Cedeño, A., & Rosso, P. (2010). An evaluation framework for plagiarism detection. In *Proceedings of the 23th international conference on computational linguistics (Coling 2010)*. Beijing, China.

Potthast, M., Eiselt, A., Barrón-Cedeño, A., Stein, B., & Rosso, P. (2011). Overview of the 3rd international competition on plagiarism detection. In *Notebook for PAN at CLEF 2011*. Amsterdam, Netherlands.

Rao, S., Gupta, P., Singhal, K., & Majumder, P. (2011). External & intrinsic plagiarism detection: VSM & discourse markers based approach. In *Notebook for PAN at CLEF 2011*. Amsterdam, Netherlands.

Seo, F. J., & Croft, W. B. (2008). Local text reuse detection. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'08)*.

Shivakumar, D. N., & García-Molina, H. (1995). SCAM: A copy detection mechanism for digital documents. In *Proceedings of the second annual conference on the theory and practice of digital libraries*. Austin, Texas.

Si, A., Leong, H. V., & Lau, R. W. H. (1997). CHECK: A document plagiarism detection system. In *Proceedings of the 1997 ACM symposium an applied computing* (pp. 70–77). San Jose, CA, USA.

Stein, B., & Eissen, S. M. Z. (2006). Near similarity search and plagiarism analysis. In *Society 2006. As a selected paper from the 29th annual conference of the German classification society (GfKl)* (pp. 430–437).

Suárez, P., González, J. C., & Villena-Román, J. (2011). A plagiarism detector for intrinsic plagiarism. In *Notebook for PAN at CLEF 2011*. Amsterdam, Netherlands.

Zechner, M., Muhr, M., Kern, R., & Granitzer, M. (2009). External and intrinsic plagiarism detection using vector space models. In *Proceedings of the SEPLN 2009 workshop on uncovering plagiarism, authorship and social software misuse (PAN 2009)*. CEUR-WS (Vol. 502). Donostia-San Sebastian, Spain.