Copyright ⓒ 2019 KSII

The Sequence Labeling Approach for Text Alignment of Plagiarism Detection

Leilei Kong[2], Zhongyuan Han[2] and Haoliang Qi[1,3,*]
[1]School of Electronic Information Engineering, Foshan University, Foshan, China
[2]School of Computer Science and Technology, Heilongjiang Institute of Technology
Harbin, China
[3] State Key Laboratory of Digital Publishing Technology of China
Beijing, China
[e-mail: kongleilei1979@gmail.com, hanzhongyuan@gmail.com, haoliangqi163@163.com]
*Corresponding author: Haoliang Qi

Received October 13, 2018; revised December 21, 2018; accepted March 6, 2019; published September 30, 2019

Abstract

Plagiarism detection is increasingly exploiting text alignment. Text alignment involves extracting the plagiarism passages in a pair of the suspicious document and its source document. The heuristics have achieved excellent performance in text alignment. However, the further improvements of the heuristic methods mainly depends more on the experiences of experts, which makes the heuristics lack of the abilities for continuous improvements. To address this problem, machine learning maybe a proper way. Considering the position relations and the context of text segments pairs, we formalize the text alignment task as a problem of sequence labeling, improving the current methods at the model level. Especially, this paper proposes to use the probabilistic graphical model to tag the observed sequence of pairs of text segments. Hence we present the sequence labeling approach for text alignment in plagiarism detection based on Conditional Random Fields. The proposed approach is evaluated on the PAN@CLEF 2012 artificial high obfuscation plagiarism corpus and the simulated paraphrase plagiarism corpus, and compared with the methods achieved the best performance in PAN@CLEF 2012, 2013 and 2014. Experimental results demonstrate that the proposed approach significantly outperforms the state of the art methods.

Keywords: Plagiarism Detection, Text Alignment, Sequence Labeling, Probabilistic Graphical Model, Conditional Random Fields

This research was supported by the National Natural Science Foundation of China (No.61806075, No. 61772177), the National Social Science Fund of China (No.18BYY125), the Natural Science Foundation of Heilongjiang Province (No. F2018029), and the State Key Laboratory of Digital Publishing Technology of China.

http://doi.org/10.3837/tiis.2019.09.026 **ISSN : 1976-7277**

1. Introduction

The problem of plagiarism has recently increased because of the digital resources available on the Web. The research on plagiarism detection has gained the extensive attention from academia to industry: more and more research has been carried out, and lots of plagiarism detection software has been developed [1]. Text alignment is an essential and challenging task of plagiarism detection. Text alignment aims to identify all contiguous maximal-length passages of reused text between a suspicious document (containing the plagiarized text segments) and a source document (the document which the suspicious document plagiarizes) [1-3].

Numerous methods for text alignment are proposed with the vast majority based on heuristic methods. Generally, given a suspicious document and a source document, the methods for text alignment firstly determine whether the two text segments are a pair of plagiarism matches according to some predefined heuristic rules. And then, these identified matches are merged based on some heuristics rules to form the final contiguous plagiarism passages.

Heuristic-based methods for text alignment have achieved excellent performance on no-obfuscation and low-obfuscation plagiarism detection. Taking the highest *PlagDet* score (an overall evaluation metric of text alignment proposed by PAN (Uncovering Plagiarism, Authorship, and Social Software Misuse) in [1-3] and 1 is the optimal value) as an example, the method of R.Torrejón13 achieved 0.94 on PAN 2012 no-obfuscation sub-corpus, and the method of Oberreuter12 got 0.84 on PAN12 low-obfuscation sub-corpus. However, for artificial-high obfuscation plagiarism sub-corpus, the highest score on the *PlagDet* is only 0.39 [1,4].

The evaluation results described above show that the traditional heuristic methods have a serious limitation on text alignment. We analyze the primary reason may be the application of the heuristics. As we know, the heuristic methods depend more on the experiences of experts, which makes the heuristics lack of the abilities for continuous improvements. For example, the existing methods identify the matches based on some predefined rules and merge the matches using the predefined rules (such as merging the exact matches if they are adjacent in both suspicious and source document). However, such methods cannot define all the rules to deal with the complex relationships between the pairs of text segments and maybe neglect the observations extracted from the neighborhood of each match and the features expressing the structural relations at the different position, especially in the context of high obfuscation plagiarism. All of these have the inevitable consequences for text alignment.

To overcome the defects of heuristics, in this paper, we formalize the problem of text alignment as a sequence labeling issue at first, then, introduce the probabilistic graphical model of machine learning into the task of text alignment: giving an observation sequence composed by the pairs of text segments, tag the sequence with the specific labels to uncover the plagiarism. Based on Conditional Random Field (CRF), a classical probabilistic graphical model, we present a sequence labeling approach for text alignment in plagiarism detection, called TA-CRF. TA-CRF no longer follows the way of identifying the plagiarism matches and merging them separately. On the contrary, it combines the process of matching and merging into a single model. The proposed method also benefits from the CRF's discriminative learning framework that can model the various interrelationships between the text segments to decide whether the current text pair should belong to a plagiarism passage. Especially, using CRF, the proposed method can incorporate the arbitrary textual features and context features

as evidence more effectively. The contributions of the present work are as follows.

- We reformulate the text alignment in plagiarism detection as the context of sequence labeling and propose the sequence labeling approach for text alignment. The proposed method improves the existing heuristic methods for text alignment from a model perspective.
- We show that CRF is particularly suitable directed toward the problem of text alignment because of its ability to design the flexible observation functions. Furthermore, we consider a context neighborhood of text segments pairs, for more matching precisely these pairs using their positions.

The rest of this paper is organized as follows. In Section 2, we review the related works of text alignment in plagiarism detection. In Section 3, we analyze the problem of text alignment, propose the model based on CRF for text alignment, and describe the different design of the potential functions. In Section 4, we choose PAN@CLEF 2012 artificial high obfuscation plagiarism corpus and the simulated paraphrase plagiarism corpus, which have never achieved a significant improvement in recent years, to evaluate the various aspects of the proposed method, and we report the experimental results and the performance comparisons with the state-of-the-art text alignment methods. In the last section, we conclude our research.

2. Related Work

From the current typical work on text alignment in plagiarism detection, we review the concept and the primary approaches for text alignment.

Potthast *et al.* introduced the task of text alignment for plagiarism detection [1-3]: let t= (s_{susp}, d_{susp}, s_{src}, d_{src}) denote a plagiarism case where s_{susp} is a passage of suspicious document d_{susp} and s_{src} is a plagiarized version of some source passage in source document d_{src}. Given d_{susp}, the task of a plagiarism detection is to detect t by reporting a corresponding plagiarism detection t'= (r_{susp}, d_{susp}, r_{src}, d'_{src}). The process that t' detects t iff $s_{susp} \cap r_{susp} \neq \Phi$, $s_{src} \cap r_{src} \neq \Phi$, and $d_{src}=d'_{src}$ is called text alignment.

Given a document pair (d_{susp}, d_{src}), the current text alignment methods adopted the process described in **Fig. 1**, in which *matching* refers to identifying the matches from d_{susp} and d_{src}, and *merging* is to merge these short matches into aligned text passages of maximal length between the suspicious and the source documents, rather than the scattered matches [2, 3].

