

Graph transformer for cross-lingual plagiarism detection

Oumaima Hourrane, El Habib Benlahmar

Laboratory Information Technology and Modeling, Faculty of Sciences Ben Msik, Hassan II University of Casablanca, Casablanca, Morocco

Article Info

Article history:

Received Sep 8, 2021

Revised Apr 21, 2022

Accepted May 20, 2022

Keywords:

Cross-lingual plagiarism

Graph neural network

Graph transformer

Knowledge graphs

ABSTRACT

The existence of vast amounts of multilingual textual data on the internet leads to cross-lingual plagiarism which becomes a serious issue in different fields such as education, science, and literature. Current cross-lingual plagiarism detection approaches usually employ syntactic and lexical properties, external machine translation systems, or finding similarities within a multilingual set of text documents. However, most of these methods are conceived for literal plagiarism such as copy and paste, and their performance is diminished when handling complex cases of plagiarism including paraphrasing. In this paper, we propose a new graph-based approach that represents text passages in different languages using knowledge graphs. We put forward a new graph structure modeling method based on the Transformer architecture that employs precise relation encoding and delivers a more efficient way for global graph representation. The mappings between the graphs are learned both in semi-supervised and unsupervised training mechanisms. The results of our experiments in Arabic-English, French-English, and Spanish-English plagiarism detection show that our graph transformer method surpasses the state-of-the-art cross-lingual plagiarism detection approaches with and without paraphrasing cases, and provides further insights on the use of knowledge graphs on a language-independent model.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Oumaima Hourrane

Laboratory Information Technology and Modeling, Faculty of Sciences Ben Msik, Hassan II University of Casablanca

Morocco

Email: oumaima.hourrane@gmail.com

1. INTRODUCTION

Plagiarism is the use of original text data without providing adequate references. This phenomenon is accentuated when the root of plagiarism is in a different language, which is known as cross-lingual plagiarism. Although some research works have been carried out on monolingual plagiarism analysis, to our awareness, cross-lingual plagiarism analysis is still an emerging natural language processing task that has been studied in the literature. The task can be described as follows: Given a suspect document in a certain language, we are interested in checking if it is plagiarized from one or a set of original documents written in other language.

Current cross-lingual plagiarism detection approaches usually employ syntactic and lexical properties, external machine translation (MT) systems, or computing similarities between multilingual documents. Yet, these methods are conceived for literal plagiarism such as copy and paste, and their performance is diminished when handling complex cases of plagiarism including paraphrasing. The literal plagiarism form “copy and paste” in theory, the most easily detected and identifiable textual similarity. Certainly, the detection of this form is similar to checking the identity between two texts. To ingeniously

carry out this analysis automatically, it is required to carry out a word-for-word comparison of texts. Since this process is far too time-consuming to be integrated into commercially oriented or online solutions, as is the case with most of the anti-plagiarism tools, alternative methods had to be produced, which is one of the goals of this present work.

Furthermore, it is an accepted fact that automatically detecting textual semantic similarities such as paraphrase does not amount to detecting a possibility of plagiarism. Plagiarism is copying or paraphrasing text without citing the original reference, but in the case of textual similarity, we cannot know if the texts are similar literally or semantically, and consequently to correlate this similarity with plagiarism. It will then be up to a human to identify whether or not any similarities detected count as plagiarism. Certainly, they can be resulting from coincidences or from properly cited references. In this work, we do not pass judgment or make any decisions; we only focus on finding similar passages between two texts.

We can describe plagiarism detection process as a system composed of two consecutive tasks [1]. The first task is the candidate source retrieval of suspicious documents to compare later, and the second task is the detailed comparison, which is finding alignments of similar passages of pairs of documents, between the suspect document being processed and each of the sources returned by the first task. This paper focuses only on the second task: the cross-lingual comparison between suspect texts and a fixed number of candidate source texts.

In this paper, we proposed cross-lingual graph transformer-based analysis (CL-GTA), an approach for cross-lingual plagiarism detection that aim to represent the whole context by using knowledge graphs simultaneously to broaden and connect the concepts in a textual document. For graph representation, we propose a new model called Graph-Transformer that depends completely on the multi-head attention mechanism [2]. The graph transformer enables direct representation of relations between any two nodes without considering their remoteness in the graph. At last, we evaluate our method and compare it against the state-of-the-art using a dataset composed of manually and automatically created paraphrases, we also evaluate the performance of the analysis using paraphrases only.

The rest of the paper is structured as follows. In Section 2 we cite the state-of-the-art methods in cross-lingual plagiarism detection. In Section 3 we describe the background on transformers and graph neural networks. Then we describe the knowledge graph creation and the graph transformer model for graph representation, and then we conclude the section with the general framework for cross-lingual plagiarism detection. We evaluate in section 4 our approach for Spanish–English, French–English, and Arabic–English corpora, and comparing our results with various state-of-the-art approaches. We also show the results of detecting only paraphrases.

2. RELATED WORK

This section reviews the methods of cross-language similarity computing that have been employed for cross-Lingual plagiarism detection. An effectual algorithm for cross-languages with lexical and syntactic similarities is the cross-language character n-gram (CL-CnG) [3]. It is basically similar to some other monolingual plagiarism detection models [4], [5]. This model is syntax-based that employs character n-grams to model texts, namely, after text segmentation into 3-grams, the authors transformed it into tf-idf matrices of character 3-grams, after that, they used a weighting mechanism and cosine similarity as a metric for similarity computing.

Various methods exist that use parallel corpora, which is called cross-language alignment-based similarity analysis (CL-ASA) [6], [7]. This type of analysis is usually based on a statistical Machine Translation system. It determines how a text passage is probably the translation of other text using a statistical bilingual dictionary – generated with parallel a corpus which contains translation pairs. To make text alignment, this method takes into account the translation probability distributions and the variances in size of parallel texts in distinct languages.

