Ensemble Transformer for cross lingual semantic textual similarity

Poorya Piroozfar
Iran University of Science and Technology
Mohammad Abdous
Iran University of Science and Technology
Behrouz Minaei Bidgoli (✉ b_minaei@iust.ac.ir)
Iran University of Science and Technology

Research Article

Keywords: sentence similarity, cross lingual sentences, sentence representation, transformers, Semantic similarity

Posted Date: January 17th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-3860915/v1

Additional Declarations: No competing interests reported.

Ensemble Transformer for cross lingual semantic textual similarity

Poorya Piroozfar [a], Mohammad Abdous [a] and Behrouz Minaei Bidgoli [a1]
[a] *Department of Computer Engineering, Iran University of Science and Technology, Iran*
E-mails: *poorya_piroozfar@alumni.iust.ac.ir, mohammadabdous@comp.iust.ac.ir, b_minaei@iust.ac.ir*

Abstract

Today, it is particularly important to recognize the semantic similarity between texts in different languages due to the emergence of new natural language processing models like ChatGPT and Bard. These models can provide more accurate and comprehensive answers to users' questions by identifying semantic similarity between two texts in different languages. Cross-lingual semantic similarity refers to the process of calculating similarity between two pieces of text in different languages. This paper aims to present an improved method for finding similarities between sentences in different languages. Some of the current methods create the same vector space to achieve this, while others use machine translation to translate the text into another language and then determine similarity between the two sentences using monolingual sentence similarity methods. The degree of similarity is expressed as a number between 0 and 5. Over the past few years, the progress in language models based on transformers has paved the way for improvements in detecting text similarity. This article discusses the utilization of ensemble models with transformers to determine the semantic similarity of sentences in Persian and English languages utilizing the Persian-English corpus. According to our findings, this ensemble approach has a correlation rate of 95.28% in detecting the extent of semantic similarity between cross-lingual sentences. These results indicate that our method surpasses previous techniques for discovering similarities between sentences in different languages.

Keywords: sentence similarity, cross lingual sentences, sentence representation, transformers, Semantic similarity

1 Introduction

Sentences can be evaluated for similarity within a language or across multiple languages. The objective is to measure the degree of semantic similarity between sentences on a scale of 0 to 5, where 0 signifies no similarity and 5 denotes complete semantic similarity. The primary obstacle is to determine a fitting semantic vector that can accurately represent the semantic complexity of sentences. One method to achieve vector representation of sentences is to utilize the representation vector of the constituent words [1].

A crucial factor to consider in the task of determining sentence similarity is that it is dependent on the language, and the precision of the methods differs across various language pairs. This difference is contingent on the availability of data for each language pair. To tackle this problem, we constructed a dataset for the Persian-English language pair [2] and devised ensemble methods

[1] Corresponding Author. ORCID: https://orcid.org/0000-0002-9327-7345

using this dataset. Through this approach, we were able to enhance the performance of sentence similarity in the Persian-English language pair.

Measuring semantic similarity between words, terms, sentences, paragraphs, and documents is a vital aspect of natural language processing and computational linguistics [3]. It has numerous applications, including question answering systems, machine translation, information retrieval, fraud detection, and more [4-8]. In this article, we explore various techniques for determining semantic similarity across languages and introduce an improvement by proposing an ensemble method based on transformer models. A crucial aspect highlighted in this research is that two sentences with similar semantics should have vectors that are positioned closely in vector space, which enables the determination of their semantic similarity using different criteria.

This article focuses solely on enhancing the task of detecting semantic similarity between cross-lingual sentences, specifically in the Persian-English language pair. We achieve this by utilizing multilingual transformers, which are fine-tuned with a dataset.

This article examines the cross-linguistic similarity of Persian-English sentence pairs for the first time, accomplished using various ensemble methods in transformer-based models. These methods entail combining scores obtained from each transformer by taking their average or concatenating the vectors obtained from the transformers. Furthermore, we streamline the complexity of the method by utilizing dimensionality reduction techniques after vector concatenation. In one of the proposed combined methods, rather than relying on cosine similarity, we determine the similarity of sentence pair vectors using a trained neural network and another technique for vector connection.

