IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 22, 2021, accepted December 16, 2021, date of publication December 23, 2021,
date of current version January 4, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3137830

A Simple Yet Robust Algorithm for Automatic Extraction of Parallel Sentences: A Case Study on Arabic-English Wikipedia Articles

MAHA JARALLAH ALTHOBAITI
Department of Computer Science, Taif University, Taif 21944, Saudi Arabia
e-mail: maha.j@tu.edu.sa

ABSTRACT Parallel corpora are vital components in several applications of Natural Language Processing (NLP), particularly in machine translation. In this paper, we present a novel method for automatically creating parallel sentences from comparable corpora. The method requires a bilingual dictionary as well as an adequate word-vectorisation method. We use Arabic and English Wikipedia as a comparable corpus to apply our proposed method and construct a parallel corpus between Arabic and English. The created Arabic-English corpus consists of 105,010 parallel sentences with a total number of 4.6M words. During our study, we compared two methods of word vectorisation, word embedding and term frequency-inverse document frequency, in terms of their usefulness in computing similarities between well-formed and syntactically ill-formed sentences. We also quantitatively and qualitatively examined the parallel corpus produced by our proposed method and compared it with other available Arabic-English parallel corpora counterparts: GlobalVoices, TED, and Wiki-OPUS. We explored the main advantages and shortcomings of these corpora when used for NLP applications, such as word semantic similarity identification and Neural Machine Translation (NMT). The word semantic similarity models trained on our parallel corpus outperformed models trained on other corpora in the task of English non-similar word identification. Our parallel corpus also proved competitive when building Arabic-English NMT systems, yielding results comparable to those of the automatically created Wiki-OPUS corpus and of the manually created TED corpus, while achieving results superior to the smaller GlobalVoices corpus.

INDEX TERMS Automatic creation of parallel corpus, automatic sentence alignment, deep learning, neural machine translation, transformer model, word embedding.

I. INTRODUCTION

A parallel text consists of an original text in one language placed alongside its translations into one or more languages. Parallel text alignment is the identification of the corresponding sentences in both halves of the parallel text [1]. The collection of parallel texts aligned at sentence level is usually called a parallel corpus. Parallel corpora are crucial resources for many Natural Language Processing (NLP) applications, such as machine translation and cross-lingual information retrieval [2]–[4]. Parallel corpora are not easily acquired and they are challenging to construct due to the time and effort required for creating parallel texts by the means of translation and then aligning the sentences. On the other hand, comparable corpus is built from non-sentence-aligned and untranslated bilingual documents that are topic-aligned such as bilingual news articles, bilingual blogs, and Wikipedia [1], [5]. The comparable corpora are largely available online. In order to construct a parallel corpus from comparable corpora, it is necessary to conduct sentence alignment where sentences in the source text are mapped to their corresponding units in the target text. There are several proposed algorithms that can be utilised for sentence alignments [6]–[13]. Moreover, many studies that have constructed parallel sentences from comparable corpora relied heavily on computational processes to perform sentence alignment, such as building

The associate editor coordinating the review of this manuscript and approving it for publication was Nazar Zaki.

statistical machine transaction systems [14] or implementing a binary classifier for parallel sentence identification trained on a seed parallel corpus [15].

Despite the large amount of online comparable data between Arabic and other languages, efforts to exploit the data to create parallel corpora are limited. Therefore, Arabic parallel corpora are still limited in terms of size, domains, paired languages, and covered topics. Moreover, the available Arabic-X parallel corpora, automatically derived from a comparable corpus (e.g. Wikipedia) are polluted with noisy data, such as repeated sentences and sentences written in a different language (e.g., the existence of English sentences in the Arabic monolingual part of the Arabic-English parallel corpus). This can be attributed to the fact that a general methodology has been adopted to collect parallel corpora from comparable data without performing proper language-dependent pre-processing steps to remove language-dependent problems. For example, in Arabic, diacritical marks are optional and can cause problems if not handled when automatically aligning sentences. In addition, many Arabic letters have more than one form and can be written interchangeably, such as Alif and yaa'. This creates the problem of data sparsity, especially for automatic sentence alignment methods based on lexical information.

In this paper, we present a novel approach to automatically create parallel corpora from comparable ones. The proposed method requires only a bilingual dictionary as well as a method for word vectorisation. We compared two word vectorisation methods, Term Frequency-Inverse Document Frequency (TFIDF) and word embeddings, to compute similarities between sentences that have syntactic errors. We utilised Wikipedia as a case study to apply our approach and create an Arabic-English parallel corpus. In addition to quantitative evaluation of the resulting parallel corpus, the corpus was also evaluated qualitatively by employing it to build word semantic similarity models and neural machine translators from Arabic to English and from English to Arabic. The performance of the systems built using our parallel corpus was then compared with the performance of other systems trained on other available Arabic-English corpora. The main contributions of this paper are as follows:

- We proposed a new method to automatically extract parallel sentences from comparable corpora based on a bilingual dictionary, producing a high-quality parallel corpus without noise, such as repeated or incorrect parallel sentences.
- We compared TFIDF and word embeddings used for vectorising the words and computing similarities between well-formed sentences and ill-formed sentences that resulted from dictionary-based translation.
- We built a parallel corpus based on our proposed method and used it to implement NLP applications, such as word semantic similarity identification and neural machine translation.
- We quantitatively and qualitatively examined the parallel corpus produced by our method and compared it with other available parallel corpora counterparts, exploring the main advantages and shortcomings of these corpora when they are used for NLP applications, such as machine translation.

The paper is organised as follows: Section 2 discusses the text similarity measures, automatic sentence alignment algorithms, and the available Arabic-English parallel corpora, their sources, and the methods used to build them. Wikipedia, a comparable corpus utilised in our study, is described in Section 3. A summary of our proposed method is explained in Section 4. The three components of the proposed method, namely, document alignment, dictionary-based translation, and automatic sentence alignment with the limitations of our method are explained in Sections 5, 6, and 7 respectively. Experimental settings are illustrated in Section 8, while Section 9 presents the analysis of the resulting parallel corpus and the quantitative comparison with other available corpora. In Section 10, we qualitatively evaluate our parallel corpus by employing it to develop NLP applications: word semantic similarity identification and neural machine translation, as well as performing cross-corpus evaluation with other available corpora. Finally, we present the conclusion of the proposed method and future work.

II. RELATED WORK

A. TEXT SIMILARITY

Text similarity plays an essential role in many NLP tasks such as word-sense disambiguation, automatic essay scoring, machine translation, information retrieval, and text summarisation [16]–[18]. Measuring the similarity between two text strings determines their lexical and semantic closeness. The lexical similarity involves word-for-word comparison and does not consider word order or the meaning of the words in context. The semantic similarity, on the other hand, takes context into consideration when determining the similarity between words based on information gained from large corpora or semantic networks such as WordNet [19], [20].

Practically, the purpose of text similarity measures is to determine the distance between two text strings. Some text similarity metrics measure the distance between two text strings for approximate string matching or comparison, such as Damerau-Levenshtein, Jaro–Winkler, and n-gram similarity algorithms [21]. The Damerau-Levenshtein distance between two strings is the minimum number of operations required to transform one string into the other. The operations include insertions, deletions, or substitutions of a single character, or transposition of two adjacent characters [22], [23]. The Jaro distance is the number and order of the common characters between two strings with typical spelling deviations taken into account. The Jaro–Winkler is an extension of Jaro distance; it uses a prefix scale p that gives more favourable ratings to strings that match from the beginning for a set prefix length l [24]–[26]. The n-gram algorithms compare the n-grams of characters or words in two strings, then the distance is computed by dividing the number of similar n-grams by the maximum number of n-grams. The n-gram is a consecutive sequence of n items from a given sample of text or speech [27], [28]. The Jaccard similarity is an example of a popular text similarity measure that determines the distance between two strings by dividing the number of shared terms by the number of unique terms in both strings [29].

Some similarity measures require text strings to be represented as vectors of features before the distance between these features is measured. There are several methods of text vectorisation by which numerical features that capture the semantics of the texts are calculated. Examples of such methods are Bag of Words (BoW) [30], TFIDF [31]–[33], Continuous BoW (CBOW) model and Skip-Gram model, and Pre-trained word embedding models [34]–[36].

The most popular similarity measures between vectors are Euclidean distance and cosine similarity. Euclidean distance is defined as the square root of the sum of squared differences between the corresponding elements of the two vectors. Cosine similarity measures the similarity between two vectors of an inner product space. The semantic relatedness between two text strings can be expressed using a cosine measure between their corresponding vectors. Therefore, cosine similarity is typically used with many corpus-based semantic similarity measures. Examples of semantic similarity measures are Explicit Semantic Analysis (ESA) and Pointwise Mutual Information - Information Retrieval (PMI-IR) [37], [38]. These measures determine the similarity between two text strings based on the information gained from large corpora. A popular corpus-based similarity measure is Latent Semantic Analysis (LSA) [39], which assumes words that are close in meaning occur in similar pieces of text. A matrix containing word counts per paragraph/document is constructed from a large corpus, and then the Singular Value Decomposition (SVD) is used to reduce the number of columns, while preserving the similarity structure among rows. The cosine of the angle between two vectors formed by any two rows (rows represent unique words) is computed to compare the similarity between these two words [39]. Some studies have used external resources, such as Wikipedia, to find semantic similarity for distinct words, leveraging a large number of topics covered by Wikipedia, as well as its huge size. The study in [40] used a custom fuzzy thesaurus for text similarity analysis on Wikipedia. A custom fuzzy thesaurus, which is a matrix of all distinct words and their semantic relationships, was constructed by calculating the distance correlation factors between two distinct words. Given two distinct words w_1 and w_2, their distance correlation factor was computed based on the sets of all occurrences of words w_1 and w_2 in a common document, the number of words between w_1 and w_2 for each occurrence, and the frequency of w_1 and w_2 in that document [40].

In our study, we used word embeddings and TFIDF as text vectorisation methods to capture the semantics of the text strings before utilising the cosine similarity measure to compare those strings. In our case, we have two sentences: a well-formed Arabic source sentence from an Arabic Wikipedia article and its corresponding ill-formed pseudo-Arabic sentence using a dictionary (i.e., word-by-word translation using an Arabic-English dictionary without handling word order or structure correctness). In our study, we compared word embeddings and TFIDF by using them with cosine similarity to compute the similarity between a well-formed sentence and its corresponding ill-formed sentence.

B. SENTENCE ALIGNMENT ALGORITHMS

Sentence alignment is the most important step in parallel data preparation. It can be defined as the process of finding the existing relationships between the sentences of two texts that are known to be mutual translations [11], [41], [42]. Manual sentence alignment is expensive and time-consuming, and therefore, automatic methods have been proposed to conduct sentence alignment. Most methods developed for automatic sentence alignment can be divided into three categories: length-based, lexical-based, and partial similarity (cognate) based [10], [11], [13], [42], [43]. Also, automatic sentence alignment can be a hybrid of the two aforementioned categories. Length-based approaches use length information (e.g., a long sentence in one language is likely to be translated into a long sentence in the other language) to estimate the probabilistic scores of the proposed sentence pairs. The most popular length-based approach is the one suggested by Gale and Church [8]. On the other hand, lexical-based algorithms for automatic sentence alignment usually make use of lexical information derived from dictionaries or translated parallel sentences [44]. This section presents the proposed algorithms and tools proposed for automatic sentence alignment.

The study of [6] uses both sentence length and lexical correspondences to derive the final alignment. The proposed method is composed of two passes. First, it aligns the corpus using a sentence-length-based model. Then, it uses alignments obtained from the first pass to train a word-translation model (IBM Translation Model 1) [45], eliminating rare words and low-probability translations to reduce the size of the model. Finally, it uses the probabilities computed by the sentence-length-based model to dramatically reduce the search space explored by a second model that is more accurate, but more expensive (the word-correspondence-based model). The study in [10] designed a hybrid algorithm, *Hunalign*, which combines dictionary and length-based methods. The first step is to translate each word in the source texts based on a dictionary. Then, the pseudo-target text that resulted from the word-for-word translation is compared against the actual target text on a sentence by sentence basis. The similarity score between two sentences is computed using token-based and length-based components. The token-based score is the number of shared words in the two sentences, normalised with the larger token count of the two sentences. The length-based component computes similarity based on the character counts of the original texts incremented by one, where the score is based on the ratio of longer to shorter. The Hunalign tool calculates the similarity score for every sentence pair around the diagonal of the alignment matrix. The Hunalign tool can be used for any two languages. In the case of the absence of a bilingual dictionary, the tool performs length-based alignment, then estimates the bilingual dictionary from the results and reiterates the process. However, it is not designed to handle corpora of over 20k sentences. The Bleualign method proposed by [9] is based on the automatic translation of source text, where dynamic programming is used to find the path that maximises the BLEU score between target text and translation of source text. In order to conduct automatic translation of source text, the method requires building an SMT system with Moses, in which the training set is aligned based on the sentence-length (number of characters). The probability score for each sentence pair is computed and dynamic programming is used to find the maximum likelihood alignment. The study of [13] presents Lingua-Align, a toolbox for automatic tree-to-tree alignment that uses a local discriminative classification approach for a sequential alignment procedure. The features are extracted from parallel treebanks. The toolbox supports the extraction of features, including contextual information, as well as the integration of external tools for word alignment. Yalign is another tool for extracting parallel sentences from comparable corpora [12]. The Yalign tool depends on two components: a sentence similarity metric and a sequence aligner. The similarity metric estimates how likely two sentences are to be a translation of each other. On the other hand, a sequence aligner produces an alignment from a set of sentences of two documents, which maximises the sum of the individual similarities (per sentence pair). It uses a variation of the Needleman-Wunch algorithm to find an optimal alignment between the sentences of two documents. The Needleman-Wunch algorithm is one of the first applications of dynamic programming to compare biological sequences and its purpose is to find all possible alignments with the highest score [46]. Needleman-Wunch cannot handle cross-matching alignments (i.e., sentences in the source text must be matched in the same order in the target text) or alignments from two sentences into a single one.