There are two other approaches employing concepts from knowledge graphs like in this paper, they are referred to as cross-language thesaurus-based similarity analysis (CL-TSA) models. The first approach is called MLPlag [8], where the authors used EuroWordNet ontology [9] that changes words into language-independent forms. They also presented two measures of similarity: Symmetric Similarity Measures (MLPlag SYM) which is derived in part from the traditional vector space model (VSM), the second measure is Asymmetric Similarity Measure (MLPlag ASYM) which is the opposite of the previous measure. other similar method employs a multilingual semantic graph to construct knowledge graphs that represent the context of documents [10].

Other cross language similarity analysis is the cross-language explicit semantic analysis (CL-ESA) [11]. It is built on the classic explicit semantic analysis (ESA) model. This approach models the semantic

meaning of a text by an embedding based on the vocabulary retrieved from Wikipedia, to find a document within a multilingual corpus.

One of the obvious ways to analyze the cross-language plagiarism is the Translation + Monolingual Analysis (T+MA). For example, in [12], the system is simply divided into two components. The MT system that translates suspicious documents into English, they employ the transformer framework for the MT. The second component is the source retrieval it receives as inputs to the translated suspicious document's n-grams and returns documents ids from the reference English collection. Finally, the system performs the comparison between translated suspicious documents and the sources. In other approach [13], OpenNMT Library [14] is used to train an MT model as an additional requirement to estimate the pairwise similarity between sentences. For the last model, they fine-tuned the Bidirectional Encoder Representations from Transformers (BERT) Multilingual model [15] for the sentence pair classification and put the linear layer for on top of the pooled output of BERT.

In recent years, more approaches based on word embedding have been proposed for the cross-lingual semantic similarity. Lo and Simard [16] uses BERT with a similarity metric for cross-lingual semantic textual similarity. The metric is based on a unified adequacy-oriented Machine Translation quality evaluation and estimation metric for multi-languages. Another approach [17] uses word embeddings for cross-lingual textual similarity detection instead of lexical dictionaries. They present syntactic weighting in the sentence embedding. By using the Multivac toolkit that includes word2vec, paragraph vector and bilingual distributed representation features, and then assigning weights to the Part Of Speech tag of each word in the sentence. Asghari *et al.* [18] used Continuous Bag of Words Model (CBOW) and skip-gram models, and employed an averaging approach to combine word embedding to create of sentence embeddings, which are then compared using Cosine Similarity metric between source and suspect documents. Finally, an approach called Language-Agnostic Sentence Representation (LASER) [13] provides a Bidirectional Long short-term memory (BiLSTM) encoder which was trained on 93 languages, so they obtain sentence embeddings from the encoder via max-pooling of the last layer outputs and applying cosine similarity on corresponding sentence embeddings of each sentence pair.

3. METHODOLOGY

3.1. Preliminaries

3.1.1. Transformer

Transformer [2] is a neural network model primarily employed for neural Machine Translation systems. It uses a self-attention mechanism for creating both the encoder and the decoder [19]–[21], that directly represents relationships between words in a sentence, regardless of their particular position. The encoder consists of multiple identical layers and sub-layers. The first layer is a multi-head self-attention mechanism, and the second layer is a position-wise fully connected feed-forward network. The multi-head attention puts together multiple dot-product attention layers that supports parallel running. Each dot-product attention layer takes a set of queries, keys, values (q, k, v) as inputs. After that, it calculates the dot products of the query with all keys and applies a Softmax function to get the weights on the values. By stacking the set of (q, k, v)s into matrices (Q, K, V), it accepts highly optimized matrix multiplications. More precisely, the outputs can be structured as a matrix:

$$Attention(Q, K, V) = softmax(QK^T / \sqrt{d})V$$

Where d is the dimension of k and q. By arranging k attention layers into the multi-head attention, the outputs of all attention heads are combined and projected to the original dimension of x, followed by feed-forward layers, residual connection, and layer normalization, The output matrix can be written as:

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_k)W^o,$$

$$head = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

Where W_i^Q, W_i^K, W_i^V , are the projection matrices of head i .

To clarify the whole procedure, we denote the mechanism presented previously as a single function denoted as $S_{att}(x, y_{1:m})$. Given an input sentence $x_{1:n}$, the self-attention encoder iteratively calculates the sentence representation by: $x_i^L = S_{att}(x_i^L, x_{1:n}^{L-1})$. Where L is the number of layers and $x_{1:n}^0$ is word embeddings. Thus, this representation can build a direct relation with other long-distance representations. In order to preserve the sequential order of words, the position encoding technique is proposed [2] to show the

position order to the model. Therefore, the input representation will be the concatenation of word embedding and position encoding.

3.1.2. Graph neural networks (GNNs)

Graph neural networks have gained attention in different domains, such as knowledge graphs, social networks, citation networks, and drug discovery. Graph neural networks build representations of entities and edges in graph data. Their key process lies in message-passing process between the entities, where each node gathers features from its neighbors to update its representation of the local graph structure around it. The message passing operation iteratively updates the hidden features Hv of a node v , by concatenating the hidden states of v 's neighboring entities and edges. In each layer, the following equation is applied:

$$h_v^k = \sigma \left(W_k \sum_{u \in N(v)} \frac{h_u^{k-1}}{|N(v)|} + B_k h_v^{k-1} \right) \text{ Where: } k = 1, \dots, k-1$$

The first part bellow of the equation is averaging all the neighbors of node v : $W_k \sum_{u \in N(v)} \frac{h_u^{k-1}}{|N(v)|}$, while the second part is the embedding layer of node v multiplied by a bias B_k that is a trainable weight matrix generally, this part is called a self-loop activation for node v and can be described as follow: $B_k h_v^{k-1}$. Then the non-linearity activation such as sigmoid function is performed on the two parts. After L operations of message-passing, the hidden states of the last layer K are used as the embeddings of the entities, and can be described as $z_v = h_v^K$.