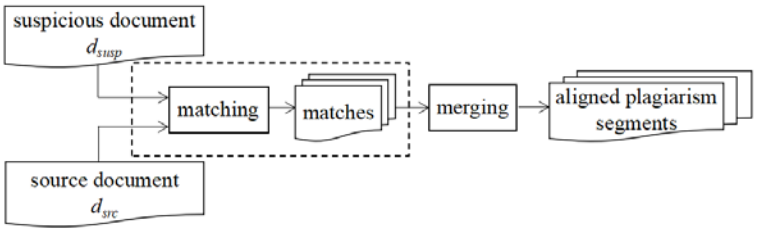


Fig. 1. Generic process of text alignment

The plagiarism matches are generally identified according to two strategies: exact matches or creating matches [1]. The exact matches are usually identified based on the features of character-based or word-based n-gram. The same character n-grams [5-7] or word n-grams [8-12] can be viewed as a pair of exact plagiarism match. The creating matches mainly rely on the features of text similarity. For example, Kong *et al.* [4, 13] and Momtaz *et al.* [14] used the sentences from the suspicious and source document as plagiarism matches based on their

Cosine distance. Sanchez-Perez *et al*. [15, 16] also applied the sentences whose sentence-overlaps were above some thresholds as matches. Stamatatos exploited the similarity of the stop word sequence to identify the plagiarism matches [17]. Daud *et al*. used Latent Dirichlet Allocation (LDA) and parts of speech (POS) tags to compute the semantic and syntactic similarities of the sentences to obtain the matches [18].

When giving the matches identified by the process of matching between the suspicious document and the source document, the matches are commonly merged into the aligned text passages using the rule-based heuristic methods with various constraints and restrictions [2]: merging the adjacent matches into passages in both suspicious and source document or merging the matches that the distances between them are below some threshold. For example, Suchomel *et al*. exploited the threshold of 4000 characters [19], Alvi *et al*. [20] and Gillam *et al*. [21] used 200 and 900 characters as threshold, respectively, and Kong *et al*. [4, 13] and Sanchez-Perez *et al*. [15] merged the adjacent sentences based on some similarity distance measures.

The rule-based heuristic methods impose some restrictions in text alignment: the predefined rules cannot cover all kinds of merging conditions, and sometimes these rules are conflicted with each other. When the possible distribution of position structure of candidate matches changing, such methods have no abilities to define all rules during the merging process. For example, the matches are commonly adjacent in no obfuscation and low obfuscation plagiarism since they can be identified more accurately, while for high obfuscation plagiarism, such as paraphrase or summary plagiarism, the distances among the candidates are not regular. The simple rules are difficult to deal with various complicated situations in merging, and however, are difficult to control.

In recent years, some biological information methods [5, 22], clustering-based methods [23, 24] and structure-based methods [25, 26] have introduced into the field of text alignment. However, these methods have not been successful than rule-based methods on the public evaluation corpus [3].

According to the framework of plagiarism detection proposed by the PAN@CLEF, the task of text alignment is defined as an “aligning” issue, yet, most of the existing research still stays in the heuristic matching and merging. The existing text alignment framework limits the further improvements of text alignment performance, which inevitably affects the performance of plagiarism detection. Boosting the text alignment performance requires the new perspective.

3. Sequence Labeling for Text Alignment

In this section, first we formalize the text alignment task as a sequence labeling problem and model text alignment by means of the probabilistic graphical model. Then, we analyze which of the probabilistic graphical model is more suitable for text alignment. Finally, we present the Text Alignment model based on CRF, denoted as TA-CRF.

3.1 Problem Analysis

Let $d_{susp}=\{s_1, s_2, ..., s_i, ..., s_n\}$ denote the suspicious document, and $d_{src}=\{r_1, r_2, ..., r_j, ..., r_m\}$ is its plagiarism source document, where s_i and r_j are the segments of text in d_{susp} and d_{src}, respectively (e.g., a sentence), n and m are the number of the segments in d_{susp} and d_{src}, and 1, 2, …, n (m) represents the positional order of these segments.

In previous research, each s_i is given the exact or approximate match(es) using a function $f(s_i, r_j)$. As described in Section 2, the function f commonly is defined by the heuristic-based

methods. In practice, for example, f can be a function that gets the exact matches of the same character-based or word-based n-grams, or the most probable one in all possible candidates, or identifies all the possible matches in the source document for each s_i. The Cosine distance, Dice distance and Jaccard coefficient have been chosen as the function f. **Fig. 2 (a)** describes the example of the matches.

Using the identified matches, we can get the aligned plagiarism passages as **Fig. 2 (b)** shown, where the dashed rectangles denote the probable plagiarism passages.

Observing the **Fig. 2 (b)**, if we can tag the label of each pair (s_i, r_j) accurately, then, it is easier to get the aligned plagiarism passages by exploiting the annotated labels. Thus, we conclude the problem of text alignment as: for a pair of input segments (s_i, r_j), we suppose that it is associated with an output label $y \in \{c_i\}$ indicating whether or not the pair should be considered in a plagiarized passage, then, the objective of text alignment is to tag the label for each (s_i, r_j).

Let t=(s_{susp}, d_{susp}, s_{src}, d_{src}) denote the set of plagiarism passages. Suppose that we tag the possible labels as (1) the pair is a beginning of a plagiarism passage, denoted as B; (2) the pair is an ending of a plagiarism passage, denoted as E; (3) the pair is a middle part of a plagiarism passage, denoted as M; and (4) the pair is not in a plagiarism passage, denoted as N. Then a probable labeling scheme in **Fig. 2 (a)** and **(b)** can be represented in **Fig. 2 (c)**.

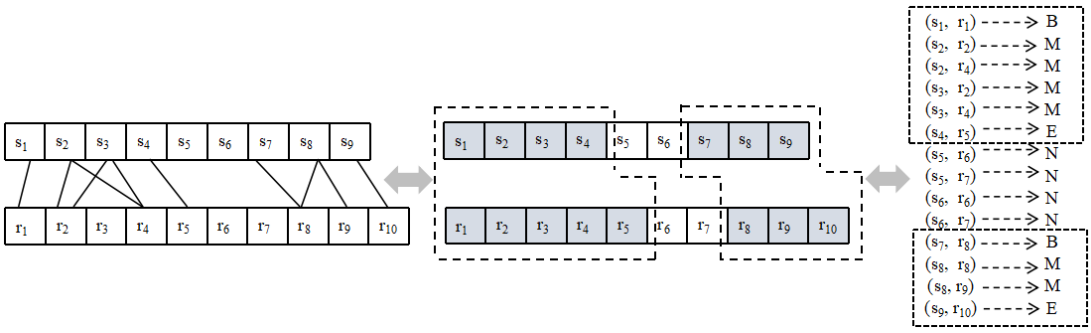


Fig. 2. Matching, merging and the pairs of matches with labels

What's more, the facts described above suggest an intuitive idea of learning a model to tag the pair (s_i, r_j) by applying the observed data for achieving the plagiarism passages. In the previous research, the labels of matches were determined by the heuristic-based methods, and each pair is considered independently, which makes the label of each match be merely regarded as the input of merging process and the relationship between the labels be not taken into account either. Given a sequence of pairs, different to the existing works, we want to tag the labels not only relying on the pairs themselves but also considering the relationship between labels, i.e., the context of the text segment. The analysis can be shown in **Fig. 3**. For comparing, we also list the idea proposed by the previous research.