In our study, we propose a new method to align sentences from two documents. The method does not require building an automatic translation system or to provide a parsed text corpus, annotating at a syntactic or semantic level. Our proposed method requires only a bilingual dictionary and depends on length-based as well as position-based features of the sentences in the two documents (i.e., the source sentences in one document and their corresponding translated sentences in the other). The position-based method to align sentences provides a simple way to allow cross-matching. In addition, our method of sentence alignment relies on word vectorisation methods, wherein the words of the sentences are numerically represented, then, the sentence vectorisation is computed utilising its words vectorised values, as will be seen in detail below.

C. PARALLEL CORPORA

This paper focuses on Arabic-English parallel corpora as a case study for our proposed algorithm, and thus we highlight the available Arabic-English parallel corpora. A number of Arabic-English parallel corpora have been constructed from comparable corpora (e.g., Wikipedia and bilingual news articles), human-translated documents (United Nations parliamentary documents), and movie subtitles [47]–[51].

Wikipedia parallel sentences [48] that cover 20 languages and 36 bitexts, including the Arabic-English language pair, are publicly available on OPUS[1] [52]. They were automatically collected from Wikipedia, aligned, and filtered. Specifically, a specialised web crawler was used to automatically obtain topic-aligned documents in any language supported by Wikipedia. Then, a sentence aligner, namely the Hunalign tool [10], was used to align the sentences. To filter the aligned sentences, a translation engine was used to find correct alignments between two languages (*L1* and *L2*) by translating *L1* sentences into *L2* and comparing the *L2* translated sentences (i.e., sentences that were translated from L1 to L2) with *L2* source sentences. They trained their specialised translation system using parallel data from various domains in the OPUS project [52]. The training was conducted using the Moses open-source SMT toolkit [53]. The Arabic-English Wikipedia-OPUS (abbrev. Wiki-OPUS) corpus available at OPUS[2] contains 151,136 parallel sentences. However, the corpus contains a considerable number of repeated sentences, as well as noisy ones, such as short sentences containing only numbers and sentences not written in the correct language. In other words, a large number of English sentences exist in the Arabic monolingual part of the parallel corpus. Moreover, the aforementioned method for collecting the Wikipedia-OPUS corpus requires parallel corpora for initial training and also depends on an external sentence alignment tool. Therefore, the accuracy of the output of the method (i.e., parallel sentences) relies on the quality of the data used for initial training and the external tool for sentence alignment. GlobalVoices is a parallel corpus, containing news articles from the website Global Voices.[3] During the collection of the corpus, the sentence alignment was carried out with Gargantua, the fast unsupervised sentence aligner described in [7]. The corpus was compiled and provided by CASMACAT [51]. Thorough manual corrections were made to the corpus quarterly. The latest version includes data up to December 2018. The Arabic-English GlobalVoices 2018Q4 contains 64,986 parallel sentences.

The TED Talk Parallel Corpus [50], [52], [54] contains the original content published by the TED Conference website.[4] The majority of TED talks are in English, but they are available on the TED website together with subtitles in various languages, including Arabic. These subtitles are the translations by volunteers of English talks into other languages. The TED2020 corpus, publicly available on OPUS,[5]

[1] OPUS, the open source parallel corpus, is a collection of translated texts from free online data on the web.

[2] https://opus.nlpl.eu/Wikipedia.php

[3] https://globalvoices.org/

[4] https://www.ted.com/

[5] https://opus.nlpl.eu/TED2020.php

was crawled from the translated subtitles for about 4,000 TED talks. The Arabic-English TED2020 corpus includes 403,716 sentence pairs [55]. The subtitles of online educational videos for 225 languages were utilised to build a parallel corpus called the QCRI Educational Domain (QED) Corpus, formerly QCRI AMARA Corpus [47], [56]. The content of the corpus was collected from various educational sources such as Coursera, Khan Academy, and Udemy. The Arabic-English portion of the QED corpus, available at OPUS,[6] includes 500,898 sentences. Regarding human-translated documents, the United Nations (UN) Parallel Corpus v1.0 is composed of the official records and parliamentary documents of the UN that were manually translated between 1990 and 2014, covering the six official languages of the UN, including Arabic [49]. The Arabic-English part of the UN corpus is composed of 111,241 documents, totalling 18,539,207 sentences.

TABLE 1. Available Arabic-English parallel corpora.

Corpus	# Sentences	Source
GlobalVoices v2018Q4	64,986	News articles
Wiki-OPUS	151,136	Wikipedia
TED2020 v1	403,716	TED video subtitles (Technology, Entertainment, Design)
QED	500,898	Video subtitles (Education)
UN	18,539,207	Official records and parliamentary documents

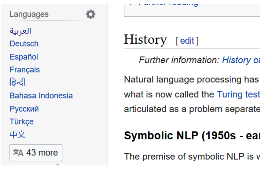
III. WIKIPEDIA

Wikipedia[7] is a free online encyclopedia available in a wide range of languages. It is collaboratively written by a large number of anonymous volunteers. Since its creation in 2001 up until the 3rd of June 2020, the total number of content pages has grown to 53,648,811 articles written in 315 languages [57].

A. WIKIPEDIA AS A COMPARABLE CORPUS

Each Wikipedia content article is uniquely identified by its title, which is a sequence of words separated by an underscore. The title is usually the most common name of the entity/topic explained. The description part is made up of the body of the text, which describes the entity/topic of the article. It usually contains hyperlinks to guide the reader to additional information about the concept denoted by the anchor text (aka link label). These links can direct the reader to other pages, as they indicate the entities' association with other Wikipedia pages [58], [59]. For example, in the English Wikipedia article describing "COVID-19", many hyperlinks are used to refer readers to various relevant entities such as "World Health Organization", "Wuhan", and "Centers for Disease Control and Prevention," that are featured in their own articles on English Wikipedia.

Wikipedia articles on the same topic in different languages are also hyperlinked via "inter-language" links, otherwise called "langlinks". For example, the English Wikipedia article on "Natural language processing" has links to equivalent articles on the same topic in 52 different languages, as shown in Figure 1. There is a link in the English Wikipedia article for "Natural language processing" to the equivalent article titled (*mçAljħ AllɣAt AlTbyçyħ*, "Natural language processing")[8] in the Arabic Wikipedia edition. This allows us to align the Wikipedia articles at the page (i.e., document) level [61]–[63].



Further information: History o

Natural language processing has
what is now called the Turing test
articulated as a problem separate

Symbolic NLP (1950s - ea

The premise of symbolic NLP is w

FIGURE 1. Links to Wikipedia pages for the same topic "Natural Language Processing" in different languages.

Wikipedia can be generally described as a mixture of noisy parallel and comparable corpora [64]. According to Fung and Cheung in their study [65], a noisy parallel corpus is one which contains non-aligned sentences that are nevertheless still roughly bilingual translations of the same document, with some deletions and insertions of paragraphs. The comparable corpus, on the other hand, is one that contains non-translated non-sentence-aligned bilingual documents, but are topic-aligned. Some topic-aligned Wikipedia article pairs are almost directly parallel, as authors translate articles from another language, while other topic-aligned article pairs contain no parallel sentences at all, because the authors write the content themselves. When extracting similar texts from Wikipedia article pairs, we leverage the fact that Wikipedia is a topic-aligned comparable corpus, as will be shown in the following sections.

IV. SUMMARY OF PROPOSED METHOD

Figure 2 shows the general framework of our proposed method for extracting parallel sentences from Wikipedia. The steps of this method can be summarised as follows:

1) Align articles that are on the same topic in Arabic (Ar) and English (En) languages.
2) Translate English sentences into Arabic using a bilingual dictionary (Pseudo-Arabic sentences).
3) Automatically align Arabic sentences in the actual Arabic article and the pseudo-Arabic sentences (i.e., the Arabic sentences obtained by translating the corresponding English article) depending on the following:

[6] https://opus.nlpl.eu/QED.php

[7] https://www.wikipedia.org/

[8] Throughout the entire paper, Arabic words are represented as follows: (*HSB transliteration [60]*, "English gloss").

- Adequate pre-processing (language-dependent cleaning, stop word removal, and normalisation),
- Sentence vectorisation (Term Frequency Inverse Document Frequency (TFIDF), word embedding vectors),
- Similarity measure,
- Sentence position and length.

The proposed method is straightforward and can be applied to any comparable corpus in order to extract parallel sentences. We employed Wikipedia in our paper, because it is a known free example of comparable corpus. In our study, the Arabic-English language pair was chosen as the use case. Our proposed method, however, can be easily applied to any language pair, as long as the appropriate bilingual dictionary is available. The only step of our proposed method that may differ based on the language pair is 'adequate pre-processing'. In morphologically rich languages, pre-processing steps are crucial to obtaining accurate results.

We utilise literal translation based on a bilingual dictionary (word-for-word translation) in order to simplify our method and to reduce the computational time, and to avoid building and training machine translation systems or using online systems. In addition, literal translations usually lead to syntactically ill-formed sentences (pseudo-Arabic sentences). Therefore, in our study we compared TFIDF and word embeddings to measure similarities between two types of sentences: well-formed sentences and ill-formed sentences. The TFIDF and word embeddings are used to vectorise each word in the sentence. Then, the final sentence vector is computed utilising the vectors of its content words. For translation, the Arabic-English dictionary, by the ArabEyes Team [66], is used in our study, because it covers a large number of topics. The dictionary was developed based on an Arabic word list for spell checking that contains 9 million Arabic words [66], [67] The following sections detail the previously listed steps for which we utilised English and Arabic editions of Wikipedia to build a high quality parallel corpus. We also describe the advantages and limitations of our proposed method.

V. DOCUMENT ALIGNMENT

The comparable corpus usually contains documents on the same topic in various languages. The first step is to prepare the comparable corpus by performing document-level alignment. In Wikipedia, the quality of document-alignment articles is very high, because inter-language links are used to connect articles about a specific topic in different languages. These inter-language links are annotated by Wikipedia's users. Furthermore, the conceptual mismatch between the linked articles is not common. Our proposed method relies on inter-language links to align the Wikipedia documents by topic. The Arabic article that does not have a corresponding English article or vice-versa will not be considered among the document-aligned articles. The document-aligned articles are used in the following steps in order to extract parallel sentences.

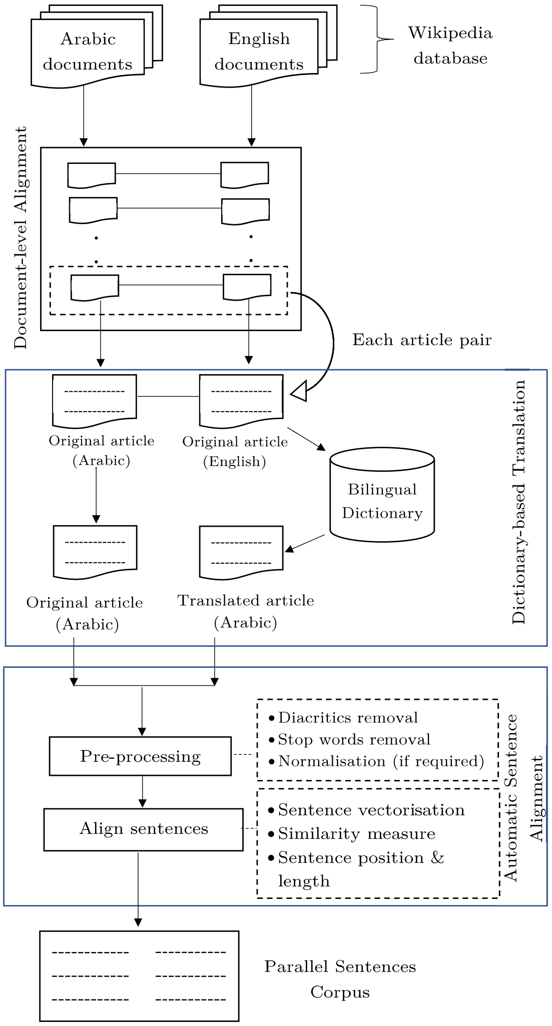

Arabic documents
English documents
Wikipedia database
Document-level Alignment
Each article pair
Original article (Arabic)
Original article (English)
Bilingual Dictionary
Original article (Arabic)
Translated article (Arabic)
Dictionary-based Translation
Pre-processing
• Diacritics removal
• Stop words removal
• Normalisation (if required)
Align sentences
• Sentence vectorisation
• Similarity measure
• Sentence position & length
Automatic Sentence Alignment
Parallel Sentences Corpus

FIGURE 2. Summary of proposed method.

VI. DICTIONARY-BASED TRANSLATION

With all document-aligned articles in place, the next step is to translate English articles into Arabic. We do not rely on any Machine Translation (MT) systems to perform the translation. Our method requires only a bilingual dictionary of terms in English and their translations in Arabic. The scope and the scale of the dictionary affect the quality of the translation. As Wikipedia is an open domain database, we devoted special attention to the dictionary in terms of the covered domains.