3.1.3. Knowledge graph

A knowledge graph is a graph relating entities and concepts and can assist a machine to learn human common-sense. The core of our approach is to use a graph representation that allows an alignment across languages. To build knowledge graphs for this purpose we employ Extended Open Multilingual Wordnet [22] which offers a wider set of concepts in several languages to date We will present this semantic network in the next paragraph, then in Section 3.2.2, we introduce the steps needed to obtain our multilingual knowledge graphs of documents.

WordNet. Wordnet is a wide electronic lexical database for English [23], [24], with a hierarchical formation of concepts, where more specific concepts derive information from their neighbors, more general concepts. Nouns, verbs, adjectives, and adverbs are clustered into sets of synonyms denoted as synsets, each representing a discrete concept. Synsets are interrelated using conceptual lexical and semantic relations. Secondly, WordNet labels the semantic relations among words, whereas the groupings of words in a dictionary do not follow any specific pattern other than meaning similarity. As mentioned before, we use Extended Multilingual Wordnet with large wordnets over 26 languages and smaller ones for 57 languages. It is made by combining wordnets with open data from Wiktionary, and the Unicode Common Locale Data Repository.

3.2. Model architecture

3.2.1. Creating knowledge graphs

In this section, we present the steps to create the Knowledge Graphs. We build the knowledge graph by searching WordNet for paths connecting pairs of synsets in V . At first, we preprocess the text segment using tokenization, multi-word extraction, lemmatization, part-of-speech tagging (POS), and to obtain the list of tuples (lemma, tag). Next, we create an initially empty knowledge graph $G = (V, E)$, such that $V = E = \emptyset$. We populate the vertex set V with the set of all the synsets in WordNet which contain any $\langle lemma, tag \rangle$ tuple T in the text segment language L .

Finally, for each pair $\{v, v'\} \in V$ such that v and v' do not share any lexicalization in T , foreach path in WordNet $v \rightarrow v_1 \rightarrow \dots \rightarrow v_n \rightarrow v'$, we set: $V := V \cup \{v_1, \dots, v_n\}$ and $E := E \cup \{(v, v_1), \dots, (v_n, v')\}$. Consequently, we put all the path nodes and edges to graph G . The length of each path is limited to a maximum of three [25]. Finally, we obtain a knowledge graph that represents the semantic context of the text by populating the graph with intermediate relations and nodes.

3.2.2. Knowledge graph notation

We denote E and R as the set of entities and relations respectively. A triple is defined as (h, r, t) , where $h \in E$ is the head, $r \in R$ is the relation, and $t \in E$ is the tail of the triple. Let x_i represent the set of all triples that are true in a world, and x_i' represent the false ones. A knowledge graph is a subset of x_i .

3.2.3. Graph-transformer

After creating graph knowledge, the next step consists of representing the graphs by weighting all concepts (entities) and semantic relations. Current graph neural networks calculate the node representation using a function of the input node and all its the receptive field of adjacent neighborhoods, which leads to inefficient long-distance information exchange. Therefore, we propose a new mechanism as shown in Figure 1, and known as Graph Transformer which enables relation aware global communication.