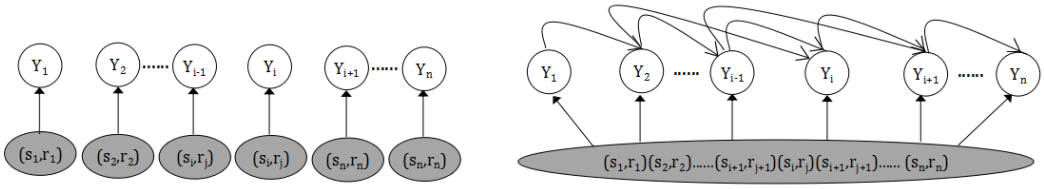


Fig. 3. The difference between the existing methods(left) and the proposed idea(right)

Note that given a pair (s_i, r_j) on the training data, we can get the output label of (s_i, r_j) to make learn a model for tagging possible. Given a collection of training data, let (s_i, r_j) denote a pair of text segment, $x_{i,j}$ be the feature vector extracted by function ϕ according to (s_i, r_j), and $y_{i,j}$ be the label of (s_i, r_j), and then the learning framework of text alignment can be depicted in **Fig. 4**.

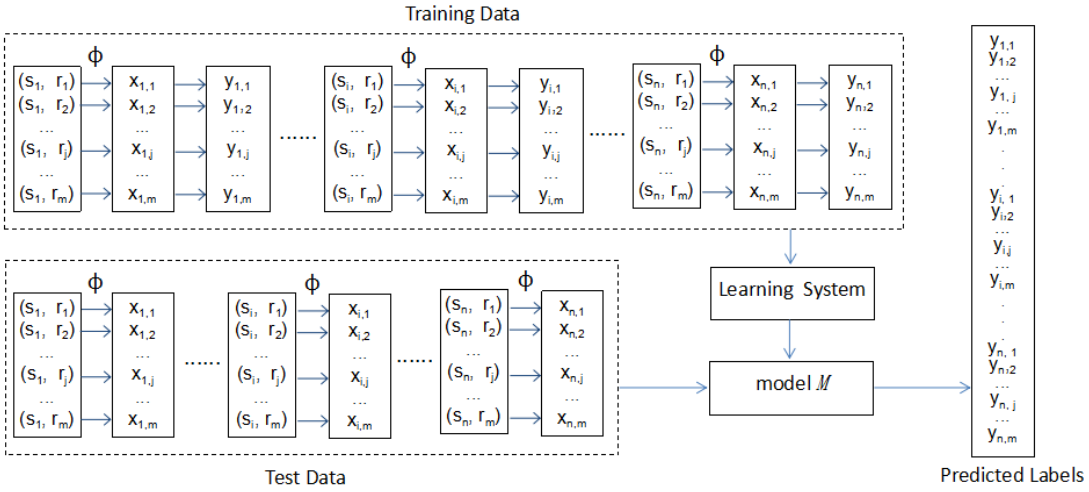


Fig. 4. The learning framework of text alignment

Observing the framework shown in **Fig. 4**, the essential problem of text alignment lies in: learning a model M for tagging the label $y_{i,j}$ on each observation data $x_{i,j}$, and predicting the labels using the learned model when given a new sequence of pairs.

3.2 Model Selection

In this paper, we convert the problem of text alignment into a machine learning problem instead of the heuristic methods.

Let T denote the sequence of pairs on training data,

$$T = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$$

Note that each y_i is a predefined discrete value. The analysis can be carried out from the following two angles.

If we view y_i as a category label to represent which categories x_i belongs to, then the model can be learned by the classification-based methods. In another word, if we view y_i as a discrete sequence of observation features x_i, which describes the tagging information of the current text pair, then the issue can be considered as a problem of classification. We can use the classification-based methods to learn model M, and the goal of the learning system is to learn the conditional probability $P(Y|X)$ on the training data. Then the learned classifier $P(Y|X)$ can be used to predict the output classification label y_{N+1} when given a new input instance x_{N+1}.

On the other hand, if we view the learning problem as an issue of sequence labeling and denote $x_i = (x_i^{(1)}, x_i^{(2)}, ..., x_i^{(n)})^T, i = 1,2,..., N$ as the input observation sequence, and the value of Y_i(i=1, 2, ..., n) are the sequence of golden labels, the learning system will learn a conditional probability distribution model M based on the training data $P(Y|X)$. Then given a new input observation sequence $x_{N+1} = (x_{N+1}^{(1)}, x_{N+1}^{(2)}, ..., x_{N+1}^{(n)})^T$, the Model M in **Fig. 4** gives the corresponding output label sequence Y_{N+1} according to the learned condition probability distribution model $P(Y|X)$. **Fig. 5** shows the classification model and the probabilistic

graphical model for sequence labeling (taking Hidden Markov Model and the Linear-chain CRF as examples) according to [27] and [28].

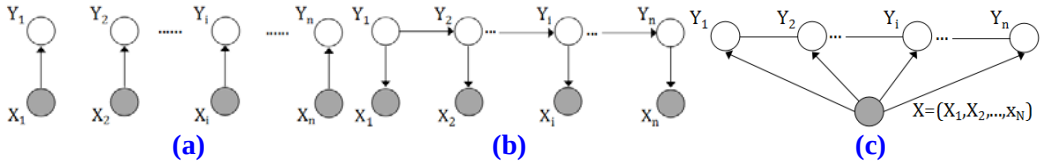


Fig. 5. The idea of Classification Model(a), Hidden Markov Model(b) and Linear-Chain CRF (c)

Classification Model. Observing **Fig. 5**, the classification model cannot take good advantage of the context information since each instance is considered independently. However, as we know, since most of the plagiarized segments are usually adjacent to each other in location, the information of candidate pair itself, the context information and the tagged labels are all the essential elements for text alignment. For example, for an identified plagiarized segment, it is very likely to "conjure" another plagiarized segment round its appearance, that is, if (s_i, r_j) is labeled as plagiarized, then (s_{i-1}, r_{j-1}) and (s_{i+1}, r_{j+1}) may be likely to be plagiarized. That is the basic idea of many text alignment algorithms adopted. Fortunately, these issues that the classification models have not resolved in text alignment may be avoided by the sequence labeling models based on probabilistic graphical in a particular way for its ability in modeling the probabilistic distribution and exploiting the prior probability.

Hidden Markov Model (HMM). HMM [29] assigns the joint probability to the paired observation and labels sequences and estimates the parameters according to maximizing the joint likelihood of training examples by enumerating all possible observation sequences, which are marginalized over the hidden variables. Hence, HMM is a generative method. HMM have been shown excellent performance in many applications. However, in our alignment problem, the objective is to find the values of the hidden variables, given the observations. Thus, at this point, HMM is not optimal. An existing common consensus is that the parameters or the model derived and optimized based on maximizing the discrimination function on training data may lead to higher accuracy on real-world problems [30, 31]. So, we employ a discriminative framework to address our alignment problem. On the other hand, the multiple interacting features or long-range dependencies of the observations cannot be represented in HMM [32].

Conditional Random Field (CRF). CRF is a discriminative model with the abilities to learn the temporal dependencies between node labels. Compared with HMM, CRF has some advantages directed toward the problem of text alignment. First, CRF specifies the probabilities of possible label sequences given an observation sequence. Therefore, they do not expend modeling effort on the observations [32]. Second, the conditional probability of the label sequence can depend on arbitrary features of the observation sequence without forcing the model to account for the distribution of those dependencies [32]. Lastly, the framework of CRF enables the relaxation of some conditional independence assumptions of Bayes network, thus authorizing more general dependency structures, without increasing the decoding complexity [33]. In particular, exploiting CRF, matching and merging, the two different phases in text alignment, can be integrated into a unified sequence labeling process. To the authors' knowledge, CRF has never been used for text alignment in plagiarism detection.

According to the analysis described above, CRF has advantages over the approaches based on the classification models and the HMM to the problem of text alignment. In the following sections, we formalize our text alignment task as a sequence labeling problem. And then, we

explain how to use the CRF to model the text alignment task.