We used an Arabic-English dictionary that covers a large number of topics [66]. The dictionary was developed based on an Arabic word list for spell checking that contains nine million Arabic words [67]. As we used a dictionary, the process was carried out by translating each English word into Arabic separately (i.e., literal translation). The word-for-word translation does not consider how the words are used together in a sentence. Indeed, computing similarities between two sentences required for our proposed approach does not require grammar to be taken into account. In addition, we opted to translate from English to Arabic using a bilingual dictionary in order to avoid the complications generated by the opposite translation (i.e., from Arabic to English). That is, translating from Arabic to English necessitates finding rich morphological Arabic words in the dictionary, a challenging task owing to Arabic's complex clitics and affixes. This may require the use of light stemmers to reduce Arabic words to their stems in order to locate these stems in the dictionary and find their corresponding English words. One of the limitations of using a bilingual dictionary is that parallel Arabic-English sentences that contain idioms may not be translated properly by way of literal translation, which leads to a low similarity score between such Arabic and English sentences; therefore, these sentences will be excluded from the parallel corpus. This issue, however, does not affect the overall goal, since the precision of the collected parallel sentences is preferable over the recall (i.e., the ability to extract every possible pair of parallel sentences in the article pair). That is, our proposed algorithm is unsuitable if the goal is to extract every possible parallel sentence from the comparable corpus.

VII. AUTOMATIC SENTENCE ALIGNMENT

At this stage, we have topic-aligned article pairs, each of which contains an Arabic article and an Arabic translation of the corresponding English article. From now on, the Arabic translation of the English article will be referred to as "pseudo-Arabic article". The two main components of the automatic sentence alignment algorithm are shown in Figure 2. More details will be shown below.

A. PRE-PROCESSING

The topic-aligned article pairs (i.e., Arabic articles and Pseudo-Arabic articles) should pass through a pipeline of pre-processing steps. During pre-processing, the texts of the article pairs are cleaned in preparation to compute the similarities between the two text strings. In the case of Arabic, diacritical marks are removed, so the sentences can be used to measure similarities between sentences. Diacritical marks are optional since written Arabic words can be diacritised, partially diacritised, or entirely undiacritised. The next step is to remove stop words. A stop word is a commonly used word such as (*min*, "from") and (*Ǎlý*, "to"). The decision to remove stop words from sentences is to avoid computing the similarities between the sentences based on their common words. We used the list of 68 Arabic stop words provided by Khoja and Garside [68].

Furthermore, raw Arabic text is characterised by inconsistent variation that causes noise and data sparsity, making computing similarities and aligning sentences more challenging. For example, different forms of the letter *Alif* can be written interchangeably. Therefore, we carried out orthographic normalisation by (a) normalising different forms of *Alif* to a *bare Alif*, (b) normalising *tA' marbuta*ħ to *hA'*, and (c) normalising *Alif maqSwra*ħ to *dotted yA'*. We used the normaliser provided by the AraNLP[9] library [69].

B. SENTENCE ALIGNMENT

1) SENTENCE VECTORISATION

The first step of sentence alignment is to model each sentence in the vector space. We examined two different word modelling methods: (a) neural network-based word embedding models (e.g., Skip-Gram model) and (b) TFIDF. The TFIDF measure is a numerical statistic that reflects how important a term is to a document in a corpus. In our study, we use the TFIDF statistic to reflect how important a term is to a sentence in an article. So, each term is weighted by dividing the word frequency by the number of sentences in the article containing the word. The mathematical formula for TFIDF that we employed is as follows:

$$tfidf(t, s, S) = tf(t, s) * idf(t, S)$$

where t indicates the term; s indicates a sentence; and S is the sentence space and can be seen as $S = s_1, s_2, \ldots, s_n$ where n is the number of sentences in the document (i.e., Wikipedia article). The $tf(t, s)$ is simply to calculate the number of times the term t appears in the sentence s. Regarding the idf, the complete mathematical formula is as follows:

$$idf(t, S) = log\frac{|S|}{1 + |s \in S: t \in s|}$$

The numerator $|S|$ refers to the number of sentences in the document. The denominator $|s \in S: t \in s|$ indicates the number of times in which the term t appears in the sentence s given that the sentence s is in the sentence space S. In the case of using TFIDF to model sentences, normalisation is applied to both articles (i.e., Arabic articles and pseudo-Arabic articles) in order to reduce data sparsity and compute TFIDF correctly.

Word embedding models, on the other hand, are one of the popular representation methods for article vocabulary [34], [70]. Word embedding representation captures a large number of syntactic and semantic word relationships [34], [70], [71]. It maps the word into a vector by training a neural network on a large corpus. We used fastText[10] pre-trained word vectors for Arabic, which were trained on Wikipedia dump and obtained using the Skip-Gram model. The dimension of the vector is 300. The vector that represents each sentence (after pre-processing and removing stop words) is computed by averaging the sum of vector representations of all its words. Figure 3 illustrates the representation methods we used to map a sentence into a vector space. When using the word embedding model, we do not apply normalisation unless the word in the given sentence could not be found among the words in the pre-trained vectors. For example, when the letter Alif (Â) of the word (Âlɣý, "cancel") is written as a bare Alif (Alɣý, "cancel") in a given sentence in Arabic Wikipedia as seen in Figure 4, this may produce no results when searching for the word in the pre-trained word embedding vectors. No result arises because the letter

[9] AraNLP is available at https://github.com/Maha-J-Althobaiti/AraNLP

[10] An open-source and free library that allows users to learn text representations and text classifiers. Available at: https://fasttext.cc

Alif of the word (Âlγý, “cancel”) in the pre-trained model is defined as Alif with hamzah and is modelled into vector space according to this written form (Alif with hamzah). Accordingly, normalising the word by changing the bare Alif into Alif with hamzah is necessary to find matching words according to the pre-trained model. Therefore, in the case of the word embedding model, normalisation is only considered a necessary pre-processing step, if required. In addition, normalisation should not always be to change all forms of Alif into a bare Alif, but to also change from the bare Alif to Alif with hamzah or to other forms of Alif depending on how the words are defined in the pre-trained word vectors.

We map the words into vector space using different levels based on the utilised method. In the case of TFIDF, it is computed for each word at the document level (i.e., the TFIDF value of each word is computed based on its frequent appearance in all sentences in the Wikipedia article). The size of the vector depends on the size of the vocabulary in the document (i.e., Wikipedia article). In the case of word embedding, the word vectors are learned on the whole corpus. We utilise a pre-trained word embedding (i.e., fastText wiki word vectors) where each word is mapped into a vector based on its existence in the entire Wikipedia dump and based on the contexts in which it appears (see Figure 3).

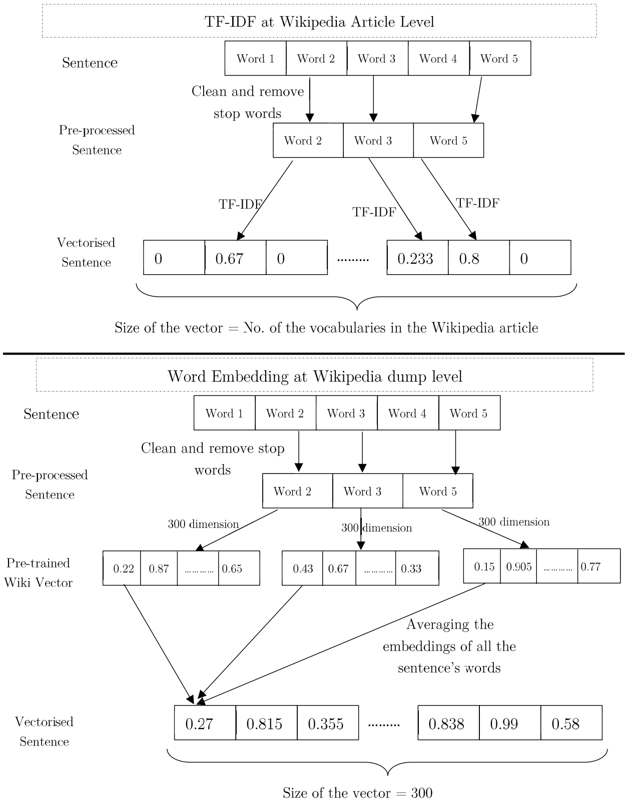
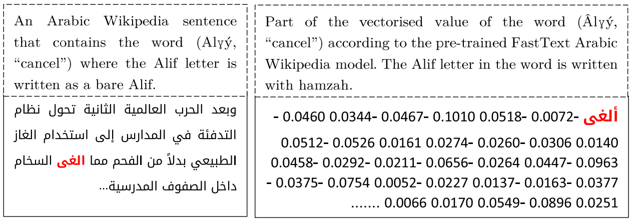


FIGURE 3. Sentence vectorisation methods.

2) SIMILARITY MEASURE

As a similarity measure, we use cosine similarity to calculate the similarity between sentences in each article pair: an Arabic article and its corresponding pseudo-Arabic article. The cosine similarity measures how similar two texts (e.g., documents or sentences) are in meaning, irrespective of their size. Mathematically, cosine similarity calculates the similarity between two vectors of an inner product space. It measures the cosine of the angle between the two vectors [72]. Given vectors of two sentences, a and b, f $\cos(\theta)$ is represented as a dot product and magnitude as follows:



An Arabic Wikipedia sentence that contains the word (Alγý, “cancel”) where the Alif letter is written as a bare Alif.	Part of the vectorised value of the word (Âlγý, “cancel”) according to the pre-trained FastText Arabic Wikipedia model. The Alif letter in the word is written with hamzah.
وبعد الحرب العالمية الثانية تحول نظام التدفئة في المدارس إلى استخدام الغاز الطبيعي بدلاً من الفحم مما الغى السخام داخل الصفوف المدرسية...	- 0.0460 0.0344- 0.0467- 0.1010 0.0518- 0.0072- ألغى 0.0512- 0.0526 0.0161 0.0274- 0.0260- 0.0306 0.0140 0.0458- 0.0292- 0.0211- 0.0656- 0.0264 0.0447- 0.0963 - 0.0375- 0.0754 0.0052- 0.0227 0.0137- 0.0163- 0.0377 .... 0.0066 0.0170 0.0549- 0.0896 0.0251

FIGURE 4. Normalisation and the pre-trained word vectors.

$$\cos(\theta) = \frac{a.b}{||a||||b||} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}}$$

The resulting cosine similarity ranges from -1 (i.e., the two sentences are totally opposite) to 1 (i.e., the two sentences are exactly the same) and the values that fall between denote intermediate similarity or dissimilarity. In our study, we use a *threshold* value to evaluate the cosine similarity score computed for bilingual sentence pairs. This means that the bilingual sentence pair will be added to the final parallel corpus only if it obtains a cosine similarity score greater than the *threshold* value.

3) SENTENCE POSITION AND LENGTH

In order to increase the accuracy of extracting parallel sentences, we take into account other factors, such as the position and length of the parallel sentences in the two articles. Indeed, one of the problems that faces the automatic alignment algorithm is the computational cost of calculating similarity between each sentence in the article of one language and all sentences in the article of the other language. It is necessary to calculate a similarity score for $(n_{L1} * n_{L2})$ sentence pairs where n_{L1} is the number of sentences in one language article while n_{L2} is the number of sentences in the other language article. In order to reduce the calculation cost, the similarity scores are calculated between each sentence at position (i) in one language article and the three sentences at positions (i−*1*), (i), and (i+*1*) in the other language article. We refer to this set of potential sentence pairs as the pool of *candidates*. The potential sentence pair that has the highest similarity score, which is also greater than the *threshold* value, is added to the parallel corpus and removed from the pool of *candidates*. This step is vital in order to avoid obtaining a parallel corpus with repeated sentences; repeated sentences occur when one English sentence can exist in the final corpus as a corresponding parallel sentence to more than one Arabic sentence.

Algorithm 1: Automatic Sentence Alignment

```
Input: Arabic Wikipedia articles $WikiAr = \{da_1, da_2, \ldots, da_n\}$ English
       Wikipedia articles, $WikiEn = \{de_1, de_2, \ldots, de_r\}$
Output: ParaC: Parallel corpus of Arabic-English
for $i \leftarrow 1$ to $n$ do
    $AlignedDocs = \{da_1\&de_1, da_2\&de_2, \ldots, da_m\&de_m\}$
for $i \leftarrow 1$ to $m$ do
    $docArPsuedo_i \leftarrow literalTranslate(de_i)$
    $SntEn \leftarrow$ splits $de_i$ into sentences
    $SntAr \leftarrow$ splits $da_i$ into sentences
    $SntArPsuedo \leftarrow$ splits $docArPsuedo_i$ into sentences
    pre-process($SntAr$ & $SntArPsuedo$)
    $l_1 \leftarrow size(SntAr)$
    $j \leftarrow 1$
    while $j \neq l_1$ do
        $l_2 \leftarrow size(SntArPsuedo)$
        $found \leftarrow False$
        $maxVal \leftarrow 0.0$
        for $k \leftarrow 1$ to $l_2$ do
            /* Sentence position and length */
            if $[(k = (j-1) ork = jork = (j+1))$ and $(length(SntAr_j) >$
              $length(SntEn_k)/2andlength(SntAr_j) < length(SntEn_k) * 2)]$
              then
                $removeDiacritics(SntAr_j, SntArPsuedo_k)$
                $removeStopWords(SntAr_j, SntArPsuedo_k)$
                $normalise(SntAr_j, SntArPsuedo_k)$
                $vectoriseSentnce(SntAr_j, SntArPsuedo_k)$
                $MetricsArr \leftarrow measureSimilarity(SntAr_j, SntArPsuedo_k)$
                $kArr \leftarrow k$
        $maxVal \leftarrow max(MetricsArr)$
        if $maxVal > threshold$ then
            $index \leftarrow$ kArr[index of maxVal in MetricsArr]
            $found \leftarrow True$
        if $found$ then
            add $SntAr_j$ & $SntEn_{index}$ to ParaC
            remove $SntEn_{index}$ from SntEn
            remove $SntArPsuedo_{index}$ from SntArPsuedo
            remove $SntAr_j$ from $SntAr$
            $l_1 \leftarrow l_1 - 1$
        else
            $j \leftarrow j + 1$
```

The proposed algorithm for automatic sentence alignment takes into account the sentence length. In fact, the sentence length has proved beneficial since the earliest research involving sentence alignment [6], [8], [10]. Unlike many previous studies, we do not rely on the number of characters when aligning sentences from two articles. The condition of sentence length in our algorithm is based on the number of tokens in each sentence. We observed that the difference in the number of tokens between parallel sentences is usually not a large value. Therefore, we set a sentence length condition in our algorithm requiring the number of tokens in the sentence of one language $|s\_tokens_{L1}|$ be greater than half the number of words in the corresponding sentence in the other language $|s\_tokens_{L2}|$, while simultaneously remaining less than the double that number (i.e., $|s\_tokens_{L2}|/2 < |s\_tokens_{L1}| < |s\_tokens_{L2}| * 2$). The main steps of our proposed automatic sentence alignment are explained in Algorithm 1.