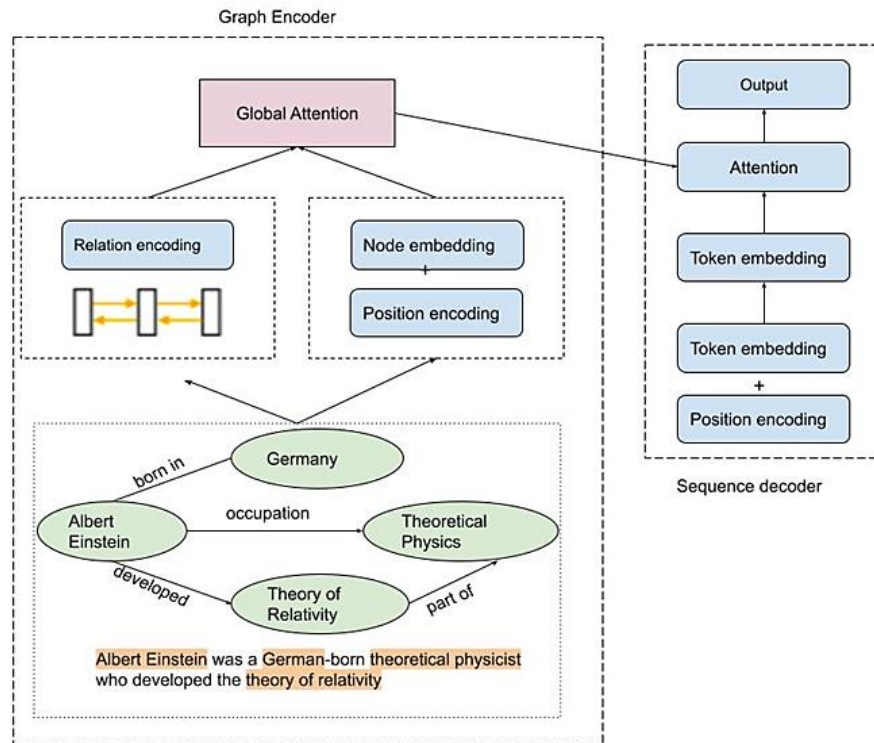


Figure 1. Graph-transformer architecture

Node representation. The most essential characteristic of this model is that it has a fully connected interface of random input graphs. Each node can directly send and get information to another node no matter if they are directly connected or not. This is achieved by the relation-enhanced global attention setting. In short, the relation between any node pair is presented as the shortest relation path between them. These paths between the two entities are used then as an input to relation encoding process, we denote the resulting learned vector as r_{ij} : the relation between node i and j . As the vanilla multi-head attention, we compute our attention as follow: $[r_{i \rightarrow j}; r_{j \rightarrow i}] = W_r r_{ij}$, where we divide the relation encoding into forward encoding $r_{i \rightarrow j}$ and backward encoding $r_{j \rightarrow i}$. The node vectors are initiated by the sum of the node embedding and position encoding. Multiple layers of the global attention are then combined to calculate the final node representation. For each layer, a node vector is updated based on all other node vectors and the corresponding relation encodings.

Relation encoding. In this work, to represent the relationship between two nodes we used the shortest path approach, because it usually offers the nearest and most crucial relationship between them. Based on the sequential characteristic of this relationship, we used a bi-directional Gated Recurrent Unit (GRU) [26] to get a probabilistic representation of it. We denote the shortest path between node i and node j as p_{ij} :

$$r_{i \rightarrow j} = GRU_f(r_{i \rightarrow j-1}, p_{ij})$$

$$r_{j \rightarrow i} = GRU_f(r_{j \rightarrow i-1}, p_{ij})$$

The final relation encoding is expressed as $[r_{i \rightarrow j} \cdot r_{j \rightarrow i}]$ which is the addition of the last hidden states of forward and backward GRUs.

Sequence decoder. After the graph encoding, we learn a mapping between two graphs: $G \rightarrow G'$, where $G = (\text{node}_1, \dots, \text{node}_n)$. This mapping is learned both in semi-supervised and unsupervised training mechanisms. We use the encoder-decoder mechanism to map the node vectors into low dimensional space. The encoder learns the node representation of the input sentence and the decoder employs this representation to rebuild in reverse order the sentence. The sequence decoder reflects the same process as the transformer decoder. We update the hidden state at each time step by computing a multi-head attention mechanism over the output of the encoder and the previously generated words. Finally, we minimize the error between input sentences and reconstructed output-sentence during the training as follow: $E_{\text{rec}} = \|s - \hat{s}\|^2$.

3.2.4. Cross-language plagiarism detection framework

We explain in this section in detail the framework for cross-lingual plagiarism detection. It is originally proposed by [10] as well as the post-processing analysis of similarities between text segments. As shown in Figure 2. Given a source document d_L in a language L and a suspicious document d_{L_0} in a language L_0 , we process documents in a four main step:

- (i) **Text segmentation.** We first segment the documents to be compared, to obtain the sets of segments S_L and S_{L_0} by using a sliding window of five sentences with a two sentences step to produce the segments.
- (ii) **Creating knowledge graphs.** Next, we implement the procedure presented in Section 3.1 to create the graph sets G and G_0 of the text segments S_L and S_{L_0} .
- (iii) **Graph representation.** It is the global graph representation as presented in Section 3.3.
- (iv) **Knowledge graphs similarity.** Find K nearest vectors by cosine similarity from source documents.
- (v) **Post-processing analysis of similarities.** After obtaining the set of the similarities between the text chunks of the source document d_L and suspected documents d_{L_0} , we use the method proposed by [27] to analyze the similarity scores and identify which segments of the suspected document are cases of plagiarism. Briefly, for each text chunk of d_L , we get the top five most similar chunks of document d_{L_0} . Then, iteratively we run the algorithm until convergence that aggregates the segments of P_G with a distance δ lower than a threshold $thres_1$. At last, we select as plagiarism the cases which combine more than $thres_2$ text segments. A function offsets offers the start and end offsets of the plagiarism case. We use this algorithm to evaluate all the models compared in the evaluation section.

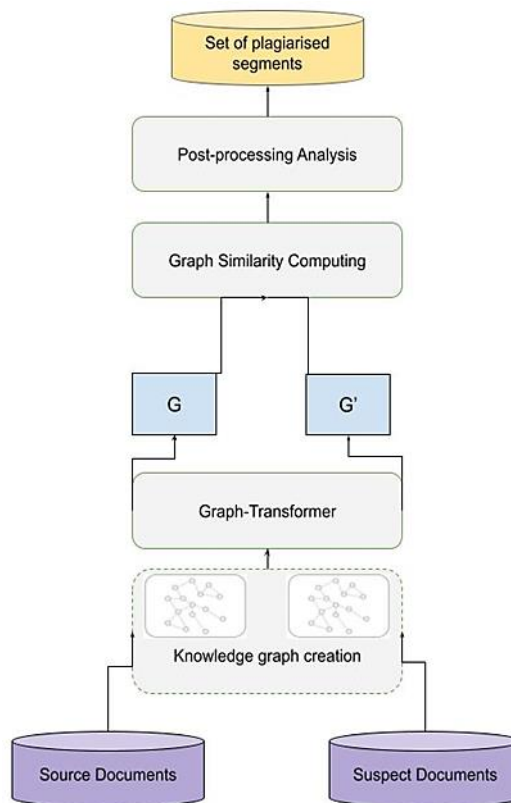


Figure 2. Cross-language plagiarism detection framework

4. EXPERIMENTS

We evaluate and compare our CL-GTA for plagiarism detection model with several state-of-the-art approaches in the task of cross-lingual plagiarism analysis. Given a collection of source documents DL0 in a language L0 and a suspect document dL in a language L, we would like to identify all the plagiarized segments of dL from the source documents DL0. We used as an evaluation metric the scores: precision, recall, granularity, and plagdet [28].