3.3 Text Alignment as a Sequence Labeling Problem

3.3.1 Linear-chain CRF Framework for Text Alignment

The purpose of plagiarism text alignment is to tag the pair of text segments; or equivalently, to predict the label sequence of the given sequences of text segment pairs. We view this issue as a sequence labeling problem, in which the labels are the possible positions of the plagiarism segment, for example, the start or the end of a plagiarism passage.

We convert the pair of text segment (s_i, r_j) into a discrete sequence of observation features firstly. Let $x_i = (x_i^{(1)}, x_i^{(2)},..., x_i^{(n)})^T, i = 1,2,..., N$ be the input observation sequence of length N. Our task is to predict the corresponding label sequence $y_i = (y_i^{(1)}, y_i^{(2)},..., y_i^{(n)})^T$ that best matches the observed sequence. In the view of CRF, the learning system constructs a model M based on the training data, represented as a conditional probability distribution:

$$P(Y^{(1)}, Y^{(2)},..., Y^{(n)} \mid X^{(1)}, X^{(2)},..., X^{(n)})$$

where each $X^{(i)}$ (i=1,2, ..., n) is all possible observations, and each $Y^{(i)}$ (i=1,2, ..., n) is all possible labels. Then using this learned conditional probability distribution model, given a new observation sequence $x_{N+1} = (x_{N+1}^{(1)}, x_{N+1}^{(2)},..., x_{N+1}^{(n)})^T$, we can find the label sequence $y_{N+1} = (y_{N+1}^{(1)}, y_{N+1}^{(2)},..., y_{N+1}^{(n)})^T$ with the maximum conditional probability $P((y_{N+1}^{(1)}, y_{N+1}^{(2)},..., y_{N+1}^{(n)})^T \mid (x_{N+1}^{(1)}, x_{N+1}^{(2)},..., x_{N+1}^{(n)})^T)$.

Then the framework shown in **Fig. 4** can be specified as **Fig. 6**.

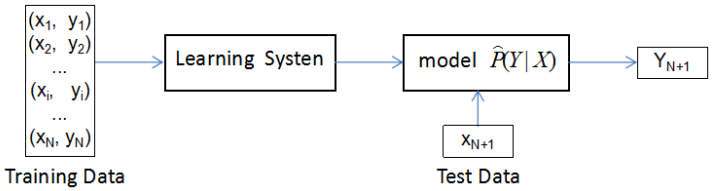


Fig. 6. The framework of sequence labeling based on CRF

Considering the linear-chain CRF shown in **Fig. 5**, let X=(X_1, X_2, ..., Xn) and Y=(Y_1, Y_2, ..., Yn) are all the random variable sequence represented by the linear-chain, then the linear-chain conditional probability distribution satisfies the following Markov property:

$$P(Y_i \mid X, Y_1,..., Y_{i-1}, Y_{i+1},..., Y_n) = P(Y_i \mid X, Y_{i-1}, Y_{i+1}), i = 1,2,..., n \quad (1)$$

Suppose the value of random variable X is x, then the probability of Y=y can be written as (Lafferty et al., 2001):

$$P(y \mid x) = \frac{1}{z(x)} \exp\left(\sum_{i,k} \lambda_k t_k (y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l (y_i, x, i) \right) \quad (2)$$

where $Z(x)$ is a normalization constant to make the distribution sums to one.

$$Z(x) = \sum_y \exp\left(\sum_{i,k} \lambda_k t_k (y_{i-1}, y_i, x, i) + \sum_{i,l} \mu_l s_l (y_i, x, i) \right) \quad (3)$$

In Eq. (2), t_k is the transition feature function, s_l is the status feature function, and λ_k and μ_l are the corresponding weights of t_k and s_l.

t_k is defined on the edges to control the transitions between the labels and is used to depict the correlation between the two adjacent labeling variables and the impacts of the observation sequence on labels. Different choices for the transition feature function t_k can control the different prior probabilities of the label sequences. Since the label denote whether the current text segment is the plagiarized one, the objective of the transition functions is to model the label of each pair of text segment accurately.

s_l is defined on vertices to link the current label to the observation variables and describe the impacts of the observations on the labeling variables. The status feature function reflects the information represented by the labels. When giving a pair of candidate text segment, the status feature function is used to judge whether the current pair of text segment belongs to a plagiarized passage.

In our task of text alignment, we suppose that t_k and s_l are all the given functions with predefined features, for example, a Boolean vertex feature s_l is "true" if the Cosine distance of the pair of text segment is above a threshold and the tag y_i is "in the middle of a plagiarism segment", and a transition feature function t_k is "1" if the number of the common words is above a threshold, the tag y_i is "in the middle of a plagiarism segment" and the tag y_{i-1} is "at the beginning of a plagiarism segment".

Since the same feature has the definition at each position i, the linear-chain CRF defined in Eq. (2) can be simplified as Eq. (4):

$$P(y \mid x) = \frac{1}{z(x)} \exp \sum_{k=1}^{K} w_k f_k(y, x) \tag{4}$$

And the normalization factor $Z(x)$ is:

$$Z(x) = \sum_{y} \exp \sum_{k=1}^{K} w_k f_k(y, x) \tag{5}$$

where K is the total member of transition features and status features. Using the equations described in Eq.(4) and (5), the most important consideration when defining our linear-chain CRF for text alignment lies in specifying the features $f_k(y, x)$ of the templated factor and learning its parameters.

3.3.2. Features

In this paper, we consider several types of features in order to model text alignment accurately. In this section, we propose three types of features reference to [32] and transpose them into the linear-chain CRF's framework by exhibiting the corresponding feature functions.

Label-observation features. If we are interested in the features depending only on the input observation, we could define the status feature function as our templated factors as follows:

$$f_k(y, x) = 1_{\{y=\tilde{y}\}} q_k(x) \tag{6}$$

where $q_k(x)$ is the input observation function described in 3.3.3 and each feature is nonzero only for a single output configuration $\tilde{y}$. The label-observation features mean that each feature is nonzero only for a single output configuration $\tilde{y}$, and we just need to learn a separate set of weights for each output configuration. For label-observation features, each label of the pair of text segment is determined only relying on the observation function defined on the pairs of text segments.

Edge-observation features. If the factor for every transition can depend on all of the observation functions, it is called edge-observation features [27]. We define edge-observation feature in Eq. (7):

$$\begin{aligned} f_k(y_{i-1}, y_i, x_i) &= q_k(x_i)1_{\{y_i=y\}}1_{\{y_{i-1}=y'\}} \quad \forall y, y' \in C, \forall k \\ f_k(y_i, x_i) &= q_k(x_i)1_{\{y_i=y\}} \quad \forall y \in C, \forall k \end{aligned} \tag{7}$$

where y_{i-1} is the neighboring label of y_i and C is the set of labels. Edge-observation features allow the use of current observation functions, the current label, and its contiguous labels to define features. So that the features like "Cosine Distance x_i is above 0.8, label y_i is Middle and label y_{i-1} is Beginning" can be used in edge-observation features.

Node-observation features. In the style of node-observation features, the transition factors no longer depend on observation functions, but only decided by its neighboring label. We define the node-observation feature in Eq. (8):

$$\begin{aligned} f_k(y_{i-1}, y_i, x_i) &= 1_{\{y_i=y\}}1_{\{y_{i-1}=y'\}} \quad \forall y, y' \in C, \forall k \\ f_k(y_i, x_i) &= q_k(x_i)1_{\{y_i=y\}} \quad \forall y \in C, \forall k \end{aligned} \tag{8}$$

Using node-observation features, the feature "Cosine Distance x_i is above 0.8, label y_i is Middle and label y_{i-1} is Beginning" can be simplified as "label y_i is Middle and label y_{i-1} is Beginning" and "Cosine Distance x_i is above 0.8 and label y_i is Middle".