The position-based condition we set reduces the computational cost of finding similarities between sentences. It also provides a way of flexibility in terms of handling alignments that cross each other (e.g., sentence three in source text may best match against sentence two in the target text). In addition, use of the threshold raises the bar for accuracy of the final resulted corpus. That is, potential parallel sentences are only added to the final corpus if their similarity scores are greater than the threshold value. One limitation of our algorithm is its preference of precision over recall, resulting in some but not all possible parallel sentences being added to the final corpus. For example, if one sentence in the source text exists at position 3 and its corresponding sentence in the target text exists in position 6 or 7, then the algorithm will not be able to extract them as parallel sentences because of their far away positions in the two texts. In this case, the similarity scores is only calculated between the sentence at position 3 in the source text and the three sentences at positions 2, 3, and 4 in the target text. Therefore, not all optimal parallel sentences will be extracted by our algorithm. In addition, our proposed algorithm cannot handle the two-into-one alignment whereby two sentences in a source text correspond to a single sentence in the target text or vice versa.

VIII. EXPERIMENTAL SETUP

In this paper, we use Arabic and English editions of Wikipedia that were released on 1 April 2020, which are available as a database dump. As of 1 April 2020, the English edition of Wikipedia had 6,046,435 articles and the Arabic edition contained 1,038,435 articles [57].

We set the threshold used to evaluate the cosine similarity score computed for bilingual sentence pairs to 0.5. This means that the bilingual sentence pair will be added to the parallel corpus if it obtains a cosine similarity score greater than 0.5. Indeed, this threshold value has been chosen based on a preliminary experiment on randomly selected data and a manual analysis of its results. We randomly selected 10 Wikipedia article pairs in which each document pair covers the same topic. We asked two human translators, fluent in Arabic and English, to manually align sentences in these articles. Two sentences are only considered aligned if both human translators agree that the two sentences are aligned. Table 2 shows the results of human sentence alignment. Out of 116 sentences in Arabic articles and 143 sentences in the corresponding English articles, 38 sentence pairs were aligned and considered parallel.

The same 10 document pairs were presented to our system, including our proposed algorithm for automatic sentence alignment. Firstly, we experimented with TFIDF as a method of vectorising the texts and with various threshold values for the cosine similarity score, ranging from 0.1 to 0.9. Our experiments revealed that when the threshold was equal to 0.1, the algorithm extracted a large number of parallel sentences (50 parallel sentences), but 76% of these sentence pairs were aligned in error. Increasing the threshold to 0.2 decreased the number of parallel sentences aligned by our algorithm to 27 parallel sentences. Nevertheless, the errors also decreased; approximately 55% of the parallel sentences were aligned mistakenly. Regarding the threshold 0.9, the algorithm failed to find any sentence pair that had cosine similarity greater than 0.9. The same result was achieved when

TABLE 2. Human sentence alignment.

	# Sentences		
	Arabic	English	Human Aligned
Doc 1	5	11	3
Doc 2	16	17	6
Doc 3	10	16	4
Doc 4	18	20	4
Doc 5	17	26	6
Doc 6	11	13	8
Doc 7	6	8	2
Doc 8	14	13	1
Doc 9	11	11	1
Doc 10	8	8	3
Total	116	143	38

the threshold was set to 0.8. These two values are very high, resulting in an empty parallel corpus. We discovered that setting the threshold to 0.5 resulted in 8 correctly aligned sentence pairs. Nevertheless, the value 0.5 as a threshold for the cosine similarity score led to low recall; only 8 sentence pairs out of 38 were permitted to align. Furthermore, increasing the threshold to 0.6 led to the correct alignment of only 3 sentence pairs. The accuracy was also as high as the threshold of 0.5 with no errors at all in aligning sentences. Secondly, the settings of our system were changed so that sentences were vectorised by pre-trained wiki word vectors provided by fastText. In general, our system's automatic sentence alignment performance was not solid, regardless of the similarity score value used as a threshold to accept or reject the parallel sentences extracted by the system. A similarity score value equal to or less than 0.7 leads to a low precision rate below 0.25. Increasing the similarity score to 0.9, however, leads to 100% correctly extracted parallel sentences. Nevertheless, the number of correctly extracted parallel sentences was very low with the recall reaching only 0.03; therefore, only one sentence was correctly extracted among 38 existing parallel sentences in all 10 document pairs. Figure 5 illustrates the performance of sentence vectorisation methods with various values of similarity scores.

The best results of our proposed system can be obtained when using TFIDF as a method to map a sentence into a vector space. The TFIDF score for each word in each sentence, as shown in Figure 3, represents how important the word is to the sentence in the given Wikipedia article. As previously stated, computing TFIDF for each word does not require taking the word's context into account. On the other hand, the word embedding model does not only take context into account, but it also assigns the word its numeric value based on the whole Wikipedia corpus, as we used pre-trained word vectors trained on the whole Wikipedia corpus using a neural network. Truly, our method aligns sentences by comparing each sentence in any Arabic Wikipedia article with other sentences in the corresponding English article after translating them into Arabic using a dictionary. This means the translation is literal (i.e., word-for-word translation) and overlooks the context and the syntactic information. In addition, we remove stop words from Arabic sentences and the Pseudo-Arabic sentences in the pre-processing step before aligning the sentences. This leads us to the inference that the inclusion of context in the form of word vectors using the word embedding model may negatively impact computation of the similarity of ill-formed sentences (in our case, stop words are removed and one of the sentences to be compared is a result of word-for-word translation). Another possible reason why TFIDF is better than word embedding models for automatic sentence alignment in our system is that in word embeddings, the sentence vector is calculated as an average of all word vectors in the sentence. This does not reflect the true numeric representations of the sentences, let alone the fact that the sentences are ill-formed based on our approach, as illustrated above. We decided to use the TFIDF method to vectorise the sentences in our approach. The threshold for the cosine similarity score to accept or reject the extracted parallel sentences was set to 0.5.

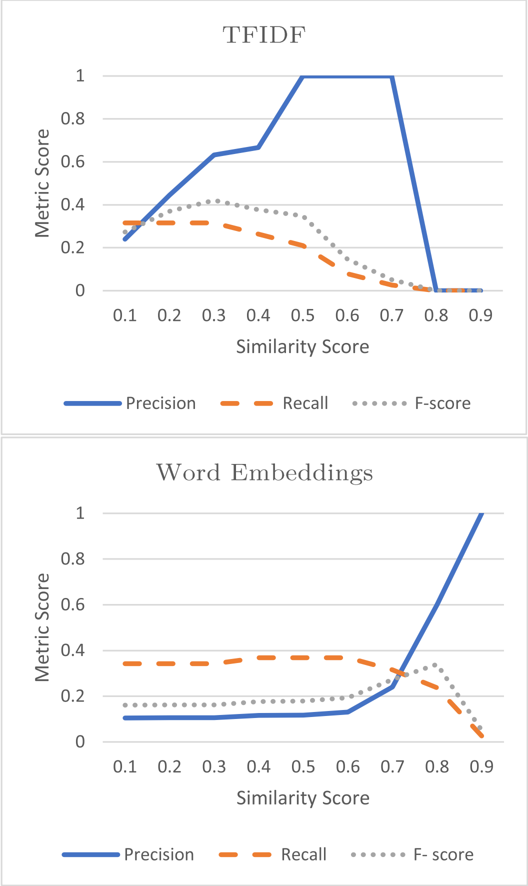


FIGURE 5. Comparison of the performance of sentence vectorisation methods with various values of similarity scores for automatic sentence alignment.

We only performed our experiments on the first three paragraphs of any Wikipedia article to limit the possible candidate pairs when computing similarity scores and aligning sentences. This decision was based on our observations that most parallel sentences between two Wikipedia article pairs (i.e., Wikipedia articles in English and Arabic on the same topic) appear in the first three paragraphs in each pair of articles. In addition, the TFIDF for each term, according to our proposed approach (see Section VII-B), will be computed at the Wikipedia article level. Since this computation costs more time, we decided to limit the search space to the first three paragraphs of each Wikipedia article pair. Our proposed system took 19 days to extract parallel sentences from the English and Arabic editions of Wikipedia dump. The PC used has a CPU @ 4.01 GHz, RAM with 32.0 GB and 952 GB of an SSD hard disk.

IX. ANALYSIS

A. OUR PARALLEL CORPUS STATISTICS

Using the aforementioned proposed method and settings, we were able to extract 105,010 parallel sentences. The number of words in the Arabic and English sentences is equal to 2,186,807 and 2,462,587 respectively. Figure 6 shows the number of vocabularies (i.e., distinct words) for each language and their ratio to the total number of words in the corpus. The average length of an Arabic sentence is 20.82 words, while the English sentences have 23.45 words on average. We plan to make our parallel corpus freely available to the research community.[11]

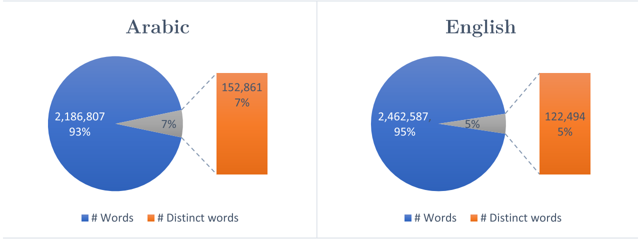


FIGURE 6. Number of words and vocabularies for Arabic and English sentences in the parallel corpus.

As previously mentioned, we only included the first three paragraphs of each Wikipedia article pair to extract sentences; although using the Wikipedia article pairs in their entirety would certainly increase the total number of extracted parallel sentences, but significantly more time would also be required.

B. QUANTITATIVE COMPARISON WITH AVAILABLE PARALLEL CORPORA

This section presents a quantitative comparison between our parallel corpus and other available parallel corpora for Arabic and English. We selected these parallel corpora based on their availability and domains. That is, we took care to diversify the sources from which these corpora were derived. These corpora are GlobalVoices v2018Q4 parallel corpus extracted from news articles, TED parallel corpus extracted from TED 2020 talk subtitles, and Wikipedia-OPUS parallel corpus extracted from the 2014 version of Arabic and English Wikipedia. More information about these corpora is expounded in Section II. The comparison factors include the number of words, the number of vocabularies, the number of sentences, average sentence lengths, and the ratio of repeated sentences in the corpus. Table 3 and Table 4 present the results of this comparison.

TABLE 3. Quantitative comparison between our parallel corpus and other available parallel corpora (words).

Parallel Corpus	Languages	# Words	# Distinct words
GlobalVoices v2018Q4	Arabic	1,116,506	177,055
	English	1,361,478	117,785
TED2020 v1	Arabic	5,653,050	452,384
	English	6,884,345	188,272
Wiki-OPUS	Arabic	2,581,508	371,647
	English	2,795,872	288,993
Our Parallel Corpus	Arabic	2,186,807	152,861
	English	2,462,587	122,494

TABLE 4. Quantitative comparison between our parallel corpus and other available parallel corpora (sentences).

Parallel Corpus	Languages	# Sentences	Average Length of Sentence	# Repeated Sentences
GlobalVoices v2018Q4	Arabic	64,986	17.18	1469 (2.26%)
	English	64,986	20.95	1508 (2.32%)
TED2020 v1	Arabic	403,716	14.00	6836 (1.69%)
	English	403,716	17.05	8943 (2.22%)
Wiki-OPUS	Arabic	151,136	17.08	5299 (3.50%)
	English	151,136	18.49	5415 (3.58%)
Our Data	Arabic	105,010	20.82	0 (0%)
	English	105,010	23.45	0 (0%)

As shown in Table 4, there are repeated sentences in the available Arabic-English parallel corpora. These repeated sentences constitute more than 3% of the total number of sentences in the Wiki-OPUS parallel corpus. In addition, there are a considerable number of sentences in the Arabic part of the Wiki-OPUS parallel corpus that are written in English rather than Arabic. For example, the sentence "The cave was pushed to −340 m during 1982-1987." exists in both the Arabic and English parts of the parallel corpus although we would expect to find the corresponding Arabic sentence in the Arabic part of the corpus. Our method efficiently produced a corpus with no repeated sentences.

X. APPLICATIONS

This section illustrates the qualitative evaluation we performed to examine the quality of our parallel corpus in comparison to other available corpora (i.e., TED, GlobalVoices, and Wiki-OPUS). We utilised our parallel corpus to build

[11] URL https://github.com/Maha-J-Althobaiti/Ara_Eng_Parallel_Corpus

NLP applications and assess its performances in comparison to other available corpora when utilised to train the same NLP applications. Firstly, we utilised our parallel corpus and other corpora to build word embedding models and test them to capture semantic similarity between words in both Arabic and English. Then, we built several Arabic-English Neural Machine Translation (NMT) systems trained on our parallel corpus and the other three corpora, and compared their performance. In the aforementioned NLP applications, as will be explained in details below, the results confirm that our parallel corpus performs on par with other available corpora in most implemented NLP applications and performs better than other corpora in the application of identifying the English non-similar words.