4.1. Evaluation metrics

We used as an evaluation metric the scores: precision, recall, granularity, and plagdet [28]. We denote S as the set of plagiarism cases in the suspect documents and R as the set of plagiarized sequences. The characters for a plagiarized case are denoted as $s \in S$. Likewise, the characters for a plagiarized text are represented as $r \in R$. Following these notations, and we measure the precision and the recall at the character level of R under S as follow:

- **The precision** represents the fraction of fragments found which cases of plagiarism are really. It measures the number of characters correctly returned as plagiarized on the total number of characters returned. The precision can be expressed as follow: $Precision(S, R) = \frac{1}{|R|} \sum_{r \in R} \frac{|U_{s \in S} s \cap r|}{|r|}$.
- **The recall** represents the fraction of plagiarized text that has been found. It measures the number of characters correctly returned as plagiarized over the number total number of characters to be returned as plagiarized. The recall can be expressed as follow: $Recall(S, R) = \frac{1}{|RS|} \sum_{s \in S} \frac{|U_{r \in R} s \cap r|}{|s|}$.
- **The F_{macro}** is an F-measure macro which takes into account the size of the plagiarized passages instead of only considering the absolute number of plagiarized passages. This is the harmonic mean between precision and recall. It can be expressed as follow: $F_{macro} = \frac{2 \cdot Precision(S, R) \cdot Recall(S, R)}{Precision(S, R) + Recall(S, R)}$

There is an issue that plagiarism detectors sometimes report overlapping or multiple detections for a single plagiarism case, and precision and recall do not perform well for that. We approach this problem by measuring the detector's granularity.

- **The granularity** is a measure first introduced in the work of Potthast et al. [28]. It determines whether a fragment is detected in whole or in pieces. This measure penalizes cases where passages, which are found, plagiarized, overlap. The granularity can be expressed as follow: $Granularity(S, R) = \frac{1}{|S_R|}$, where $S_R \in S$ are cases identified by detectors in R, and $R_S \in R$ are detections of S.
- **Plagdet (plagiarism detection)** is a measure combining precision and recall oriented for plagiarism detection and granularity. Plagdet is expressed as follow: $Plagdet(S, R) = \frac{F_{macro}}{\log_2(1 + granularity(S, R))}$.

4.2. Construction and properties of the corpus

In this work, our dataset consists of English, French, and Spanish documents. We decide to reuse already existing collections of parallel and comparable corpora in order to constitute a base for our corpus. These datasets are presented as follow:

- **Europarl** [29] is a corpus for cross-language and MT research. This corpus embodies about 10,000 parallel documents of the European Parliament exchanging transcriptions, in French, English and Spanish languages.
- **JRC-Acquis** [30] is usually used in cross-language and MT tasks. This corpus represents extracts of Acquis Communautaire. It consists of 10,000 parallel documents, available in French, English, and Spanish languages.
- **Wikipedia** is usually used as a parallel corpus of multiple languages. We chose to use 10,000 of French, English, and Spanish aligned documents. In total, this corpus contains 30,000 documents.
- **PAN 2011 corpus** has been used for the cross-lingual plagiarism detection competition of PAN at CLEF [27]. This corpus contains portions of writings of similar books in multiple languages. These texts are from books freely accessible on the Gutenberg Project website, available as Spanish–English (ES–EN) pairs.
- **Conference papers.** We used the processed conference papers corpus [31]. These are English–French conference papers that were first published in one language and then translated by their authors to be published in other language. A sum of 35 pairs of English–French conference papers was retrieved.
- **English–Arabic parallel corpora.** The corpus of the Arabic-English case was taken from different parallel corpora. It consists of 547 aligned passages from 58,911 pairs from the United Nations Parallel Corpora [32], the OPUS collection of translated texts from the web [33] and King Saud University corpus [34]. We used another corpus prepared by [35] which has roughly 2085 of paraphrased translated pairs which will be used when evaluating only paraphrasing cases.

4.3. Evaluation protocol

To precisely evaluate the methods of detecting cross-lingual plagiarism on our corpus, we present as follow, a dedicated evaluation protocol. We denote a parallel or comparable corpus as C , which is made up of N pairs of documents, such that for each document $d_i \in D$ a corresponding document $d_{0_i} \in D_0$ exists, where i is an integer between 1 and N , we decide to compare all the N documents available in D to M documents of D_0 . This has the advantage of avoiding making N^2 comparisons and thus having much too long computation times in the case of too large corpora. Each document d_i is compared to its corresponding document d_i and to $M-1$ other documents randomly selected with replacement in D_0 . The d_0 document can thus be set more than once. M is set at 1,000 documents agreeing with the state of the art [36].

The graph encoder and sequence decoder use randomly initialized node and word embeddings respectively. To prevent overfitting, we apply dropout with the drop rate of 0.2 [34]. We apply a special *UNK* token to replace with a rate of 0.33 the input nodes. We used Adam optimizer for parameter optimization with $\beta_{a_1} = 0.9$ and $\beta_{a_2} = 0.999$ [37]. We adopt the same learning rate of a standard transformer [2], and then we encode all shortest paths into vector representation by the relation encoding.

For a fair evaluation, we compared our CL-GTA model with the state-of-the-art Cross-Language Character N-Gram CL-C3G [3], Cross-Language Alignment-based Similarity Analysis CL-ASA [38], Cross-Language Explicit Semantic Analysis CL-ESA [11], and Cross-Language Knowledge Graph Analysis (CL-KGA) in [39] models. We also used the length model of [40] as a baseline.