3.3.3. Observation Functions and Labels

For simplicity, we have used just some common text similarity distance measurements based on the lexical matching as the observation functions, as shown in **Table 1**, where s_i and r_i are the text segments of the suspicious document and the source document, respectively, $|s_i|$ and $|r_i|$ are the total terms number of s_i and r_i, $|s_i \cap r_i|$ is the number of matching terms (or characters), $|s_i \cup r_i|$ is the total number of terms (or characters) of s_i and r_j, x and y are the term vectors of s_i and r_j, $\|x\|$ and $\|y\|$ represent the length of the vector x and y, m is the number of matching characters of s_i and r_j, and t is half the number of transpositions s_i and r_j.

Table 1. Observation functions

Observation Functions	Computing Methods	Observation Functions	Computing Methods
Jaccard Coefficient	$JC(s_i, r_j) = \frac{\lvert s_i \cap r_j \rvert}{\lvert s_i \cup r_j \rvert}$	Euclidean Distance	$ED(s_i, r_j) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
Common Char Length Ratio	$CCLR(s_i, r_j) = \frac{2 \cdot m}{\lvert s_i \rvert + \lvert r_j \rvert}$	Cosine Distance	$CD(s_i, r_j) = \frac{x \cdot y}{\lVert x \rVert \cdot \lVert y \rVert}$
Matching Coefficient	$MC(s_i, r_j) = \frac{\lvert s_i \cap r_j \rvert}{\lvert s_i \rvert + \lvert r_j \rvert}$	Jaro Distance	$JD(s_i, r_j) = \frac{m}{3 \times \lvert s_i \rvert} + \frac{m}{3 \times \lvert r_j \rvert} + \frac{m-t}{3 \times m}$
Ngram Distance	$ND(s_i, r_j) = \sum_{i=1}^{n} \frac{\lvert x_i - y_i \rvert}{\lvert x_i \rvert - \lvert y_i \rvert}$	Levenshtein Distance	——

We define two types of labels in our model. One is defined as C_1={0,1}, where 0 indicates that the current pair of text segment is not involved in belong to a plagiarized passage,

otherwise the opposite. Since sometimes the boundary labels have different characteristics than other labels, for example, there being a non-plagiarized segment ahead and an identified plagiarized segment followed usually indicates a beginning of a plagiarized passage, we distinct the boundary label with others, and define another type of label, denoted as C_2={*B*, *E*, *M*, *N*}, where *B*, *E* and *M* are the beginning, ending and middle part of a plagiarized passage, respectively, and *N* denotes the current pair is not in a plagiarism passage.

3.3.4. Parameters and Decoding

According to Eq. (2), the Linear-chain CRF are determined by the transition feature function t_k, status feature function s_l, and the corresponding weights λ_k and μ_l. The parameters $\theta = (\lambda_1, \lambda_2, ...; \mu_1, \mu_2, ...)$ is determined from the training data *T* with empirical distribution $\tilde{P}(x, y)$. Lafferty described an iterative scaling algorithm that maximizes the log-likelihood objective function to learn the parameters [32].

Given a CRF $P(Y|X)$ and an observation sequence *X*, the purpose of prediction is to work out a labeling sequence y^* with maximum conditional probability, in another word, tagging the observation sequence:

$$\begin{aligned} y^* &= \arg\max_y P_w(y \mid x) \\ &= \arg\max_y \frac{1}{z(x)} \exp \sum_{k=1}^{K} w_k f_k(y, x) \\ &= \arg\max_y \sum_{k=1}^{K} w_k f_k(y, x) \end{aligned} \tag{9}$$

For computing the most probable assignment y^*, we use the famous Viterbi algorithm to assign a sequence of labels to a new input [27, 32].

4. Experiments

In this section, we report the experimental results on the public available plagiarism detection corpus PAN@CLEF 2012. We first describe the experimental datasets used in Section 4.1. Sections 4.2 contains the description of baselines and the experimental evaluation metrics. In Section 4.3, the proposed method TA-CRF with the state-of-the-art methods are compared using the performance metrics of *PlagDet*, *Recall*, *Precision* and *Granularity*, and we show that TA-CRF improves the text alignment performance for high-obfuscation plagiarism detection significantly.

4.1. Datasets

The datasets used in this work is the text alignment corpus of PAN@CLEF 2012, which has been used for evaluating the participating text alignment algorithms in 2012, 2013 and 2014 plagiarism detection tracks. PAN@CLEF is a plagiarism detection algorithms evaluation competition organized by Cross-Language Evaluation Forum (CLEF). Following PAN@CLEF official runs, we denote the corpus as *PAN12 training corpus* and *PAN12 testing corpus*, respectively. These datasets contain the various plagiarism types designed by PAN@CLEF. Since the no obfuscation and the low obfuscation plagiarism detection have yielded perfect results: the highest *PlagDet* is 0.9452 on no obfuscation sub-corpus [9] and 0.8442 on low obfuscation sub-corpus [2], we only choose two high obfuscation sub-corpus, PAN12-artificial-high-obfuscation sub-corpus and PAN12-simulated-paraphrase sub-corpus,

to evaluate the proposed approach. The statistics for the two sub-corpuses are described in **Table 2**.

Table 2. Statistics for the PAN 2012 text alignment corpus

Sub-Corpus	# of Suspicious Document		# of Plagiarism Cases		Avg. Words of suspicious and source document		Avg. Cosine Similarity	
	Training Corpus	Testing Corpus	Training Corpus	Testing Corpus	Training Corpus	Testing Corpus	Training Corpus	Testing Corpus
12-artificial-high-obfuscation	1000	500	3,244	584	4,060,061	567,789	0.423	0.347
12-simulated-paraphrase	1000	500	1,081	1,135	293,697	296,631	0.477	0.479

4.2. Experimental Settings

4.2.1. Baselines

For comparing the effectiveness, we choose Kong12 [13] as our strong baselines. And other three state-of-the-art methods, Sanchez-Perez14 [15], Oberreuter12, and R. Torrejón13, which achieved the first place in other PAN12 sub-corpus for reference.

Kong12 succeeded in winning the first place not only on entire PAN12 testing corpus [1] in the evaluation of PAN@CLEF 2012 but also on entire PAN13 testing corpus2 [2] in the evaluation of PAN@CLEF 2013 with respect to *PlagDet*. Kong *et al*. proposed to use the Cosine distance and the Jaccard coefficient to compare the similarity of two vectors at the sentence-level [2] to obtain the matches, and used a heuristic algorithm to merge the matches. More details can be found in [4] and [13].

Sanchez-Perez14 achieved the highest *PlagDet* on the entire PAN13 testing corpus2 in the evaluation of PAN 2014 [3, 15]. Sanchez-Perez *et al*. applied the Cosine distance and the Dice coefficient to get the matching sentence pairs and a kong12-like heuristic algorithm was used to merge the plagiarism matches [15].

Oberreuter12 got the first place on the entire PAN12 testing corpus in the evaluation of PAN 2013 and the second place on the entire PAN13 testing corpus in the evaluation of PAN 2013 [2] with respect to *PlagDet*. They applied character 18-grams as features to obtain the exact matches and merged the matches if they were adjacent in position.

R. Torrejón13 won the best performance in all the algorithms submitted to PAN in 2013 [2]. The sorted word 3-grams and two kinds of sorted word 1-skip-3-grams were used as the matching features [9]. Similarly, a position-based algorithm was used to merge the matches.