A. WORD SEMANTIC SIMILARITY

a: MOST SIMILAR WORDS IDENTIFICATION

To evaluate the quality of our Arabic-English parallel corpus, we trained a word embedding model (Skip-Gram model [70]) on the Arabic and English parts of our parallel corpus. Then, the trained models were examined to determine semantic similarity between words and capture the interrelatedness. Also, we evaluated our parallel corpus against other freely-available parallel corpora, namely, GlobalVoices, TED, and Wiki-OPUS. For each one of these corpora we built word embedding models for Arabic and English, leading to a total of six models for the three corpora. Again, we examined these models by utilising them to capture sematic relatedness between words.

b: NON-SIMILAR WORD IDENTIFICATION

The Arabic word embedding models trained on the aforementioned four corpora, including our parallel corpus, were also examined to perform the task of non-similar word identification. In this task, a list of words is given and the word embedding model finds the non-similar word in the list.

c: RESULTS OF ARABIC WORD SIMILARITY MODELS

Table 5 shows some examples of Arabic words and their top four nearest neighbours, according to the Arabic word embedding models for the four corpora. The word embedding model trained on the Arabic part of our parallel corpus successfully found the most similar words to any given words. Indeed, the model's performance is on par with those models trained on the other corpora (i.e., Wiki-OPUS, TED, and GlobalVoices), if not better. For example, the Arabic model trained on our corpus captures the top four nearest neighbours to the Arabic word (AlnAs, “people”) as follows: (AlÂfrAd “individuals”, AlÂšxAS “persons”, Albašar “humans”, and AlAxryn “others”). These four words are more similar to the given word (AlnAs, “people”) than the words produced by the model trained on Wiki-OPUS (yaçrfwn “they know”, and Ânahum “that they”, for example). Moreover, the results of the word embedding model trained on our corpus far exceed the word embedding model trained on the talk-based corpus, such as TED or a corpus collected from a news domain like GlobalVoices. The TED model considers the words (lÂnahum “because those”) and (yurydwn “they want”) among the four most similar words to the word (AlnAs, “people”), while the GlobalVoices model ranks the word (AlTryqah “method”) as the second most similar word to the word (AlnAs, “people”).

TABLE 5. Examples of Arabic words and their top four nearest neighbours according to the Arabic word embedding models for the four corpora; English translations are used for indicative purposes and written between double quotes, underlined words indicate wrong predictions (i.e., cannot be considered similar words, let alone being nearest neighbours).

	GlobalVoices	TED	Wiki-OPUS	Our Parallel Corpus
Word	Top 4 neighbours	Top 4 neighbours	Top 4 neighbours	Top 4 neighbours
AlnAs “people"	- Alkaθyr “many"	- lÂnahum “because those"	- AlÂšxAS “persons"	- AlÂfrAd “individuals"
	- AlTryqah “method"	- yurydwn “they want"	- AlAxryn “others"	- AlÂšxAS “persons"
	- Albald “country"	- yaçrfwn “they know"	- Ânahum “that they"	- Albašar “humans"
	- AlAxryn “others"	- yafçlwn “they do"	- yaçrfwn “they know"	- AlAxryn “others"
yanAyr “January"	- AlÂwl “the first"	- šahar “month"	- desambar “December"	- ywnyw “June"
	- AlθAny “the second"	- Âusbwç “week"	- Auktubar “October"	- Auktubar “October"
	- desambar “December"	- AlçAšr “the tenth"	- mArs “March"	fbrAyr “February"
	- fbrAyr “February"	- desambar “December"	- Abryl “April"	- mAyw “May"
nahar “river"	- Alnahar “the river"	- Alnahar “the river"	- Alnahar “the river"	- wanahar “and river"
	- Twl “length"	- dafn “burial"	- linahar “to the river"	- Alnahar “the river"
	- DifAf “banks of the river"	- raTbaħ “wet"	- yatadfaq “it flows"	- HwD “basin"
	- kawArθ “disasters"	- ÂnhAr “rivers"	- yaSub “it pours"	- linahar “to the river"

Table 6 shows the results of the four word embedding models on the task of non-similar word identification. All four models, as seen in Table 6, succeeded in identifying the non-similar words. This also confirms that the quality of our parallel corpus can be regarded equivalent in quality to other well-known available parallel corpora for Arabic and English.

d: RESULTS OF ENGLISH WORD SIMILARITY MODELS

The performances of the English models were examined capturing word semantic similarity in order to *find the most similar words* and to *identify non-similar words* as seen in Table 7 and Table 8.

The English part of our parallel corpus performed well on both tasks in comparison with other corpora. In fact, the model trained on our corpus yielded better results for the non-similar identification task than other models. For instance, from the list of words (camel, lion, wolf, flower),

TABLE 6. Non-similar word identification task using Arabic word embedding models trained on four various corpora.

	Global-Voices	TED	Wiki-OPUS	Our Parallel Corpus
List of words	Non similar word	Non similar word	Non similar word	Non similar word
qalb “heart", madynħ “city", manTqħ “region", muHAfĎħ “province"	qalb “heart"	qalb “heart"	qalb “heart"	qalb “heart"
yanAyr “January", fbrAyr “February", kwkb “planet", dysmbar “December"	kwkb “planet"	kwkb “planet"	kwkb “planet"	kwkb “planet"
Alhind “India", AlHaswb “computer", AlkitAb “book", Alqalam “pen"	Alhind “India"	Alhind “India"	Alhind “India"	Alhind “India"
ryAĎħ “sport", Tabyb “doctor", mumarDħ “nurse", muçlim “teacher"	ryAĎħ “sport"	ryAĎħ “sport"	ryAĎħ “sport"	ryAĎħ “sport"

TABLE 7. Examples of English words and their top four nearest neighbours according to the Arabic word embedding models for the four corpora.

	GlobalVoices	TED	Wiki-OPUS	Our Parallel Corpus
Word	Top 4 neighbours	Top 4 neighbours	Top 4 neighbours	Top 4 neighbours
committee	member	commission	Storting	committees
	forming	Senate	council	meeting
	accreditation	hoped	electing	Consultative
	rejected	headquarters	meeting	council
Finland	Saami	inland	Sweden	Sweden
	Canada	Kazakhstan	Finnish	Denmark
	tolerate	Turkey	Helsinki	Iceland
	affecting	mainland	Russia	Norway
garden	green	gardener	furniture	grove
	planting	compost	cultivate	orchard
	Akka	gardening	botanical	gardens
	fruit	amusement	pine	flower

only our corpus model was able to detect the non similar word (flower), while the other models extracted (camel) as a non-similar word in the list.

B. ARABIC-ENGLISH NMT SYSTEMS

The parallel corpus is commonly used to build NMT systems. The higher the quality of the parallel corpus, the better the performance of the machine translator. In this section, we utilised our automatically created parallel corpus and its counterpart corpora, namely, GlobalVoices, TED, and Wiki-OPUS, in order to build NMT systems for both Arabic-to-English and English-to-Arabic language pairs. Then, we compared the results of the NMT systems to assess the quality of our parallel corpus and to determine its efficiency in relation to its counterparts when used to build the MT systems. For each of the four corpora, we randomly selected 2,500 sentences and dedicated them to testing, while the remaining data in each corpus was dedicated to training.

TABLE 8. Non-similar word identification task using English word embedding models trained on four various corpora.

	Global-Voices	TED	Wiki-OPUS	Our Parallel Corpus
List of words	Non similar word	Non similar word	Non similar word	Non similar word
door, gate, portal, tiger	portal	tiger	tiger	tiger
chair, table, desk, child	child	child	child	child
camel, lion, wolf, flower	camel	camel	camel	flower
student, manager, engineer, moon	moon	moon	moon	moon
bee, wardrobe, scorpio, butterfly	scorpio	wardrobe	scorpio	wardrobe

We utilised the Transformer neural network to develop NMT systems. The Transformer neural network is a model architecture depending entirely on an attention mechanism to derive global dependencies between the input and the output [73], [74]. The training parameters were kept constant when training all NMT systems as follows: dimension of the model = 256, number of layers = 6, batch size = 64, dropout rate = 0.1, number of epochs = 1000, number of heads = 8, and feed-forward neural network units = 1024.

We built four NMT systems for translating from English into Arabic using our parallel corpus and its counterparts: GlobalVoices, TED, and Wiki-OPUS. Figure 7 shows the accuracy of the English-to-Arabic systems during the training process.

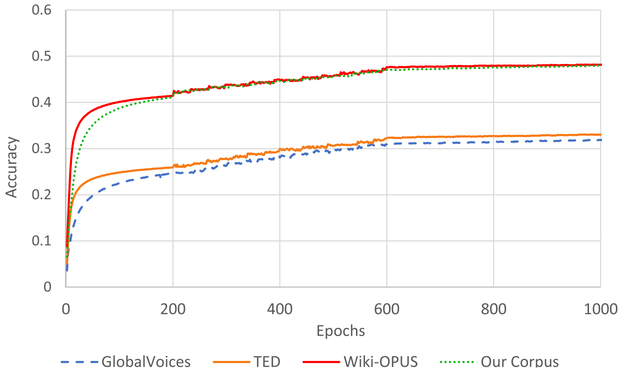


FIGURE 7. The accuracy of the English-to-Arabic translation models of GlobalVoices, TED, Wiki-OPUS, and our corpus during the training phase.

During the training, the accuracy of the model of our corpus increases continuously through the 1000 epochs, competing with the Wiki-OPUS model and outperforming the GlobalVoices and TED models.

We also utilised the Transformer neural network to develop Arabic-to-English NMT systems for the four corpora.

Figure 8 shows the accuracy of the Arabic-to-English translation systems during the training process.

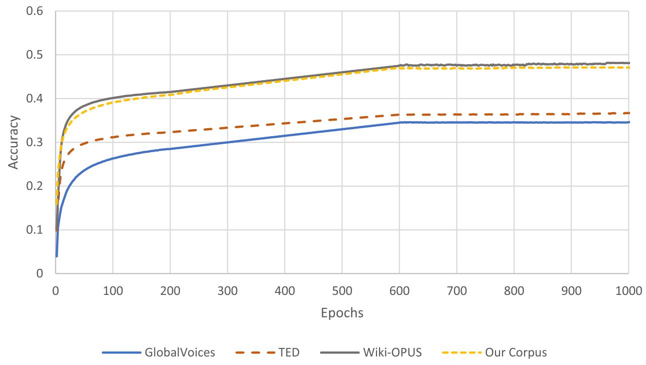


FIGURE 8. **The accuracy of the Arabic-to-English translation models of GlobalVoices, TED, Wiki-OPUS, and our corpus during the training phase.**

e: EVALUATION

We evaluated the results of the aforementioned NMT systems using two metrics: the Bilingual Evaluation Understudy (BLEU) [75] and the Metric for Evaluation of Translation with Explicit Ordering (METEOR) [76]. The BLEU metric depends on modified precision and an n-gram-based lexical similarity to compute the scores. Therein, the number of common n-grams in the corresponding generated and reference translations is divided by the total number of words in the generated translations in the test corpus. The comparison between generated and reference translations is made regardless of word order. We used the implementation of the BLEU score provided by the NLTK, the Python Natural Language Toolkit Library.[12] On the other hand, the METEOR[13] includes a fragmentation penalty that considers how well-ordered the matched words in the generated translation are in relation to the reference translations. The METEOR metric is also characterised by its flexibility to not only lexically match words (i.e., exact matching) but also somatically match words using stemming, paraphrases, and WordNet synonyms. Unlike the BLEU score, recall is taken into account, in addition to precision, by METEOR when computing the scores [77]. We utilised the BLEU score because of its popularity and the long tradition of using it for MT evaluation. The METEOR was also used in our study, because it has been shown to correlate much better with human judgment [76]. In addition, we performed a manual analysis of the generated translations produced by various NMT systems, presenting some examples of source sentences, their reference translations and the generated translations produced for them by NMT systems.

We conducted two types of evaluations: within-corpus evaluation (i.e., training and test sets derived from the same dataset) and cross-corpus evaluation (i.e., training and test sets are derived from two different datasets). As we used the METEOR metric to assess the results of the systems (i.e., Arabic generated translations), we scored the results of each NMT system twice. The first time we used the METEOR metric without pre-processing the Arabic translations (i.e., without tokenising, normalising and pre-segmenting the text). The second time we pre-processed the Arabic translations before using METEOR to compute the scores. The AraNLP library has been used to normalise and segment the Arabic text. The segmenter produces the three Penn Arabic Treebank (PATB) clitic segmentations: conjunctions, prepositions, and pronouns [69].

f: RESULTS OF ENGLISH-TO-ARABIC NMT SYSTEMS

The results for translating from English to Arabic is presented in Table 9.