The primary argument presented in this article is that two sentences that possess semantic similarity must have vectors that are in close proximity to one another in vector space, allowing their semantic similarity to be assessed using various criteria such as the cosine similarity criterion. The main objective of this article is to obtain sentence vectors that effectively capture the semantic complexity of sentences and accurately position them in vector space. We enhanced the results obtained from our combined methods by fine-tuning a new Persian-English language pair. Furthermore, we did not rely on a translation machine to determine cross-lingual sentence similarity. In summary, the innovations introduced in this article include:

1. Identifying semantic textual similarity between languages without relying on machine translation or translating one sentence to another.
2. Utilizing an ensemble method to detect cross-lingual similarities.
3. Establishing a shared vector space between Persian and English languages.
4. Simplifying the model by implementing dimension reduction techniques.

2 Related works

This section discusses the previous methods used to determine sentence similarity in different languages. Cross-lingual sentence similarity methods can be classified into two categories. The first category involves converting sentences into a single language using machine translation, followed by using monolingual similarity methods to estimate similarity. The second category consists of methods that do not utilize machine translation and determine sentence similarity by employing a shared vector space.

Douma and Menzel [9] introduced an unsupervised method based on the paragraph vector (SEF@UHH). In their approach, sentences are represented as vectors using Paragraph Vector-Distributed Bag of Words (PV-DBOW). The model is trained on various languages and utilizes multiple features to enhance its predictive power.

To use this method for identifying sentence similarity (L1-L2), the L1 language is first translated into L2 using the Google translation machine, and the model trained on the L2 language is used to replace the sentence vector. Then, the L2 language is translated into L1, and the model trained on the L1 language is used to create the sentence vector. In each case, the similarity of the sentences is calculated, and the average of the two similarity scores is taken to obtain the final score.

In these methods, the Pearson correlation between the predicted scores and the golden scores is used as the evaluation criterion. Additionally, another similarity detection criterion is utilized in addition to the cosine similarity mentioned in equation 1, where u and v represent sentence vectors.

$$\text{Formula1. Bray-Curtis: } \frac{\sum|u_i-v_i|}{\sum|u_i+v_i|}$$

The method proposed by Douma and Menzel [9] has demonstrated superior performance in the Spanish-English language pair. The results of their study indicate that the Bray-Curtis criterion can be a suitable metric for detecting similarity. Among the 7 language pairs examined, this criterion outperformed the cosine similarity metric in 5 language pairs.

Tian et al. [10] presented the ECNU ensemble method, which utilizes a combination of traditional natural language processing methods and neural networks to extract features. The final similarity score is obtained by averaging the scores obtained from traditional natural language processing and neural networks. This method achieved the best score in finding the similarity of cross-lingual sentences in the Arabic-English and English-Turkish language pairs in the SemEval2017 dataset. In this approach, sentences are first translated into English, and then the semantic similarity between two English sentences is calculated.

The CompiLIG group presented two different implementations that were evaluated on the SemEval2017 evaluation data, with the best implementation achieving the highest score in the Spanish-English cross-language evaluation data.

In unsupervised methods based on shared semantic space [12], three steps are taken to calculate cross-lingual semantic similarity. First, word vector substitution is performed using n-gram characters based on the skipgram model [13]. In the second step, a common semantic space is created for two languages by using linear transformation and obtaining a matrix, and the word vectors are transferred to the common semantic space. To obtain this linear transformation matrix, various methods have been proposed, which are discussed in detail below. In the last stage, three different methods can be used to obtain the sentence embedding vector through the semantic vector of the word and calculate the similarity of sentences.

One of the main issues with many of the previous methods, such as the ECNU ensemble method, is the requirement for parallel resources and the construction of a translation machine to translate one of the sentences into the language of the second sentence, followed by utilizing monolingual similarity methods to determine the degree of similarity [14].

There are also methods that, although they do not require a translation machine, use parallel resources to build cross-lingual vector representations of words to perform cross-lingual similarity at the word level [15].