4.2.2. Performance Measures

According to the set of PAN, the most critical performance measure of the text alignment is the *PlagDet*. We use *PlagDet* as the main evaluation measure to evaluate the proposed approach, TA-CRF. The other evaluation measures, such as *Precision*, *Recall*, and *Granularity* are also provided for reference in our experiments results.

Let T denote the set of plagiarism cases in the corpus and R denote the set of detection reported by a plagiarism detector for the suspicious documents. To simplify notation, a plagiarism case $t=<s_{susp}, d_{susp}, s_{src}, d_{src}>$, $t \in T$, is represented as a set t of references to characters of d_{susp} and d_{src}, specifying the passages s_{susp} and s_{src}. Likewise, a plagiarism detection $r \in R$ is represented as r. Based on this notation, *Precision* and *Recall* of R under T can be measured as follows.

$$Precision(T,R)=\frac{1}{|R|}\sum_{r\in R}\frac{\left|U_{t\in T}(t\cap r)\right|}{|r|} \tag{10}$$

$$Recall(T,R)=\frac{1}{|T|}\sum_{t\in T}\frac{\left|U_{r\in R}(t\cap r)\right|}{|t|} \tag{11}$$

where if r detects t then $t\cap r$ equals to $t\cap r$, otherwise, $t\cap r$ is Φ.

PAN@CLEF takes the influence of scattered passages into account and designs an evaluation measure *Granularity* to punish the scattered matching passages, and integrates *Granularity* into *PlagDet*, the main evaluation measure of the text alignment algorithm [1-3]. A detector's *Granularity* is quantified as follows:

$$Granularity(T,R)=\frac{1}{|T_R|}\sum_{t\in T_R}|R_T| \tag{12}$$

where $T_R\subseteq T$ are cases detected by detection in R, and $R_T\subseteq R$ is the detection of t.

All measures described above are combined into a single overall score *PlagDet* as follows.

$$PlagDet(T,R)=\frac{F_1}{\log_2(1+Granularity(T,R))} \tag{13}$$

where F_1 is the equally weighted harmonic mean of *Precision* and *Recall* that are defined in Eq. (14).

$$F_1=\frac{2*Recall*Precision}{Recall+Precision} \tag{14}$$

4.2.3. Training of Parameters

Constructing the training data. For learning TA-CRF, we leveraged the *PAN12 training corpus* to construct the training data. Given the set of plagiarism case $s=<s_{susp}, d_{susp}, s_{src}, d_{src}>$, we split d_{susp} and d_{src} into sentences firstly. For each sentence s_i in d_{susp}, if it belonged to s_{susp}, the sentence r_j in s_{src} having the highest Cosine distance with s_i was selected as the actual match of s_i, denoted as 1 or *B*, *M* and *E*, otherwise, if it did not belong to s_{susp}, the sentence r_j in d_{src} but not in s_{src} with the highest Cosine distance with s_i was chosen, denoted as 0 or *N*. Then we got a sentence sequence with labels as the training corpus.

Learning TA-CRF model. On the training corpus, we trained six TA-CRF models according to the different features and labels described in 3.3, shown in **Table 3**.

Table 3. TA-CRF models

	Label-observation features	**Edge-observation features**	**Node-observation features**
Label c={0,1}	TA-CRF-LOF-L1	TA-CRF-EOF-L1	TA-CRF-NOF-L1
Label c={*B*, *M*, *E*, *N*}	TA-CRF-LOF-L2	TA-CRF-EOF-L2	TA-CRF-NOF-L2

The tools CRF++[1] was used with all the other parameters setting to their default values except the parameter c (this parameter trades the balance between over-fitting and under-fitting).

4.2.4. Getting Candidate Matches

Since the proposed method TA-CRF works on the observation sequence, we need to prepare

[1] https://taku910.github.io/crfpp/

the candidate matches for generating the observation sequence. Followed Kong12, we split the suspicious document and source document into sentences, used the Porter stemmer for stemming, and removed the stopwords. For reasons having to do with efficiency, we chose the top one strategy to get the candidate matches: given a sentence s_i in a suspicious document, only choosing the most similar sentence r_j in the source document with the maximum Cosine distance as the match of s_i.

For high obfuscation plagiarism, this kind of strategy might miss some right plagiarism matches. However, we believe TA-CRF can rectify the inaccurate matches by exploiting the context features. The experimental results also proved our predictions.

4.2.5. Post Processing

We merged the sentences into passages according to their labels and discarded the passage pairs when their Jaccard coefficients were below a threshold followed by Kong12.

4.3. Experimental Results

In the following experimental results, we reported the per-category *PlagDet* of text alignment for the baselines and the proposed method. *Precision*, *Recall* and *Granularity* were also listed for reference. The bold values represented the best results per category. The notation *, #, & and ~ indicated that the proposed methods were statistically significant over the baselines Kong12, Sanchez-Perez14, Oberreuter12 and R. Torrejón13 respectively on the main evaluation metric *PlagDet* at the $p < 0.05$ level by using a one-tailed paired t-test. The performance reported were tested using TIRA evaluation platform for plagiarism detection [34] or published in [1-3].

Table 4 and **Table 5** illustrated the results of the proposed models and the baselines. For comparison, we also listed the official run results of Sanchez-Perez14, Oberreuter12 and R.Torrejón on *PAN12 testing corpus* according to [1-3]. The italics represented the best *PlagDet* on each sub-corpus with different plagiarism type in all submitted algorithms in PAN 2012 and 2013. Following PAN, we took *PlagDet* as the main evaluation metric.

The results in **Table 4** and **Table 5** demonstrated that among all the methods, the proposed models achieved the best results in all two collections. It was noticeable that the proposed six models of TA-CRF outperformed all the baselines significantly on PAN12-artificial-high-obfuscation subcorpus. Taking TA-CRF-LOF-L1 as an example, the relative improvements in TA-CRF-LOF-L1 over Kong12, Sanchez-Perez14, Oberreuter12 and R. Torrejón13 were 39.62%, 37.81%, 36.12% and 276.60%, respectively. On PAN12-simulated-paraphrase subcorpus, the proposed six models achieved the statistically significant improvements over Sanchez-Perez14, Oberreuter12 and R. Torrejón13, and the models TA-CRF-EOF-L1 and TA-CRF-NOF-L1 outperformed the strong baseline Kong12 significantly.

The experimental results showed that the performance improvements were mainly due to the improvement of *Recall*. Additionally, many experimental results also showed the improvement on both *Precision* and *Recall*. For example, TA-CRF-LOF-L1 performed 2.49% better than Kong12 on *Precision* and 15.63% on *Recall*. Compared with other baselines, the *Precision* and *Recall* were also improved, such as TA-CRF-EOF-L1, TA-CRF-NOF-L1 and TA-CRF-LOF-L2 on PAN12-simulated-paraphrase subcorpus.