TABLE 9. **The BLEU and METEOR scores of the English-to-Arabic translation systems of GlobalVoices, TED, Wiki-OPUS, and our corpus on test sets (within-corpus and cross-corpus evaluations). "w/o Proc" indicates without processing the text, "w/ Proc" indicates with processing the text.**

Test set: GlobalVoices			
Model	BLEU score	METEOR w/o Proc	METEOR w/ Proc
GV	28.74	36.99	41
TED	28.64	40.78	51.46
Wiki-OPUS	28.53	39.06	44.22
Our Corpus	28.59	39.36	45.91
Test set: TED			
Model	BLEU score	METEOR w/o Proc	METEOR w/ Proc
GV	31.06	29.25	32.21
TED	28.87	43.78	55.36
Wiki-OPUS	28.64	37.56	49.98
Our Corpus	28.58	39.67	50.63
Test set: Wiki-OPUS			
Model	BLEU score	METEOR w/o Proc	METEOR w/ Proc
GV	29.63	31.05	38.74
TED	28.82	46.72	56.29
Wiki-OPUS	28.71	53.45	61.63
Our Corpus	28.69	48.65	56.54
Test set: Our Corpus			
Model	BLEU score	METEOR w/o Proc	METEOR w/ Proc
GV	29.02	30.52	36.51
TED	28.79	46.84	58.77
Wiki-OPUS	28.71	51.46	64.13
Our Corpus	28.82	48.16	61.91

As can be seen, the performance of the English-to-Arabic translation system trained on our parallel corpus is generally on par with the NMT systems trained on other corpora (GlobalVoices, TED, and Wiki-OPUS), if not better than some. In terms of the BLEU score that depends only on precision and exact matching, the GlobalVoices system performs slightly better than other systems, even in cross-corpus evaluation, when the test set is derived from a corpus different from GlobalVoices data. However, the use of the METEOR metric shows that TED, Wiki-OPUS, and our parallel corpus systems perform better than the GlobalVoices system, taking into account that the METEOR metric is based on the weighted harmonic mean of unigram precision and unigram recall with

[12] https://www.nltk.org/
[13] https://www.cs.cmu.edu/ alavie/METEOR/

recall weighted higher than precision. This may indicate that our parallel corpus, TED, and Wiki-OPUS are better than the GlobalVoices corpus in terms of recall (i.e., the number of unigram matches divided by the total number of unigrams in the reference translation). On the other hand, in order to understand why the BLEU score for the GlobalVoices system is slightly higher than the other systems of the three remaining corpora, we manually analysed the generated translations. We observed that the generated translations produced by the GlobalVoices NMT system are short in length (i.e., the number of words in each generated translation) in comparison to the generated translations of other NMT systems. This leads to higher precision of the GlobalVoices system, considering the BLEU score depends only on precision, as explained before. Appendix A presents examples of English source sentences and their Arabic reference translations, along with the generated translations produced by the NMT systems of the four corpora, including our parallel corpus. In terms of METEOR scores, which are computed after pre-processing the Arabic texts, we notice that the scores increase for all corpora. However, NMT systems trained on our parallel corpus, TED, and Wiki-OPUS obtain better METEOR scores than the GlobalVoices system. Pre-processing Arabic texts involves segmenting the text into the three Penn Arabic Treebank (PATB) clitic segmentations: conjunctions, prepositions, and pronouns as well as the normalised stem. Therefore, having a translation system with a better METEOR score after pre-processing the text refers to the system's ability to handle clitics in Arabic, such as attached pronouns and preposition. That is, the system correctly translates the English sentence that contains multiple pronouns into the Arabic sentence with words containing attached pronouns, prepositions, and conjunctions. For example, the English segment: ''and they will write it by themselves'' has the possible Arabic reference translation: ''fasyktubwnhA biǍnfushm'' that contain only two words. In our study, the translation systems of TED, Wiki-OPUS, and our parallel corpus demonstrate their ability to correctly translate the English sentences into their corresponding Arabic sentences that contain morphologically rich words (e.g., words with attached related pronouns). The GlobalVoices NMT system falls short of the aforementioned NMT systems in terms of handling rich morphological words when translating texts; see Figure 9 as an example of one English source sentence and Arabic reference translation, as well as generated translations of various NMT systems.

Recognising each corpus's strengths and weaknesses helps explain the performance of the NMT systems trained on them. Wiki-OPUS contains noise such as repeated sentences, sentences written in a different language, and short sentences containing only numbers. Also, instead of containing the Arabic sentences that correspond with the English monolingual part of the corpus, the Arabic monolingual part of the Wiki-OPUS mistakenly includes a number of English sentences. This results in the presence of some Arabic-English parallel sentences in Wiki-OPUS, of which both sentences in each pair are the same English sentence. Therefore, some Arabic generated translations produced by the Wiki-OPUS system are a mix of Arabic and English words (see Appendix A). The advantage, however, is that the size of Wiki-OPUS helps increase the NMT system's performance and enables it to translate some source sentences well and compete with the systems of other corpora. Contrarily, the TED corpus has a conversation-like style, which may affect its performance in terms of cross-corpus evaluation on Wiki-based corpora, such as Wiki-OPUS and our corpus. Meanwhile, the greatest weakness of the GlobalVoices dataset is its small size.

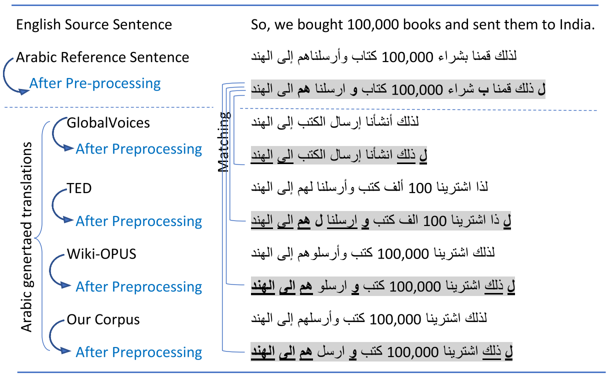

English Source Sentence
So, we bought 100,000 books and sent them to India.
Arabic Reference Sentence
لذلك قمنا بشراء 100,000 كتاب وأرسلناهم إلى الهند
After Pre-processing
ل ذلك قمنا ب شراء 100,000 كتاب و ارسلنا هم الى الهند
Matching
Arabic genertaed translations
GlobalVoices
لذلك أنشأنا إرسال الكتب إلى الهند
After Preprocessing
ل ذلك انشأنا إرسال الكتب الى الهند
TED
لذا اشترينا 100 ألف كتب وأرسلنا لهم إلى الهند
After Preprocessing
ل ذا اشترينا 100 الف كتب و ارسلنا ل هم الى الهند
Wiki-OPUS
لذلك اشترينا 100,000 كتب وأرسلوهم إلى الهند
After Preprocessing
ل ذلك اشترينا 100,000 كتب و ارسلو هم الى الهند
Our Corpus
لذلك اشترينا 100,000 كتب وأرسلهم إلى الهند
After Preprocessing
ل ذلك اشترينا 100,000 كتب و ارسل هم الى الهند

FIGURE 9. **Matching Arabic reference and generated translations after pre-processing the texts. Bolded tokens indicate the clitic segmentations after splitting them off of the words. Underlined tokens indicate the matched tokens between the reference and generated translations.**

In general, we observe that the NMT system trained on our parallel corpus performs relatively equally to systems trained on the other three corpora: GlobalVoices, TED, and Wiki-OPUS. In order to test whether the difference in performance between the NMT systems trained on the four corpora is significant, we utilised bootstrap sampling statistical tests [78], the de facto standards in NLP [79], over the systems' results (BLEU and METEOR scores). The alpha level of 0.05 was used as a significance criterion. Table 10 shows the results of each pair of our parallel corpus and the other three corpora using a bootstrap sampling with 10,000 replicates. It displays the p-values and confidence intervals of the differences between means.

Table 10 shows that the p-values >0.05 for most pairs between the NMT system trained on our corpus and those systems trained on the other three corpora. This finding suggests that the translation systems trained on the four corpora, including our corpus, are relatively similar with no significant difference. It also indicates that our parallel corpus performs equally to its counterparts when used with NLP applications such as machine translation. However, the p-values also prove that the translation system trained on our parallel corpus and the system trained on GlobalVoices are significantly different in terms of cross-corpus evaluation (i.e., tested on a different corpus from the one used in training). Our parallel corpus produces better results than the GlobalVoices corpus in terms of cross-corpus evaluation and METEOR scores as seen in Table 9. The difference between our parallel corpus and

TABLE 10. The bootstrap test results (p-values and CI) for the pairwise comparisons of English-to-Arabic NMT systems trained on our parallel corpus and its counterparts of other three corpora. "GV" indicates NMT system trained on GlobalVoices corpus. "CI" indicates 95% Confidence Interval. "p" is for p-value. "METEOR$_p$" is METEOR score computed after pre-processing Arabic texts.

		Our Corpus & GV	Our Corpus & TED	Our Corpus & Wiki-OPUS
Test set: GlobalVoices				
BLEU	p	0.616	0.5546	0.768
	[CI]	[-0.20, 0.21]	[-0.23, 0.13]	[-0.33, 0.15]
METEOR	p	0.079	0.304	0.658
	[CI]	[-1.70, 9.46]	[-5.37, 11.98]	[-8.78, 8.38]
METEOR$_p$	p	0.116	0.199	0.772
	[CI]	[-3.81, 12.59]	[-4.74, 12.27]	[-13.15, 6.83]
Test set: TED				
BLEU	p	0	0.389	0.718
	[CI]	[1.10, 3.97]	[-1.42, 2.10]	[-0.22, 0.11]
METEOR	p	0	0.095	0.237
	[CI]	[3.23, 9.97]	[-2.11, 10.19]	[-4.00, 8.08]
METEOR$_p$	p	0	0.946	0.575
	[CI]	[2.10, 7.54]	[-16.76, 1.63]	[-3.42, 4.55]
Test set: Wiki-OPUS				
BLEU	p	0.858	0.611	0.576
	[CI]	[-2.76, 0.76]	[-1.66, 1.43]	[-0.31, 0.27]
METEOR	p	0	0.285	0.901
	[CI]	[12.32, 23.15]	[-7.91, 12.92]	[-13.49, 2.80]
METEOR$_p$	p	0	0.453	0.852
	[CI]	[10.37, 26.70]	[-11.63, 12.11]	[-13.56, 3.90]
Test set: Our Corpus				
BLEU	p	0.577	0.488	0.216
	[CI]	[-2.23, 2.11]	[-2.00, 2.46]	[-0.17, 0.56]
METEOR	p	0	0.348	0.735
	[CI]	[11.45, 25.60]	[-9.02, 12.82]	[-14.02, 7.51]
METEOR$_p$	p	0	0.222	0.498
	[CI]	[16.26, 42.29]	[-9.34, 22.10]	[-16.92, 17.16]

GlobalVoices in cross-corpus evaluation is significant based on the statistical test as seen in Table 10.

g: RESULTS OF ARABIC-TO-ENGLISH NMT SYSTEMS

The results of the translation from Arabic to English are presented in Table 11. It shows the BLEU and METEOR scores of the implemented NMT systems on the test sets for both within-corpus and cross-corpus evaluations.

The Arabic-to-English system trained on our corpus achieved comparable results to the NMT systems trained on TED and Wiki-OPUS corpora although TED corpus is larger than our corpus and manually created by human translation. One observation should also be mentioned here: the performances of systems when translating from English to Arabic are better than when translating from Arabic to English. We may partially attribute this to the fact that Arabic has a complex morphology [80], [81], and therefore, translating from Arabic to English requires more training data to cover the inflected forms of the words. Also, affixes and clitics generally increase data sparsity (see Appendix B for examples of Arabic source sentences and their English reference and generated translations).

Moreover, we utilised a bootstrap sampling statistical test with 10,000 replicates and an alpha level of 0.05 over the results of Arabic-to-English translation systems, as shown in Table 12. Clearly, the difference in performance between the Arabic-to-English NMT systems of the four corpora is generally not significant, which confirms that our parallel corpus is an even match to its counterparts, even for translating from Arabic to English. Moreover, our parallel corpus outperforms the GlobalVoices corpus in terms of METEOR scores, and the statistical test indicates a significant difference between them on all test sets. As mentioned previously, METEOR metric, unlike BLEU, uses not only precision, but also recall. In addition, the METEOR metric for English flexibly finds matched unigrams and phrases using stemming and English WordNet synonyms. Our parallel corpus obtains better METEOR scores than GlobalVoices because of its ability to translate Arabic sentences into English while retaining the meaning of the source sentences as much as possible. Indeed, the generated translations of our parallel corpus contains synonyms of the words in the reference translation. For example, when the reference translation was "So the novel has won several awards", the generated translation produced by system trained on our parallel corpus was "The novel has won numerous awards and events" while GlobalVoices translation was "The novel has changed the life of the people." It is obvious that our parallel corpus accurately utilised "numerous" instead of "several" when translating the sentence and the two words are synonyms. More examples and clarification of generated translations by systems trained on other corpora can be found in Appendix B.

With all that said, the translation of words containing clitics from English to Arabic or vice versa still presents difficulty for morphologically rich languages. For example,

TABLE 11. The BLEU and METEOR scores of the Arabic-to-English translation systems of GlobalVoices, TED, Wiki-OPUS, and our corpus on test sets (within-corpus and cross-corpus evaluations).

Test set: GlobalVoices		
Model	BLEU	METEOR
GV	27.5	34.25
TED	27.54	46.58
Wiki-OPUS	27.46	46.33
Our Corpus	27.5	45.95
Test set: TED		
Model	BLEU score	METEOR w/o seg.
GV	27.57	31.71
TED	27.8	52.63
Wiki-OPUS	27.68	46.64
Our Corpus	27.78	46.67
Test set: Wiki-OPUS		
Model	BLEU score	METEOR w/o seg.
GV	27.47	35.41
TED	27.65	55.30
Wiki-OPUS	27.68	60.86
Our Corpus	27.7	54.27
Test set: Our Corpus		
Model	BLEU score	METEOR w/o seg.
GV	27.52	30.76
TED	27.63	48.48
Wiki-OPUS	27.58	55.12
Our Corpus	27.62	49.99

TABLE 12. The bootstrap test results (p-values and CI) for the pairwise comparisons of Arabic-to-English NMT systems trained on our parallel corpus and its counterparts of other three corpora. "GV" indicates an NMT system trained on GlobalVoices corpus. "CI" indicates 95% Confidence Interval. "p" is for p-value.