5. RESULTS AND DISCUSSION

The results of our experiments were broken down in two folds: (i) we compared our model with the state-of-the-art approaches, assessing the performance when detecting the cross-lingual plagiarism cases of our corpora ES-EN, FR-EN, AR-EN; (ii) we examined the performance on solely the cross-lingual paraphrasing cases of plagiarism for the Spanish–English and Arabic–English partitions.

5.1. Comparison with the state of the art

5.1.1. Results

Table 1 shows the results obtained for Spanish–English (ES-EN), French–English (FR-EN), and Arabic–English (AR-EN) partition, For the Spanish–English (ES-EN). Partition, the CL-GTA approach has the best Plagdet score of 0.62, followed by the length Model with a score of 0.604, then the Cross-Language Conceptual Thesaurus-base Similarity CL-CTS model with a score of 0.584. The difference between the scores of all the other approaches is not huge, except for the CL-C3G with a score of 0.169. This is far lower than the State-of-the-art CL-GTA. For or the French–English (FR-EN), same as the Spanish–English cases, the CL-GTA reaches the best Plagdet score of 0.584, followed by the Length Model with a score of 0.553, then the CL-CTS with a score of 0.584. For the Arabic–English partition, the results prove a different outcome, with the CL-CTS in the first place with a Plagdet score of 0.534 followed by our model CL-GTA with a score of 0.522.

Table 1. Results of comparison with the state of the art

	Method	Length Model	CL-C3G	CL-ASA	CL-ESA	CL-CTS	CL-GTA
(1). Spanish–English	Plagdet	0.075	0.018	0.056	0.070	0.092	0.112
	Precision	0.149	0.048	0.153	0.160	0.186	0.203
	Recall	0.058	0.020	0.041	0.019	0.063	0.085
	Granularity	1.000	1.000	1.001	1.061	1.000	1.000
(2). French–English	Plagdet	0.553	0.065	0.405	0.395	0.504	0.584
	Precision	0.469	0.067	0.343	0.300	0.452	0.506
	Recall	0.683	0.306	0.029	0.356	0.633	0.690
	Granularity	1.007	1.099	1.103	1.112	1.017	1.000
(3). Arabic–English	Plagdet	0.520	0.192	0.690	0.303	0.534	0.522
	Precision	0.401	0.203	0.465	0.278	0.452	0.409
	Recall	0.598	0.025	0.758	0.423	0.633	0.576
	Granularity	1.010	1.082	1.203	1.105	1.007	1.101

5.1.2. Discussion

The results for French–English compared to Spanish–English were similar but with reduced performance. Spanish, French, and English do not share many grammatical characteristics. For all partitions, the CL-C3G got the lowest result, since syntactic and lexical features are important for high character n-gram overlap. After that comes CL-ESA, since it is based on computing similarities with document collections, the

model obtained a higher number of false positives. The CL-ASA is comparable with CL-ESA but with higher precision. The third best in ranking is the CL-CTS, which is also based on knowledge graphs, but with classical weighting for nodes. The length model offered higher performance compared to the state-of-the-art. However, our CL-GTA model obtained the best results, suggesting that the proposed model benefits from the explicit relation encoding which provides a more efficient way for global graph representation, leading to better results to measure cross-lingual similarity.

Regarding the Arabic–English case, we got a slightly different outcome with the CL-CTS better than our model CL-GTA, since the two approaches are based on graphs, the main difference is that the CL-CTS used different Knowledge graph sources and different techniques of graph representation. In this case, the classical weighting proves Superior for the Arabic language. However, by changing the weights and using a far richer Knowledge graphs construction, the CL-GTA can prove to be effective prospectively.

5.2. Cross-language plagiarism detection with paraphrasing

5.2.1. Results

Table 2 shows the results of the paraphrasing cases evaluation on the PAN competition (PAN-PC-11) corpus. Our model CL-GTA reaches the state-of-the-art on this task by a Plagdet score of 0.112, followed by the CL-CTS model with a score of 0.092; then the Length Model in the third place with a score of 0.075. We report as well the results of the paraphrasing cases of the Arabic–English paraphrased translated pairs. Same as the previous results, our method CL-GTA proved superior with a Plagdet score of 0.108, followed with the Length Model with a score of 0.105, then the CL-CTS model with a score of 0.099. The difference between the scores of all the other approaches is not big for both datasets, except for the CL-C3G with a score of 0.021. This is far again lower than the State-of-the-art CL-GTA.

Table 2. Results of the cross-language plagiarism detection with paraphrasing

	Method	Length Model	CL-C3G	CL-ASA	CL-ESA	CL-CTS	CL-GTA
(1). Spanish–English	Plagdet	0.075	0.018	0.056	0.070	0.092	0.112
	Precision	0.149	0.048	0.153	0.160	0.186	0.203
	Recall	0.058	0.020	0.041	0.019	0.063	0.085
	Granularity	1.000	1.000	1.001	1.061	1.000	1.000
(2). French–English	Plagdet	0.553	0.065	0.405	0.395	0.504	0.584
	Precision	0.469	0.067	0.343	0.300	0.452	0.506
	Recall	0.683	0.306	0.029	0.356	0.633	0.690
	Granularity	1.007	1.099	1.103	1.112	1.017	1.000
(3). Arabic–English	Plagdet	0.520	0.192	0.690	0.303	0.534	0.522
	Precision	0.401	0.203	0.465	0.278	0.452	0.409
	Recall	0.598	0.025	0.758	0.423	0.633	0.576
	Granularity	1.010	1.082	1.203	1.105	1.007	1.101

5.2.2. Discussion

As mentioned before, the PAN-PC-11 dataset contains cross-lingual paraphrasing cases of plagiarism, which is a more complex form of plagiarism to detect since it restates the text using other terms in order to conceal plagiarism. We conducted hereby a further experiment to examine only paraphrasing cases of plagiarism extracted from the corpus. We observe that the differences between the results of the models were identical to the previous results using the entire dataset at a smaller scale. CL-GTA obtained higher performance compared to the other baselines.