On PAN12-artificial-high-obfuscation subcorpus, the improvements of *PlagDet* was extremely valuable. As we known, Kong12 gave fifty matches to a given suspicious sentence and then merged the identified matches via a heuristic method. However, for TA-CRF, we gave only one match for a given suspicious sentence. The better performance benefits from the

Table 4. Performance Comparison of the proposed TA-CRF and the baselines on PAN12 Testing Corpus PAN12-artificial-high-obfuscation subcorpus

		PlagDet	Precision	Recall	Granularity
Baselines	Kong12	0.3965	0.7504	0.2771	1.0076
	Sanchez-Perez14	0.4017	0.8404	0.3021	1.1532
	Oberreuter12	*0.4067*	***0.8790***	*0.2665*	*1.0290*
	R. Torrejón13	0.1470	0.5437	0.0889	1.0556
Our methods	TA-CRF-LOF-L1	**0.5536**$^{*\#\&\sim}$	0.7753	**0.4334**	**1.0061**
	TA-CRF-EOF-L1	0.4932$^{*\#\&\sim}$	0.7816	0.3696	1.0246
	TA-CRF-NOF-L1	0.4998$^{*\#\&\sim}$	0.7743	0.3736	1.0116
	TA-CRF-LOF-L2	0.5299$^{*\#\&\sim}$	0.7828	0.4064	1.0134
	TA-CRF-EOF-L2	0.4901$^{*\#\&\sim}$	0.7511	0.3800	1.0417
	TA-CRF-NOF-L2	0.4819$^{*\#\&\sim}$	0.7266	0.3681	1.0195

Table 5. Performance Comparison of the proposed TA-CRF and the baselines on PAN12 Testing Corpus PAN12-simulated-paraphrase subcorpus

		PlagDet	Precision	Recall	Granularity
Baselines	Kong12	*0.7588*	*0.9006*	*0.6577*	*1.0025*
	Sanchez-Perez14	0.7370	0.9460	0.6349	1.0434
	Oberreuter12	0.7173	0.9092	0.5923	**1.0000**
	R. Torrejón13	0.6805	**0.9752**	0.5273	1.0083
Our methods	TA-CRF-LOF-L1	0.7628$^{\#\&\sim}$	0.9544	0.6371	1.0025
	TA-CRF-EOF-L1	**0.7962**$^{*\#\&\sim}$	0.9536	0.6854	1.0024
	TA-CRF-NOF-L1	0.7954$^{*\#\&\sim}$	0.9576	**0.6862**	1.0072
	TA-CRF-LOF-L2	0.7608$^{\#\&\sim}$	0.9540	0.6346	1.0025
	TA-CRF-EOF-L2	0.7669$^{\#\&\sim}$	0.8856	0.6805	1.0048
	TA-CRF-NOF-L2	0.7643$^{\#\&\sim}$	0.8848	0.6758	1.0036

accurate sequence labeling. Even if there was only one match (maybe not accurate), due to considering the context, TA-CRF used the learned model to tag each match more accurately.

In addition, **Table 4** and **Table 5** also showed that the performance results achieved by the models that used 0 and 1 as labels were better than those using *B*, *M*, *E*, and *N* as labels. We found that the errors mainly come from the boundary labels that were labeled wrongly. From our analysis, the complex labels were more easily to lead to the errors propagation. For example, a wrong label "*B*" might incur a series of wrong labels. Additionally, we also noted that the boundary labels were difficult to tag than other labels. In our experimental data, many suspicious documents contained very short plagiarism passages, or even being composed by only one sentence. Many of these plagiarism passages were tagged wrongly, which resulted in the low *PlagDet* on some suspicious documents.

Noted that the performance improvement of text alignment on PAN12-artificial-high-obfuscation subcorpus was more significant than on PAN12-simulated-paraphrase subcorpus. We analyzed that the different plagiarism types affected the text alignment performance. For TA-CRF, we first obtained a sequence of matching sentence pairs. Yet, this sequence in high obfuscation plagiarism could not be obtained easily than those in the other plagiarism type since the former was always obfuscated by paraphrasing and summarizing to change most of its appearance, such as replacing the words with synonyms/antonyms and inserting some short phrases into the original sentence.

The influence of inaccurate matches in artificial-high-obfuscation subcorpus was obvious than that in simulated paraphrase subcorpus. In this circumstance, considering the context and the tagged labels could gain greater profits. However, in the simulated-paraphrase plagiarism, there were no apparent differences between terms. The matching sentences to be tagged were identified relatively accurately than in artificial-high-obfuscation subcorpus. Thus, the improvement of the performance on PAN12-simulated-paraphrase subcorpus was not higher than on PAN12-artificial-high-obfuscation subcorpus.

Furthermore, we also compared the TA-CRF with classification model on PAN12-artificial-high-obfuscation subcorpus. We chose SVM [35], one of the classical classification-based model, as the baseline. We trained a classification model on training corpus using SVM^{light} [1] with all the other parameters setting to their default values except the parameter *c* (the trade-off between training error and margin). The SVM method followed the same set of TA-CRF: the same input sequence, the same feature as described in 3.3.3 and the same post-processing methods. The experimental results were showed in **Table 6**. We chose Sanchez-Perez14, the baseline that achieved the highest *PlagDet*, and TA-CRF-LOF-L1, the proposed model that got the highest *PlagDet*, for comparing.

Table 6. Performance comparison of the TA-CRF and SVM on PAN12 Testing Corpus
PAN12-artificial-high-obfuscation subcorpus

		PlagDet	Precision	Recall	Granularity
Baselines	Sanchez-Perez14	0.4017	0.8404	0.3021	1.1532
Classification-based	SVM	0.3064	0.5016	0.2650	1.1914
Our methods	TA-CRF-LOF-L1	**0.5536**	0.7753	**0.4334**	**1.0061**

The experimental results of SVM verified our predictions in 3.2: compared with the methods of sequence labeling, the classification-based methods did not deal with the high obfuscation plagiarism well. This is primarily due to the fact that the context information has not been considered in classification-based methods. With the inaccurate matches, the classification-based method has not got better performance in artificial-high-obfuscation subcorpus.

5. Conclusion

In this paper, we propose to model text alignment in plagiarism detection as sequence labeling. Based on conditional random field, a classical sequence labeling model, we show that the method based on sequence labeling encompasses the different heuristic-based methods and the classification-based methods. Furthermore, it allows for the use of more flexible observation features extracted from a whole neighborhood of each matching text. In particular, we introduce six different models based on CRF using different potential functions and labels. We evaluate these models on a public corpus of text alignment in plagiarism detection. Although we use only simple lexical features in our TA-CRF, the experiments show that a significant improvement on the *PlagDet*, the primary evaluation metric, can be achieved for all the proposed six TA-CRF models by considering the sophisticated dependency structures of features. Furthermore, the TA-CRF framework also allows for designing other forms of potential functions, especially for designing the transition functions that permit the use of

[1] http://www.cs.cornell.edu/People/tj/svm_light

features to express the structural relations between observations in different positions, which will further improve the performance of TA-CRF in future.

References

[1] M. Potthast, T. Gollub, M. Hagen, J. Graβegger, J. Kiesel, M. Michel, A. Oberländer, M. Tippmann, A. Barrón-Cedeño, P. Gupta, P. Rosso, and B. Stein, "Overview of the 4th international competition on plagiarism detection," in *Proc. of 2012 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2012 Evaluation Labs, CEUR Workshop Proceedings*, pp. 101–128, 2012.

[2] M. Potthast, M. Hagen, T.Gollub, M. Tippmann, J. Kiesel, P. Rosso, E. Stamatatos, and B. Stein, "Overview of the 5th international competition on plagiarism detection," in *Proc. of 2013 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2013 Evaluation Labs, CEUR Workshop Proceedings*, pp. 301-331, 2013.

[3] M. Potthast, M. Hagen, A. Beyer, M. Busse, M. Tippmann, P. Rosso, and B. Stein, "Overview of the 6th international competition on plagiarism detection," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, pp. 845–876, 2014.

[4] L. Kong, H. Qi, S. Wang, C. Du, S. Wang, Y. Han, "Approaches for candidate document retrieval and detailed comparison of plagiarism detection - Notebook for PAN at CLEF 2012," in *Proc. of 2012 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2012 Evaluation Labs, CEUR Workshop Proceedings*, 2012.