The method proposed by Luo and Simard [16] eliminates the need for cross-lingual parallel resources or translation machines. Vector placement is performed using models and the context of words, and cross-lingual similarity search at the sentence level is achieved using the introduced criterion, YiSi-2. YiSi-2 identifies alignments between words in two sentences that maximize semantic similarity. If f and e are the words in sentences f' and e', respectively, which belong to the vocabulary set of languages F and E, and the vector representation of these words is represented by v(f) and v(e), the similarity of the two sentences can be calculated using equations 1 to 5.

$$s(e.f) = \cos\big(v(e).v(f)\big)(1\text{-}1)$$

$$\begin{aligned} w(e) &= idf(e) = \log\left(1 + \frac{|E|+1}{|E_{\exists e}|+1}\right) \\ w(f) &= idf(f) = \log\left(1 + \frac{|F|+1}{|F_{\exists f}|+1}\right) \end{aligned} \qquad (2\text{-}1)$$

$$precision = \frac{\sum_{e\in e'} \max_{f\in f'} w(e)\cdot s(e.f)}{\sum_{e\in e'} w(e)}(3\text{-}1)$$

$$recall = \frac{\sum_{f\in f'} \max_{e\in e'} w(f)\cdot s(e.f)}{\sum_{f\in f'} w(f)}(4\text{-}1)$$

$$YiSi - 2 = \frac{2\cdot precision\cdot recall}{precision+recall}(5\text{-}1)$$

In formulas (1-1) to (5-1), if the words f and e are the words in the sentences f' and e', respectively, which are in the vocabulary set of languages F and E, and are replaced with v(f) and v(e), respectively. s(e, f) represents the cosine similarity between the two vectors v(f) and v(e).

In this method, cross-lingual word vector representation involves the use of two supervised models, BiSkip and vecmap. BiSkip [17] learns bilingual representations from common contextual information in monolingual data and semantic equivalence in parallel data. In vecmap [18], training is performed on a single vector representation using word2vec on monolingual sources, followed by linear transformation training of two monolingual vectors using vecmap and the dictionaries available in each language pair. Different pre-trained languages are used as the third model, and the results are compared.

These models rely on parallel data, which can be obtained from sources such as the Europarl corpus or the United Nations Parallel Corpus, making them dependent on the availability of such resources. However, they can produce high-quality cross-lingual word vectors that can be used to determine the similarity between words or sentences in different languages.

Among the described methods, the ECNU combined method has demonstrated the best performance in detecting the similarity of cross-lingual sentences in most language pairs. It is one of the methods based on machine translation, and so far, no method has been presented that has better performance than it, especially among the methods based on machine translation.

One advantage of the ensemble method compared to other methods is the use of a variety of models and algorithms for sentence representation and the use of various methods, each of which provides a part of the requirements for finding the similarity of sentences. However, the weakness of this method is the use of machine translation, and unlike many other methods, it does not have the ability to find similarities between cross-lingual sentences directly and depends on machine translation. Multilingual and transformer-based models can be used to improve methods that do not use machine translation.

In the method based on the substitution of sentences through the paragraph vector (SEF@UHH), it has been observed that using other vector similarity detection criteria, such as Bray-Curtis, in addition to cosine similarity can sometimes improve performance and accuracy in estimating the similarity of sentences.

3 Proposed Approach

It is true that among the methods proposed and tested, machine translation-based methods have shown relative superiority in finding cross-lingual semantic similarity. However, this is considered a drawback as it always requires machine translation to find similarities between sentences, and the main weakness of such systems is the propagation of machine translation errors. Moreover, the translation of the source language into the target language may not be done accurately.

It should be noted that the previous methods used transformers and language models based on BERT less frequently. By examining the methods, it can be concluded that ensemble methods perform better than methods based on a specific model in the issue of similarity of cross-lingual sentences.

In the method presented in this article, language models based on transformers are used in an ensemble way, without the need for machine translation. This approach not only utilizes the positive aspects of previous methods but also addresses their negative aspects.

Another point to consider in ensemble methods is that the combination of models can either be in the final score resulting from the use of transformers or in their vector space (e.g., concatenating vectors). Additionally, fine-tuning the transformers can further improve the accuracy of sentence similarity. Overall, ensemble methods that combine multiple techniques and models, including transformers, have shown promise in improving the accuracy of cross-lingual sentence similarity detection.