		Our Corpus & GV	Our Corpus & TED	Our Corpus & Wiki-OPUS
Test set: GlobalVoices				
BLEU	p	0.737	0.915	0.471
	[CI]	[-0.13, 0.08]	[-0.17, 0.03]	[-0.12, 0.12]
METEOR	p	0	0.495	0.581
	[CI]	[7.08, 15.15]	[-5.76, 5.67]	[-4.54, 3.78]
Test set: TED				
BLEU	p	0.151	0.947	0.597
	[CI]	[-0.07, 0.25]	[-0.32, 0.03]	[-0.21, 0.16]
METEOR	p	0	0.953	0.485
	[CI]	[9.86, 19.37]	[-12.42, 0.90]	[-4.85, 4.76]
Test set: Wiki-OPUS				
BLEU	p	0.023	0.284	0.385
	[CI]	[-0.01, 0.38]	[-0.15, 0.22]	[-0.10, 0.14]
METEOR	p	0	0.695	0.973
	[CI]	[14.83, 23.93]	[-8.42, 4.84]	[-12.37, 0.11]
Test set: Our Corpus				
BLEU	p	0.11	0.533	0.276
	[CI]	[-0.07, 0.24]	[-0.15, 0.13]	[-0.09, 0.17]
METEOR	p	0	0.463	0.898
	[CI]	[15.32, 25.46]	[-8.07, 9.28]	[-13.75, 3.00]

the morphologically rich Arabic word "fasykfykahm" which means "So will suffice you against them" is translated wrongly by many online translators, such as Google translator which translate the Arabic word into "It will suffice them". These cases require more investigation and analysis. They can be examined from two points of view: (a) the method utilised for building the NMT system and (b) the parallel corpora collected and prepared for training the MT system.

XI. CONCLUSION

This paper described our straightforward yet robust algorithm to automatically create parallel sentences from comparable corpora. The proposed method requires only a bilingual dictionary and a word vectorisation method. The steps can be summarised as follows: (a) align articles that are on the same topic in Arabic and English, (b) translate English sentences into Arabic using a bilingual dictionary (pseudo-Arabic sentences), (c) automatically align Arabic sentences and pseudo-Arabic sentences by computing the similarity between sentences. The use of the bilingual dictionary required in the translation steps reduces the computational cost of building a machine translation system or using online ones. In addition, our proposed approach does not require correct grammar when computing similarities and matching sentences (i.e., actual Arabic sentences and pseudo-Arabic sentences). In our study, we also compared two word vectorisation methods: TFIDF and neural network-based word embedding. TFIDF computed for words at the document level proved to yield better results when computing similarity between sentences with mistakes in structures and syntax.

We utilised Arabic and English Wikipedia, a free source of comparable corpus. The resulting Arabic-English corpus consists of 105,010 parallel sentences with a total number of 4.6M words. The quantitative and qualitative assessments of the resulting corpus revealed the satisfactory quality of our parallel corpus in comparison with other available Arabic-English parallel corpora counterparts: GlobalVoices, TED and Wiki-OPUS. The Arabic and English semantic word similarity models trained on our parallel corpus competed evenly with models trained on other corpora in the task of similar word identification, but outperformed other models in the task of English non-similar word identification. Utilising our parallel corpus and the three aforementioned corpora, we also built English-to-Arabic NMT tools. The NMT tool trained on our parallel corpus yielded results comparable to those of other corpora with an average BLEU score equal to 28.62 and an average METEOR score equal to 51.03 for cross-corpus evaluation. Regarding within-corpus evaluation, the NMT system of our parallel corpus achieved a BLEU score of 28.82 and a METEOR score of 61.91. The bootstrap sampling statistical test over the results of NMT systems suggested that the systems trained on the four corpora, including our corpus, are not significantly different. This finding confirms that our parallel corpus produces results comparable to those of other corpora counterparts when used to build NLP systems such as machine translators. The use of our parallel corpus to build an Arabic-to-English NMT system also demonstrated our parallel corpus capability to train a translator that performed equally to other corpora with an average BLEU score equal to 27.66 and an average METEOR score equal to 48.96 for cross-corpus evaluation. For within-corpus evaluation, the BLEU and METEOR scores were 27.62 and 49.99 respectively. The NMT systems trained on TED and Wiki-OPUS showed better results than those trained on our parallel corpus. However, the statistical significance test (i.e., bootstrap sampling test) revealed no significant differences between them.

Our study revealed that the document-level TFIDF method for word vectorisation led to better results for aligning sentences and computing similarity scores than pre-trained word embeddings computed based on the entire Wikipedia corpus. We also found that the NMT systems trained on large corpora drawn from the open domain were better at generating accurate translations that were closer to reference translations in terms of translating morphological segments attached to the words, such as related pronouns, prepositions, and conjunctions. The METEOR score of generated Arabic translations after segmenting them into clitics showed that TED, Wiki-OPUS, and our parallel corpus returned better results in terms of learning to translate rich morphological words. The NMT system trained on our parallel corpus performed solidly, handling clitics similar to its counterparts trained on larger corpora (TED and Wiki-OPUS). On the other hand, our parallel corpus NMT system showed significantly better results than the GlobalVoices corpus when dealing with translating clitics, clearly evidenced by the METEOR$_p$ scores.

The METEOR$_p$ scores were computed after segmenting the generated Arabic translations and juxtaposing them with the segmented reference translations.

In the future, we plan to apply the approach to create parallel corpora from Wikipedia for other language/dialect pairs: (Egyptian Arabic-MSA), (Moroccan Arabic-MSA), (Egyptian Arabic-English), and (Moroccan Arabic-English). We also plan to apply our proposed algorithm to online comparable corpora, such as bilingual websites and blogs, to automatically extract parallel sentences for various language pairs.

During our study we concluded that the Arabic-to-English MT system usually performs slightly less effectively than the English-to-Arabic MT system. We may partially attribute that to the fact that Arabic has a complex morphology where affixes and clitics generally increase the data sparsity, and thus Arabic to English translation requires more training data to cover the inflected forms of the words. More investigations can be dedicated to this point to move machine translation for rich morphological languages, such as Arabic, forward. Further studies are necessary to analyse the available parallel sentences in terms of their coverage of the morphological features of the language and their size, along with how these two elements would affect the overall performance of the NLP systems when trained on these parallel corpora (e.g., neural machine translation).

We believe that the quality of the bilingual dictionary has a strong impact on the overall quality of our proposed method. In our current study, we used Wikipedia, an open domain database, and therefore, a comprehensive bilingual dictionary proved sufficient. Nevertheless, the specialised comparable corpora (e.g., from the health and education domains) require specialised bilingual dictionaries to extract parallel sentences with high accuracy. In a future study we will explore the use of minimally supervised methods to build specialised bilingual dictionaries in less time and effort.

APPENDIX A
EXAMPLES OF GENERATED TRANSLATIONS BY ENGLISH-TO-ARABIC NMT systems[14]

Example 1		
English Source Sentence		Around 1940 he went to Chicago
Arabic Reference Translation		ذهب إلى شيكاغو في حوالي عام 1940
Arabic generated translations	GlobalVoices	في عام 1940 انضم إلى الحزب في شيكاغو
	English gloss	In 1940 he joined the party in Chicago
	TED	حوالي 1940 ذهب إلى شيكاغو
	English gloss	Around 1940 he went to Chicago
	Wiki-OPUS	في حوالي 1940 ذهب إلى شيكاغو
	English gloss	In around 1940 he went to Chicago
	Our Corpus	في عام 1940 ذهب إلى شيكاغو
	English gloss	In 1940 he went to Chicago

Example 2		
English Source Sentence		Fet is a municipality in Akershus county Norway
Arabic Reference Translation		فيت هي بلدية في مقاطعة آكرشوس بالنرويج وتنتمي لمنطقة روميريك التاريخية
Arabic generated translations	GlobalVoices	ماروك بلوغ في لندن
	English gloss	Maroc blog in London
	TED	إن الحركة هي طريقة ملكية في Arkerts للإتصال بالسعادة
	English gloss	Movement is Arkerts' proprietary way of communicating happiness
	Wiki-OPUS	فيت (النرويج) هي municipality of Norman in Akershus county Norway.
	English gloss	Fet (Norway) is municipality of Norman in Akershus county Norway.
	Our Corpus	لاري هي بلدية في مقاطعة آكرشوس بالنرويج وتعد جزءا من منطقة فولو التاريخية .
	English gloss	Lari is a municipality in Akershus county, Norway, and is part of the Volo Historic District.

Example 3		
English Source Sentence		Nearly one thousand species of plants with edible leaves are known
Arabic Reference Translation		ما يقرب من ألف نوع من النباتات التي تصلح أوراقها للأكل
Arabic generated translations	GlobalVoices	جيل الطائرات المقاتلة فوق الألغام
	English gloss	The generation of fighter planes over mines
	TED	تقريباً ألف نوع من النباتات مع أوراق النباتات ذات جودة معروفة
	English gloss	Almost a thousand species of plants with leaves of known quality
	Wiki-OPUS	ما يقرب من ألف نوع من النباتات التي نعرف بالبرية
	English gloss	There are nearly a thousand species of plants that we know in the wild
	Our Corpus	يتم إنتاج أنواع أخرى من الأشجار وما يقرب من ألف نوع من النباتات تصلح للأكل
	English gloss	Other types of trees are produced and nearly a thousand types of plants which are edible

Example 4		
English Source Sentence		A new draft bill lays the groundwork for an autonomous RuNet
Arabic Reference Translation		مشروع قانون جديد يمهّد لإطلاق شبكة إنترنت روسية مستقلة
Arabic generated translations	GlobalVoices	مشروع قانون جديد يمهّد لمحاكمة مستقلة
	English gloss	A new bill paves the way for an independent trial
	TED	ورقة إنارة جديدة تضع المسألة في أرض الواقع لجنيت روزي. ممتازة
	English gloss	A new illuminating paper puts the matter on the ground for Jeanette Rosie. Excellent
	Wiki-OPUS	A new draft bills lays the groundwork for an مستقلة RuNet. —. NS.
	English gloss	A new draft bills lays the groundwork for an independent RuNet. —. NS.
	Our Corpus	إن نظام الحكم الذاتي الأصلي هو إنشاء مشروع قانون جديد من أوضاع مستقلة
	English gloss	The original autonomy system is to create a new bill of independent situations

Example 1		
Arabic Source Sentence		وفي عام 1919، تم تدريس الدورات الدراسية الجامعية للمرة الأولى باللغة الاستونية
English Reference Translation		In 1919, university courses were first taught in the Estonian language
English generated translations	GlobalVoices	The GNU League in Estonia was highlighted for the first time
	TED	An in 1919, school taught school instructions for the first time in Estonia
	Wiki-OPUS	In 1919, the undergraduate school cycles were teaching for the first time in Estonian language
	Our Corpus	The school lectures were first taught in the Estonian language

[14] Please note "English gloss" is indicative to explain the resulted Arabic translations.

APPENDIX B
EXAMPLES OF GENERATED TRANSLATIONS BY ARABIC-TO-ENGLISH NMT SYSTEMS

Example 2		
Arabic Source Sentence		ولذلك فازت هذه الرواية بالعديد من الجوائز
English Reference Translation		So the novel has won several awards
English generated translations	GlobalVoices	The novel has changed the life of the people
	TED	So these novelty won a lot of awards
	Wiki-OPUS	The novel predeceased many awards
	Our Corpus	The novel has won numerous awards and events

REFERENCES

[1] S.-W. Chan, *Routledge Encyclopedia of Translation Technology*. Evanston, IL, USA: Routledge, 2014.

[2] P. F. Brown, J. Cocke, S. A. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, and P. S. Roossin, "A statistical approach to machine translation," *Comput. Linguistics*, vol. 16, no. 2, pp. 79–85, 1990.

[3] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 3104–3112.

[4] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–15.

[5] Y. Marton, C. Callison-Burch, and P. Resnik, "Improved statistical machine translation using monolingually-derived paraphrases," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Singapore, vol. 1, 2009, pp. 381–390.

[6] R. C. Moore, "Fast and accurate sentence alignment of bilingual corpora," in *Proc. 5th Conf. Assoc. Mach. Transl. Amer.* Tiburon, CA, USA: Springer, 2002, pp. 135–144.

[7] F. Braune and A. Fraser, "Improved unsupervised sentence alignment for symmetrical and asymmetrical parallel corpora," in *Proc. 23rd Int. Conf. Comput. Linguistics (Coling)*, Beijing, China, 2010, pp. 81–89.

[8] W. A. Gale and K. Church, "A program for aligning sentences in bilingual corpora," *Comput. Linguistics*, vol. 19, no. 1, pp. 75–102, 1993.

[9] R. Sennrich and M. Volk, "MT-based sentence alignment for OCR-generated parallel texts," in *Proc. 9th Conf. Assoc. Mach. Transl. Amer.*, Denver, CO, USA, 2010.

[10] D. Varga, P. Halácsy, A. Kornai, V. Nagy, L. Németh, and V. Trón, "Parallel corpora for medium density languages," in *Recent Advances in Natural Language Processing*. Borovets, Bulgaria: Association for Computational Linguistics, 2005, pp. 590–596.

[11] X. Ma, "Champollion: A robust parallel text sentence aligner," in *Proc. 5th Int. Conf. Lang. Resour. Eval. (LREC)*, Genova, Italy, 2006, pp. 489–492.

[12] L. Alemany, E. Andrawos, R. Carrascosa, G. Berrotarán, and A. Vine. (2013). *Yalign: A Tool for Extracting Parallel Sentences From Comparable Corpora*. Accessed: Sep. 20, 2021. [Online]. Available: https://yalign.readthedocs.io/en/latest/

[13] J. Tiedemann, "Lingua-align: An experimental toolbox for automatic tree-to-tree alignment," in *Proc. Int. Conf. Lang. Resour. Eval.*, Valletta, Malta, 2010, pp. 1–30.

[14] K. Yasuda and E. Sumita, "Method for building sentence-aligned corpus from Wikipedia," in *Proc. AAAI Workshop Wikipedia Artif. Intell. (WikiAI)*, Chicago, IL, USA, 2008, pp. 263–268.

[15] C. Chu, T. Nakazawa, and S. Kurohashi, "Constructing a Chinese—Japanese parallel corpus from Wikipedia," in *Proc. 9th Int. Conf. Lang. Resour. Eval. (LREC)*, Reykjavik, Iceland, 2014, pp. 642–647.