[5] G. Oberreuter, D. Carrillo-Cisneros, I. D. Scherson, J. D.Velásquez, "Submission to the 6th international competition on plagiarism detection," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, 2014.

[6] J. D. Velásquez, Y. Covacevich Y, F. Molina F, E. Marrese-Taylora, C. Rodrígueza, F. Bravo-Marquez, "DOCODE 3.0 (DOcument COpy DEtector): A system for plagiarism detection by applying an information fusion process from multiple documental data sources," *Information Fusion*, vol. 27, pp. 64-75, 2016. Article(CrossRef Link)

[7] R. Yerra, Y.K., Ng, "A sentence-based copy detection approach for web documents," in *Proc. of FSKD'05 Proceedings of the Second international conference on Fuzzy Systems and Knowledge Discovery*, pp. 557-570, 2005. Article(CrossRef Link)

[8] C. Basile, D. Benedetto, E. Caglioti, G. Cristadoro, M. D. Esposti, "A plagiarism detection procedure in three steps: selection, matches and 'squares'," in *Proc. of SEPLN*, pp. 19-23, 2009.

[9] D.A.R. Torrejón, J. Manuel, M. Ramos, "Text alignment module in CoReMo 2.1 plagiarism detector notebook for PAN at CLEF 2013," in *Proc. of 2013 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2013 Evaluation Labs, CEUR Workshop Proceedings*, 2013.

[10] J. Kasprzak, M. Brandejs, M. Kripac, "Finding plagiarism by evaluating document similarities," in *Proc. of SEPLN*, pp. 24-28, 2009.

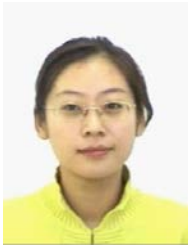
[11] J. Koberstein, Y. K. Ng, "Using word clusters to detect similar web documents," in *Proc. of International Conference on Knowledge Science, Engineering and Management*, pp. 215-228, 2006. Article(CrossRef Link)

[12] N. Meuschke, V. Stange, M. Schubotz, B. Gipp, "HyPlag: A hybrid approach to academic plagiarism detection," in *Proc. of SIGIR'18: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 1321-1324, 2018. Article(CrossRef Link)

[13] L. Kong, H. Qi, C. Du, M. Wang, Z. Han, "Approaches for source retrieval and text alignment of plagiarism detection," in *Proc. of 2013 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2013 Evaluation Labs, CEUR Workshop Proceedings*, pp. 301-331, 2013.

[14] M. Momtaz, K. Bijari, M. Salehi, H. Veisi, "Graph-based approach to text alignment for plagiarism detection in Persian documents," in *Proc. of FIRE* (*Working Notes*), pp. 176-179, 2016.
[15] M.A. Sanchez-Perez, G. Sidorov, A.F. Gelbukh, "The winning approach to text alignment for text reuse detection at PAN 2014," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings,* pp. 1004-1011, 2014.
[16] M.A. Sanchez-Perez, A. Gelbukh, G. Sidorov, H. Gómez-Adorno, "Plagiarism detection with genetic-based parameter tuning," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 01, pp. 1860006, 2018. Article(CrossRef Link)
[17] E. Stamatatos, "Plagiarism detection using stopword *n*-grams," *Journal of the Association for Information Science and Technology*, vol. 62, no. 12, pp. 2512-2527, 2011. Article(CrossRef Link)
[18] A. Daud, J.A. Khan, J.A. Nasir, A.A. Rabeeh, R.A. Naif, S. Jalal, N.R. Alowibdi, "Latent dirichlet allocation and POS tags based method for external plagiarism detection: LDA and POS tags based plagiarism detection," *International Journal on Semantic Web and Information Systems*, vol. 14, no. 3, pp. 53-69, 2018. Article(CrossRef Link)
[19] Š. Suchomel, M. Brandejs, "Improving synoptic querying for source retrieval," in *Proc. of Working Notes of the 6th International Conference of the {CLEF} Initiative . Toulouse, France: CEUR*, pp. 1-8, 2015. Article(CrossRef Link)
[20] F. Alvi, M. Stevenson, P. Clough, "Hashing and merging heuristics for text reuse detection - Notebook for PAN at CLEF 2014," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, pp. 939–946, 2014.
[21] L. Gillam, S. Notley, "Evaluating robustness for 'IPCRESS': Surrey's text alignment for plagiarism detection - Notebook for PAN at CLEF 2014," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, pp. 951–957, 2014.
[22] D. S. Glinos, "A hybrid architecture for plagiarism detection," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, pp. 958-965, 2014.
[23] S. Abnar, M. Dehghani, H. Zamani, A. Shakery, "Expanded N-Grams for semantic text alignment - Notebook for PAN at CLEF 2014," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, 2014.
[24] Y. Palkovskii, A. Belov, "Developing high-resolution universal multi-type N-Gram plagiarism detector - Notebook for PAN at CLEF 2014," in *Proc. of 2014 Cross Language Evaluation Forum Conference, Working Notes Papers of the CLEF 2014 Evaluation Labs, CEUR Workshop Proceedings*, 2014.
[25] A. Si, V.L. Hong, R.W.H. Lau, "CHECK: a document plagiarism detection system," in *Proc of the 1997 ACM symposium on applied computing*, pp. 70-77, 1997. Article(CrossRef Link)
[26] H. Zhang, T.W.S. Chow, "A coarse-to-fine framework to efficiently thwart plagiarism," *Pattern Recognition*, vol. 44, no. 2, pp. 471-487, 2011. Article(CrossRef Link)
[27] C. Sutton, A. McCallum, "An introduction to conditional random fields," *Foundations and Trend s in Machine Learning*, vol. 4, no. 4, pp. 267-373, 2012. Article(CrossRef Link)
[28] H. Li, "Statistical learning method," Tsinghua University press, pp. 195, 2012.
[29] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Readings in Speech Recognition,* vol. 77, no.2, pp. 267-296, 1989. Article(CrossRef Link)
[30] A. McCallum, K. Bellare, F. Pereira, "A conditional random field for discriminatively-trained finite-state string edit distance," in *Proc. of the Conference on Uncertainty in Artificial Intelligence*, 2005. Article(CrossRef Link)
[31] K. Fukunaga, "Introduction to statistical pattern recognition," *Academic press*, 1990. Article(CrossRef Link)

[32] J. Lafferty, A. McCallum, F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. of the 18th International Conference on Machine Learning*, pp. 282-289, 2001.
[33] C. Joder, S. Essid, G. Richard, "A conditional random field framework for robust and scalable audio-to-score matching," *IEEE Transactions on Audio Speech & Language Processing*, vol. 19, no.8, pp. 2385-2397, 2011. Article(CrossRef Link)
[34] T. Gollub, S. Burrows, B. Stein, "First experiences with tira for reproducible evaluation in information retrieval," in *Proc. of SIGIR*, pp. 52-55, 2012.
[35] C. Cortes, V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995. Article(CrossRef Link).



LeiLei Kong was born in 1979. She received the Ph.D. degree in information and communication engineering from Harbin Engineering University in 2017. She is now an associate professor
of Heilongjiang Institute of Technology. Her main research interests include intelligent information processing, information retrieval and
natural language processing.

ZhongYuan Han was born in 1977. He received the Ph.D. degree in computer science and technology from Harbin Institute of Technology in 2016. He is now a professor of Heilongjiang Institute of Technology. His main research interests include information retrieval, information filtering and data mining.

HaoLiang Qi was born in 1972. He received the Ph.D. degree in computer science and technology from Harbin Institute of Technology in 2007. He is now a professor of Foshan University. His research interests include natural language processing, information retrieval and information filtering.