3-1 Choosing suitable models for combination

As mentioned in the introduction of this section, the proposed method uses a combination of models based on transformers. In this section, we discuss how to choose these models.

To select suitable models for our proposed method, we used the Persian-English dataset presented in [2] and evaluated the models using test data. The results are shown in Table 1. These models are based on transformers and are available on Hugging Face.

Table1 Evaluation of transformer-based models for cross-lingual sentence similarity detection on the Persian-English dataset

	Models	Pearson Correlation
1	Xlm-roberta base	23.80
2	**paraphrase-xlm-r-multilingual-v1**	**86.87**
3	bert-base-multilingual-cased	47
4	distilbert-base-multilingual-cased	45.56
5	stsb-xlm-r-multilingual	78.37
6	xlm-r-100langs-bert-base-nli-stsb-mean-tokens	78.27
7	**mpnet-base-v2**	**85.10**
8	distiluse-base-multilingual-cased-v2	82.85
9	paraphrase-multilingual-MiniLM-L12-v2	80.44
10	**mstsb-paraphrase-multilingual-mpnet-base**	**83.62**

Based on the results obtained from Table 1, we selected models No. 7 (mpnet-base-v2), 2 (paraphrase-xlm-r-multilingual-v1), and 10 (mstsb-paraphrase-multilingual-mpnet-base) for combination. We chose three models because using more than this number does not significantly improve the results, and adding more models increases the complexity of the method. Therefore, selecting a maximum of three models is sufficient.

As shown in Table 1, the XLM-RoBERTa base model has weaker performance compared to the other models. This is likely due to the basic nature of this model and its lack of training on other datasets. However, incorporating it into the ensemble model may still provide useful information and improve overall performance.

3-2 Fine tuning of selected models

The fine-tuning phase in the STS task is an operation performed to improve the performance of transformers for a specific language pair using a cross-lingual corpus. During the fine-tuning operation, the weights of the last layer of transformers are updated using training data, and the vector space of sentences can be adjusted to increase the semantic richness of the representations produced by these models, leading to improved performance in semantic sentence similarity.

In our experiments, we fine-tuned the selected models using the PESTS [2] dataset to find similarities between Persian and English sentences. We fine-tuned the models using the training data of the semantic similarity corpus and improved their performance compared to the condition without fine-tuning, as shown in Table 2.

The fine-tuning operation was performed in four epochs using a batch size of 32, and cosine similarity was used as the loss function. This process involved training the last layer of the pre-trained transformer models on the PESTS dataset to adapt them to the specific task of finding cross-lingual sentence similarity. The fine-tuning process helps the models to better capture the semantic similarity between sentences in the target language pair, leading to improved performance.

3-3 Similarity prediction with feed forward neural network

In our combined method, we use a neural network instead of using cosine similarity to calculate the similarity score, as shown in Figure 1. The input of this network is a vector for each pair of sentences, and its output is a continuous number based on the similarity for each pair of sentences.

As shown in Figure 1, the similarity of the vectors can be modeled by using the multiplication of the vectors of sentences 1 and 2 (similar to the cosine similarity multiplication formula), and the distance and difference of these two vectors in the vector space can be calculated by using the subtraction of these vectors. To select the input vector of the neural network, other experiments have been performed, such as using the concatenation of vectors 1 and 2. However, we found that using the operation of multiplication and subtraction of vectors and the final concatenation of the resulting two vectors resulted in better training of the neural network for estimating the similarity score.

In this network, eight layers are used, which are trained in 150 epochs with a batch size of 50. Hyperparameters, such as the number of neuron units in each layer, are set accordingly. The neural network is trained using the fine-tuned transformer models to estimate the similarity score between sentence pairs in the target language pair.

Overall, using a neural network to estimate the similarity score allows us to better capture the nuances of cross-lingual sentence similarity and can provide more accurate results compared to using traditional similarity metrics such as cosine similarity.