[16] R. Navigli, "Word sense disambiguation: A survey," *ACM Comput. Surv.*, vol. 41, no. 2, pp. 1–69, 2009.

[17] M. A. Fauzi, D. C. Utomo, B. D. Setiawan, and E. S. Pramukantoro, "Automatic essay scoring system using N-gram and cosine similarity for gamification based E-learning," in *Proc. Int. Conf. Adv. Image Process.*, Bangkok, Thailand, Aug. 2017, pp. 151–155.

[18] D. Kravchenko, "Paraphrase detection using machine translation and textual similarity algorithms," in *Proc. Conf. Artif. Intell. Natural Lang.* Cham, Switzerland: Springer, 2017, pp. 277–292.

[19] A. Islam and D. Inkpen, "Semantic text similarity using corpus-based word similarity and string similarity," *ACM Trans. Knowl. Discovery Data*, vol. 2, no. 2, pp. 1–25, 2008.

[20] T. Kenter and M. de Rijke, "Short text similarity with word embeddings," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Melbourne, VIC, Australia, Oct. 2015, pp. 1411–1420.

[21] F. Šarić, G. Glavaš, M. Karan, J. Šnajder, and B. D. Bašić, "Takelab: Systems for measuring semantic text similarity," in *Proc. 1st Joint Conf. Lexical Comput. Semantics, Main Conf. Shared Task, 6th Int. Workshop Semantic Eval. (SemEval)*, Montreal, BC, Canada, vol. 1, 2012, pp. 441–448.

[22] E. Brill and R. C. Moore, "An improved error model for noisy channel spelling correction," in *Proc. 38th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Hong Kong, 2000, pp. 286–293.

[23] G. V. Bard, "Spelling-error tolerant, order-independent pass-phrases via the Damerau-Levenshtein string-edit distance metric," in *Proc. 5th Austral. Symp. (ACSW Frontiers)*, vol. 68. Ballarat, VIC, Australia: Australian Computer Society, 2007, pp. 117–124.

[24] M. A. Jaro, "Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida," *J. Amer. Stat. Assoc.*, vol. 84, no. 406, pp. 414–420, Jun. 1989.

[25] W. E. Winkler, "String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage," in *Proc. ERIC*, 1990, pp. 354–359.

[26] M. A. Jaro, "Probabilistic linkage of large public health data files," *Statist. Med.*, vol. 14, nos. 5–7, pp. 491–498, Mar. 1995.

[27] A. Barrón-Cedeno, P. Rosso, E. Agirre, and G. Labaka, "Plagiarism detection across distant language pairs," in *Proc. 23rd Int. Conf. Comput. Linguistics (Coling)*, Beijing, China, 2010, pp. 37–45.

[28] G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández, "Syntactic N-grams as machine learning features for natural language processing," *Expert Syst. Appl.*, vol. 41, no. 3, pp. 853–860, Feb. 2014.

[29] S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, "Using of Jaccard coefficient for keywords similarity," in *Proc. Int. Multiconf. Eng. Comput. Scientists*, Hong Kong, 2013, pp. 380–384.

[30] R. Zhao and K. Mao, "Fuzzy bag-of-words model for document representation," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 794–804, Apr. 2018.

[31] L. Havrlant and V. Kreinovich, "A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation)," *Int. J. Gen. Syst.*, vol. 46, no. 1, pp. 27–36, Mar. 2017.

[32] N. S. Mohd Nafis and S. Awang, "An enhanced hybrid feature selection technique using term frequency-inverse document frequency and support vector machine-recursive feature elimination for sentiment classification," *IEEE Access*, vol. 9, pp. 52177–52192, 2021.

[33] M. J. Althobaiti, "Country-level Arabic dialect identification using small datasets with integrated machine learning techniques and deep learning models," in *Proc. 6th Arabic Natural Lang. Process. Workshop*, Kyiv, Ukraine, 2021, pp. 265–270.

[34] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Lake Tahoe, NV, USA: Curran Associates, 2013, pp. 3111–3119.

[35] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.

[36] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, "FastText.Zip: Compressing text classification models," 2016, *arXiv:1612.03651*.

[37] P. D. Turney, "Mining the web for synonyms: PMI-IR versus LSA on TOEFL," in *Proc. 12th Eur. Conf. Mach. Learn.* Freiburg, Germany: Springer, 2001, pp. 491–502.

[38] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using Wikipedia-based explicit semantic analysis," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, Hyderabad, India, vol. 7, 2007, pp. 1606–1611.

[39] S. T. Dumais, "Latent semantic analysis," *Annu. Rev. Inf. Sci. Technol.*, vol. 38, no. 1, pp. 188–230, 2004.

[40] H. M. Ismail, B. Belkhouche, and S. Harous, "Framework for personalized content recommendations to support informal learning in massively diverse information Wikis," *IEEE Access*, vol. 7, pp. 172752–172773, 2019.

[41] J. Tiedemann, "Bitext alignment," *Synth. Lectures Hum. Lang. Technol.*, vol. 4, no. 2, pp. 1–165, May 2011.

[42] P. Li, M. Sun, and P. Xue, "Fast-champollion: A fast and robust sentence alignment algorithm," in *Proc. Int. Conf. Comput. Linguistics*, Beijing, China, 2010, pp. 710–718.

[43] M. Simard and P. Plamondon, "Bilingual sentence alignment: Balancing robustness and accuracy," *Mach. Transl.*, vol. 13, no. 1, pp. 59–80, 1998.

[44] P. Fung and K. W. Church, "K-vec: A new approach for aligning parallel texts," in *Proc. 15th Conf. Comput. Linguistics*, vol. 2, 1994, pp. 1096–1102.

[45] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Comput. Linguistics*, vol. 19, no. 2, pp. 263–311, Jun. 1993.

[46] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *J. Mol. Biol.*, vol. 48, no. 3, pp. 443–453, Mar. 1970.

[47] F. Guzmán, H. Sajjad, S. Vogel, and A. Abdelali, "The AMARA corpus: Building resources for translating the web's educational content," in *Proc. 10th Int. Workshop Spoken Lang. Technol. (IWSLT)*, Berlin, Germany, 2013, pp. 204–212.

[48] K. Wołk and K. Marasek, "Building subject-aligned comparable corpora and mining it for truly parallel sentence pairs," *Proc. Technol.*, vol. 18, pp. 126–132, Jan. 2014.

[49] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen, "The United Nations parallel corpus V1. 0," in *Proc. 10th Int. Conf. Lang. Resour. Eval. (LREC)*, Portoroz, Slovenia, 2016, pp. 3530–3534.

[50] M. Cettolo, C. Girardi, and M. Federico, "Wit3: Web inventory of transcribed and translated talks," in *Proc. 16th Annu. Conf. Eur. Assoc. Mach. Transl.*, Trento, Italy, 2012, pp. 261–268.

[51] CASMACAT. (2018). *Global Voices Parallel Corpus 2018Q4*. Accessed: Oct. 11, 2020. [Online]. Available: http://casmacat.eu/corpus/global-voices.html

[52] J. Tiedemann, "Parallel data, tools and interfaces in OPUS," in *Proc. 8th Int. Conf. Lang. Resour. Eval. (LREC)*, Istanbul, Turkey, May 2012.

[53] P. Koehn, H. Hoang, A. Birch, and C. Callison-Burch, "Moses: Open source toolkit for statistical machine translation," in *Proc. 45th Annu. Meeting Assoc. Comput. Linguistics Companion Volume Demo Poster Sessions*, Prague, Czech Republic, 2007, pp. 177–180.

[54] CASMACAT. (2013). *TED Talk Parallel Corpus*. Accessed: Oct. 12, 2020. [Online]. Available: http://www.casmacat.eu/corpus/ted2013.html

[55] N. Reimers and I. Gurevych, "Making monolingual sentence embeddings multilingual using knowledge distillation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 4512–4525.

[56] A. Abdelali, F. Guzman, H. Sajjad, and S. Vogel, "The AMARA corpus: Building parallel language resources for the educational domain," in *Proc. 9th Int. Conf. Lang. Resour. Eval. (LREC)*, Reykjavik, Iceland, 2014, pp. 1044–1054.

[57] A. Grigas. (2020). *Analytics Wikistats 2, Statistics For Wikimedia Projects. Wikimedia Foundation*. Accessed: Jun. 3, 2020. [Online]. Available: https://stats.wikimedia.org/

[58] R. Ghani, S. Slattery, and Y. Yang, "Hypertext categorization using hyperlink patterns and meta data," in *Proc. 18th Int. Conf. Mach. Learn. (ICML)*, vol. 1. San Francisco, CA, USA: Morgan Kaufmann, 2001, pp. 178–185.

[59] S. F. Adafre and M. De Rijke, "Finding similar sentences across multiple languages in Wikipedia," in *Proc. Workshop NEW TEXT Wikis Blogs Other Dyn. Text Sources*, Trento, Italy, 2006, pp. 62–69.

[60] N. Habash, A. Soudi, and T. Buckwalter, "On Arabic transliteration," in *Arabic Computational Morphology* (Text, Speech and Language Technology), A. Soudi, A. van den Bosch, and G. Neumann, Eds. Dordrecht, The Netherlands: Springer, 2007, pp. 15–22.

[61] K. Yu and J. Tsujii, "Bilingual dictionary extraction from Wikipedia," in *Proc. 12th Mach. Transl. Summit*, Ottawa, ON, Canada, 2009, pp. 379–386.

[62] P. G. Otero and I. G. López, "Wikipedia as multilingual source of comparable corpora," in *Proc. 3rd Workshop Building Using Comparable Corpora (LREC)*, Valletta, Malta, 2010, pp. 21–25.

[63] M. R. Costa-jussà, P. Li Lin, and C. España-Bonet, "GeBioToolkit: Automatic extraction of gender-balanced multilingual corpus of Wikipedia biographies," 2019, *arXiv:1912.04778*.

[64] J. Smith, C. Quirk, and K. Toutanova, "Extracting parallel sentences from comparable corpora using document level alignment," in *Proc. Hum. Lang. Technol., Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, Los Angeles, CA, USA, 2010, pp. 403–411.

[65] P. Fung and P. Cheung, "Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus," in *Proc. 20th Int. Conf. Comput. Linguistics (COLING)*, Geneva, Switzerland, 2004, pp. 1051–1057.

[66] Y. Alrasheedi. (2018). *English–Arabic Dictionary MySQL Database Based on WordList From ArabEyes Team*. Accessed: Nov. 10, 2020. [Online]. Available: https://github.com/usefksa/engAraDictionaryFrom_ArabEyes

[67] M. Attia. (2015). *Arabic Wordlist for Spellchecking*. Accessed: Nov. 19, 2020. [Online]. Available: https://sourceforge.net/projects/arabic-wordlist/

[68] S. Khoja and R. Garside. (1999). *Stemming Arabic Text*. Lancaster, U.K., Comput. Dept., Lancaster Univ. Accessed: Feb. 23, 2015. [Online]. Available: http://zeus.cs.pacificu.edu/shereen/research.htm

[69] M. Althobaiti, U. Kruschwitz, and M. Poesio, "AraNLP: A Java-based library for the processing of Arabic text," in *Proc. 9th Int. Conf. Lang. Resour. Eval. (LREC)*, Reykjavik, Iceland, 2014, pp. 4134–4138.

[70] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.

[71] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, Atlanta, GA, USA, 2013, pp. 746–751.

[72] J. Han, "Getting to know your data," in *Data Mining: Concepts and Techniques* (The Morgan Kaufmann Series in Data Management Systems), J. Han, M. Kamber, and J. Pei, Eds. Boston, MA, USA: Morgan Kaufmann, 2012, pp. 39–82. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123814791000022

[73] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Long Beach, CA, USA: Curran Associates, 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[74] X. Ji, Y. Tu, W. He, J. Wang, H.-W. Shen, and P.-Y. Yen, "USEVis: Visual analytics of attention-based neural embedding in information retrieval," *Vis. Informat.*, vol. 5, no. 2, pp. 1–12, Jun. 2021.

[75] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, Philadelphia, PA, USA, 2002, pp. 311–318.

[76] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proc. 9th Workshop Stat. Mach. Transl.*, Baltimore, MD, USA, 2014, pp. 376–380.

[77] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. Workshop Intrinsic Extrinsic Eval. Measures Mach. Transl. Summarization (ACL)*, Ann Arbor, MI, USA, 2005, pp. 65–72.

[78] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*. Boca Raton, FL, USA: CRC Press, 1994.

[79] A. Søgaard, A. Johannsen, B. Plank, D. Hovy, and H. M. Alonso, "What's in a p-value in NLP?" in *Proc. 18th Conf. Comput. Natural Lang. Learn.*, Baltimore, MD, USA, 2014, pp. 1–10.

[80] N. Y. Habash, "Introduction to Arabic natural language processing," *Synth. Lectures Hum. Lang. Technol.*, vol. 3, no. 1, pp. 1–187, Jan. 2010.

[81] M. J. Althobaiti, "Creation of annotated country-level dialectal Arabic resources: An unsupervised approach," *Natural Lang. Eng.*, pp. 1–42, Aug. 2021, doi: 10.1017/s135132492100019x.

MAHA JARALLAH ALTHOBAITI received the bachelor's degree (Hons.) in computer science from the Department of Computer Science, Taif University, Saudi Arabia, and the master's (Hons.) and Ph.D. degrees in computer science from the University of Essex, Colchester, U.K. Her thesis entitled Minimally-Supervised Methods for Arabic Named Entity Recognition.

She is currently an Assistant Professor with the Department of Computer Science, College of Computers and Information Technology, Taif University. Her research expertise and interests include Arabic natural language processing, computational linguistics, data analytics, data mining, deep learning, semi-supervised learning, and unsupervised learning.