We conducted another experiment on the Arabic–English paraphrased translated partition. In contrary to the previous results on literal plagiarism case, our model overcame the CL-CTS model with a score of 0.108, which proves that in terms of semantic similarity, is better represented with our graph transformer architecture. This result is true even when representing the linear text sequence and this is due to the attention mechanism of the transformer, which allows a model to focus on the most relevant parts of the graph, thus representing the global graph dependencies in an efficient way. This is the main goal of this paper.

6. CONCLUSION

To conclude, in this paper, we have introduced a new approach for detecting cross-lingual semantic textual similarities based on knowledge graph representations and we have also augmented a state-of-the-art method by introducing these representations. We referred to our method as CL-GTA. We then introduced the notion of graph transformer, which is a new graph representation method based on the transformer architecture that employed explicit relation encoding and offers a more efficient way to represent global

graph dependencies. To build knowledge graphs, we used extended open multilingual wordnet since it provides a wide set of concepts and languages to date. We then constructed a knowledge graph that represents the semantic context of the text segment, by creating the graph with intermediate edges and vertices. The next step was to represent the graphs by weighting all concepts (entities) and semantic relations, by using our graph transformer based on the attention mechanism. The mappings between the graphs are learned both in semi-supervised and unsupervised training mechanisms. After the graph representation step, we explain more in detail the framework for cross-lingual plagiarism detection as well as the post-processing analysis of similarities between text segments. To measure the efficiency of our methods, we compare our CL-GTA for plagiarism detection model with multiple states-of-the-art approaches in the task of cross-lingual plagiarism detection. We used as evaluation metrics the scores: precision, recall, granularity, and plagdet. The experimental results show that the use of the graph transformer mechanism provided our model with state-of-the-art performance on the Spanish–English, French–English, and Arabic–English pairs. The experiments also demonstrated its advantage with cross-language paraphrasing cases for the Spanish–English and Arabic–English pairs. For future work, we will further improve the model to reaches the state-of-the-art on the Arabic–English literal translated cases, we will as expand the experiment to cover more languages and continue exploring the use of our proposed graph transformer and multilingual Knowledge graphs for other cross-lingual similarity tasks such as multilingual text classification and cross-lingual information retrieval.




REFERENCES

- [1] K. Leilei, Q. Haoliang, W. Shuai, D. Cuixia, W. Suhong, and H. Yong, “Approaches for candidate document retrieval and detailed comparison of plagiarism detection,” 2012.
- [2] A. Vaswani *et al.*, “Attention is all you need,” 2017.
- [3] P. McNamee and J. Mayfield, “Character N-Gram Tokenization for European Language Text Retrieval,” *Information Retrieval*, vol. 7, no. 1/2, pp. 73–97, Jan. 2004, doi: 10.1023/b:inrt.0000009441.78971.be.
- [4] P. Clough, “Old and new challenges in automatic plagiarism detection,” *National Plagiarism Advisory Service*, 2003.
- [5] H. A. Maurer, F. Kappe, and B. Zaka, “Plagiarism-a survey,” *Journal of Universal Computer Science*, vol. 12, no. 8, pp. 1050–1084, 2006.
- [6] A. Barron-Cedeno, P. Rosso, D. Pinto, and A. Juan, “On cross-lingual plagiarism analysis using a statistical model,” 2008.
- [7] D. Pinto, J. Civera, A. Barrón-Cedeño, A. Juan, and P. Rosso, “A statistical approach to crosslingual natural language tasks,” *Journal of Algorithms*, vol. 64, no. 1, pp. 51–60, Jan. 2009, doi: 10.1016/j.jalgor.2009.02.005.
- [8] Z. Ceska, M. Toman, and K. Jezek, “Multilingual Plagiarism Detection,” in *Artificial Intelligence: Methodology, Systems, and Applications*, Springer Berlin Heidelberg, pp. 83–92.
- [9] C. Jacquin, E. Desmontils, and L. Monceaux, “French EuroWordNet Lexical Database Improvements,” in *Computational Linguistics and Intelligent Text Processing*, Springer Berlin Heidelberg, 2007, pp. 12–22.
- [10] M. Franco-Salvador, P. Gupta, and P. Rosso, “Cross-Language Plagiarism Detection Using a Multilingual Semantic Network,” in *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2013, pp. 710–713.
- [11] E. Gabrilovich and S. Markovitch, “Computing semantic relatedness using wikipedia-based explicit semantic analysis,” *IJCAI*, vol. 7, pp. 1606–1611, 2007.
- [12] O. Bakhteev, A. Ogaltsov, A. Khazov, K. Safin, and R. Kuznetsova, “Crosslang: the system of cross-lingual plagiarism detection,” *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, pp. 1–9, 2019.
- [13] D. Zubarev and I. Sochenkov, “Cross-language text alignment for plagiarism detection based on contextual and context-free models,” in *Papers from the Annual International Conference “Dialogue”*, 2019, pp. 809–820.
- [14] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush, “OpenNMT: Open-Source Toolkit for Neural Machine Translation,” 2017, doi: 10.18653/v1/p17-4012.
- [15] J. Devlin, C. Ming-Wei, K. Lee, and K. Toutanova, “(BERT): Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018, pp. 4171–4186.
- [16] C. Lo and M. Simard, “Fully Unsupervised Crosslingual Semantic Textual Similarity Metric Based on BERT for Identifying Parallel Data,” in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019, pp. 206–215, doi: 10.18653/v1/k19-1020.
- [17] J. Ferrero, L. Besacier, D. Schwab, and F. Agnès, “Using Word Embedding for Cross-Language Plagiarism Detection,” 2017, doi: 10.18653/v1/e17-2066.
- [18] H. Asghari, O. Fatemi, S. Mohtaj, H. Faili, and P. Rosso, “On the use of word embedding for cross language plagiarism detection,” *Intelligent Data Analysis*, vol. 23, no. 3, pp. 661–680, Apr. 2019, doi: 10.3233/ida-183985.
- [19] J. Cheng, L. Dong, and M. Lapata, “Long Short-Term Memory-Networks for Machine Reading,” 2016, doi: 10.18653/v1/d16-1053.
- [20] W. Kryściński, R. Paulus, C. Xiong, and R. Socher, “Improving Abstraction in Text Summarization,” 2018, doi: 10.18653/v1/d18-1207.
- [21] H. Chen *et al.*, “Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network,” *IEEE Transactions on Medical Imaging*, vol. 36, no. 12, pp. 2524–2535, Dec. 2017, doi: 10.1109/tmi.2017.2715284.
- [22] F. Bond and R. Foster, “Linking and extending an open multilingual wordnet,” in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 2013, pp. 1352–1362.
- [23] G. A. Miller, “Wordnet: a lexical database for english,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [24] C. Fellbaum, “A Semantic Network of English: The Mother of All WordNets,” in *EuroWordNet: A multilingual database with lexical semantic networks*, Springer Netherlands, 1998, pp. 137–148.
- [25] R. Navigli and S. P. Ponzetto, “Multilingual wsd with just a few lines of code: the babelnet api,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2012, pp. 67–72.