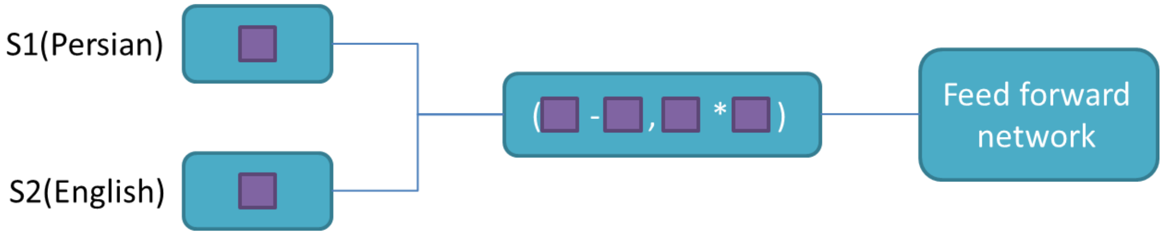

Fig.1 Calculating the similarity of sentence pairs using neural network

3-4 Architecture of ensemble methods

In this section, we describe three ensemble methods for calculating semantic sentence similarity, and each method combines transformers in a different way. These three methods provide different ways of combining the information from different transformers to improve the accuracy of cross-lingual sentence similarity detection. These methods are as follows:

Method 1: As shown in Figure 2, in this method, after fine-tuning the transformers, we obtain the vector representation of English and Persian sentences using each of the transformers and calculate the cosine similarity between sentences for each pair of sentences. To combine the results, we use the average score of similarity obtained for each pair of sentences by different transformers. Specifically, we calculate the average among the three scores obtained for each pair of sentences, resulting in one score for each pair of sentences. Finally, we calculate the Pearson or Spearman correlation between the obtained scores and the golden scores.

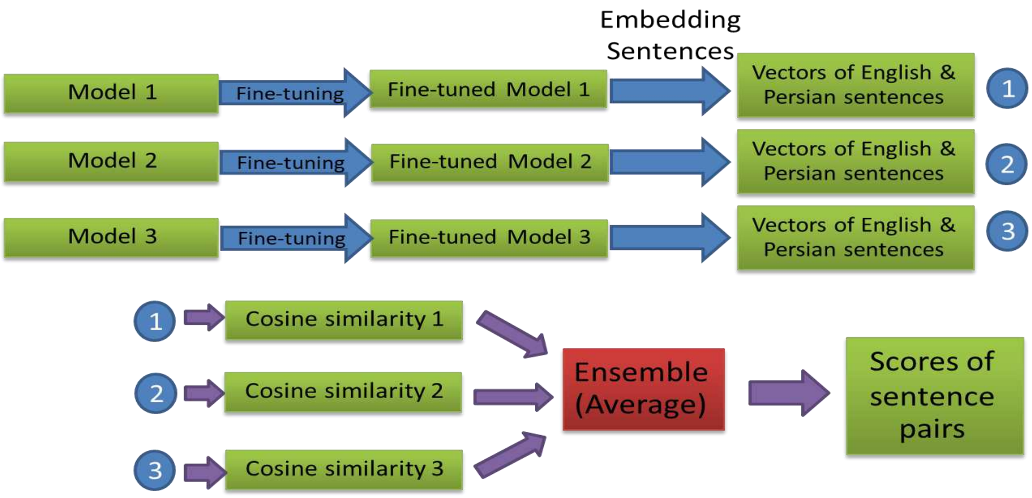

Fig.2 Architecture of ensemble method 1

Method 2: As shown in Figure 3, after fine-tuning the transformers, we obtain the embedding of English and Persian sentences using each of the transformers. In this method, for each sentence, the vectors obtained from the three transformers are concatenated to form a single vector with 2304 (3 x 768) dimensions. Then, the dimensionality of the concatenated vectors is reduced to 350 dimensions using the principal component analysis technique, as shown in Figure 4. Finally, we calculate the cosine similarity between each pair of sentences and calculate the Pearson or Spearman correlation between the obtained scores and the golden scores.