- [26] K. Cho *et al.*, “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation,” 2014, doi: 10.3115/v1/d14-1179.
- [27] P. Gupta, A. Barrón-Cedeño, and P. Rosso, “Cross-Language High Similarity Search Using a Conceptual Thesaurus,” in *Information Access Evaluation. Multilinguality, Multimodality, and Visual Analytics*, Springer Berlin Heidelberg, 2012, pp. 67–75.
- [28] M. Potthast, B. Stein, A. B. on-C. no, and P. Rosso, “An evaluation framework for plagiarism detection,” in *COLING 2010, 23rd International Conference on Computational Linguistics*, 2010, pp. 997–1005.
- [29] P. Koehn, “EuroParl: A parallel corpus for statistical machine translation,” in *Proceedings of Machine Translation Summit X: Papers*, 2005, pp. 79–86.
- [30] R. Steinberger *et al.*, “The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages,” 2006.
- [31] J. Ferrero, L. Besacier, D. Schwab, and F. Agnès, “Deep Investigation of Cross-Language Plagiarism Detection Methods,” 2017, doi: 10.18653/v1/w17-2502.
- [32] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen, “The united nations parallel corpus v1. 0,” *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 3530–3534, 2016.
- [33] J. Tiedemann, “Parallel data, tools and interfaces in opus,” *Lrec*, pp. 2214–2218, 2012.
- [34] K. Alotaibi, “The Relationship Between Self-Regulated Learning and Academic Achievement for a Sample of Community College Students at King Saud University,” *Education Journal*, vol. 6, no. 1, p. 28, 2017, doi: 10.11648/j.edu.20170601.14.
- [35] S. Alzahrani and H. Aljuaid, “Identifying cross-lingual plagiarism using rich semantic features and deep neural networks: A study on Arabic-English plagiarism cases,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1110–1123, Apr. 2022, doi: 10.1016/j.jksuci.2020.04.009.
- [36] A. Barrón-Cedeño, M. Potthast, P. Rosso, B. Stein, and A. Eiselt, “Corpus and evaluation measures for automatic plagiarism detection,” 2010.
- [37] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” 2015.
- [38] A. Barrón-Cedeño, P. Rosso, E. Agirre, and G. Labaka, “Plagiarism detection across distant language pairs,” in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, 2010, pp. 37–45.
- [39] M. Franco-Salvador, P. Rosso, and M. Montes-y-Gómez, “A systematic study of knowledge graph analysis for cross-language plagiarism detection,” *Information Processing & Management*, vol. 52, no. 4, pp. 550–570, Jul. 2016, doi: 10.1016/j.ipm.2015.12.004.
- [40] B. Pouliquen, R. Steinberger, and C. Ignat, “Automatic identification of document translations in large multilingual document collections,” *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP'03)*, pp. 401–408, 2006.

BIOGRAPHIES OF AUTHORS



Oumaima Hourrane    is a PhD candidate in the Department of Mathematics and Computer Science at Faculty of Science Ben M'Sik, Hassan II University, Casablanca, Morocco. She majored in Computer Engineering in her master study at the National School of Applied Science, Safi, Morocco in 2016. Her research interests include Machine Learning, Artificial Intelligence, Natural Language Processing, and Information Retrieval. She has published multiple research papers in national and international journals as well as conference proceedings. She can be contacted by email: oumaima.hourrane@gmail.com.



El Habib Benlahmar    is a Full Professor of Higher Education in the Department of Mathematics and Computer Science at Faculty of Science Ben M'Sik, Hassan II University, Casablanca, Morocco since 2008. He received his PhD in Computer Science from the National school For Computer Science (ENSIAS), Rabat, Morocco, in 2007. His research interests span Web semantic, Natural Language Processing, Information Retrieval, Mobile platforms, and Data Science. He is the author of over 70 research studies published in national and international journals, as well as conference proceedings and book chapters. He can be contacted by email: h.benlahmer@gmail.com.