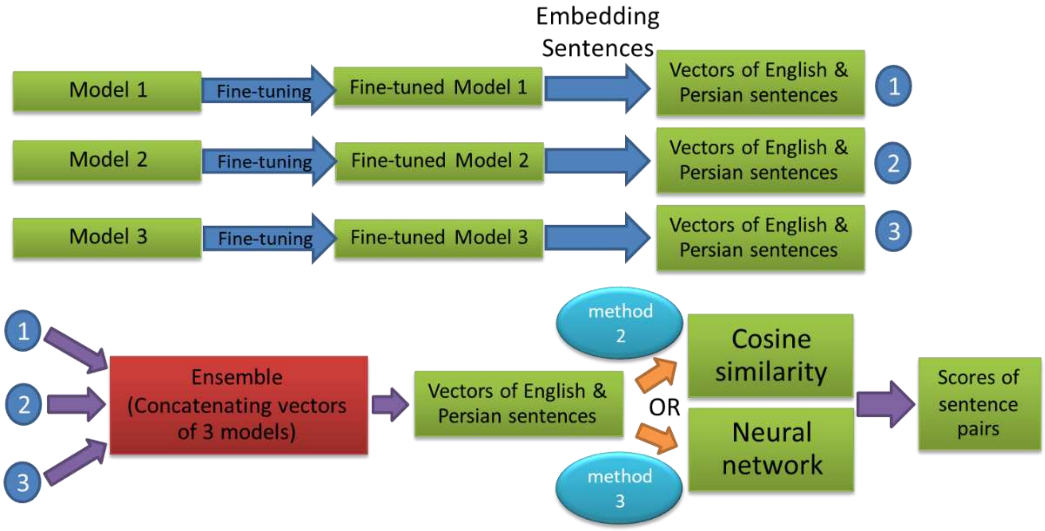

Fig.3 Architecture of ensemble method 2 and 3

Method 3: As shown in Figure 3, this method is similar to Method 2 in terms of using vector concatenation for combination, but instead of using cosine similarity, a neural network is used to calculate the similarity score. To calculate the similarity of the sentences, we perform the multiplication operation between the elements of the pair of sentences (the sentence of language 1 and the sentence of language 2) and also subtract them, resulting in two vectors. By concatenating these two vectors, a vector with 4608 dimensions is obtained for each pair of sentences (further details are given in section 3.3). This resulting vector is used as input to a neural network, and a probability based on similarity is output. The neural network is trained by minimizing the difference between the obtained score and the label score, using the training data, and evaluated using the test data. We calculate the Pearson or Spearman correlation between the obtained scores and the golden scores.

As mentioned, in Methods 2 and 3, the concatenation of the vectors obtained from the three selected transformers is used for the combination operation. Considering that the sentence vector has 768 dimensions, to reduce the complexity, it is possible to use dimensionality reduction techniques (such as principal component analysis) after concatenating these vectors. However, this technique was not used in Method 3 due to its negative effect on the learning of the neural network, which greatly reduced the final performance.

In combined methods, creating diversity among different models is crucial, and this diversity can be achieved through the models themselves, their training, or their training data. In the combined methods presented in this article, we created diversity in the training data by using one-third of the training data to fine-tune each transformer.

4 Experiments

In the previous section, we used three cutting-edge transformer models, namely paraphrase-xlm-r-multilingual-v1, mstsb-paraphrase-multilingual-mpnet-base, and mpnet-base-v2, in the combined method. As presented in Table 2, the evaluation results for the Persian-English language pair show that the third combined method performs better than the other two methods. However, it is more complicated than the other two methods, which is one of its drawbacks.

The evaluation results in Table 2 show that even the combined methods without fine-tuning outperform the individual models. This means that different base transformers excel in specific parts of the vector space, and their combination can lead to improved performance. By fine-tuning the model, we can further improve its performance.

Table 2 The evaluation results in Persian – English

	methods	Pearson Correlation
	[20] paraphrase-xlm-r-multilingual-v1	85.87
	[21] multilingual-mpnet-base	85.10
	mstsb-paraphrase-multilingual-mpnet-base	83.62
Without fine-tuning	Ensemble method 1	87.93
	Ensemble method 2	87.43
	Ensemble method 3	93.57
With fine-tuning	Ensemble method 1	94.54
	Ensemble method 2	93.79
	Ensemble method 3	**95.70**

To determine the optimal number of dimensions to reduce in combined Method 2, we performed a Pearson correlation analysis with different numbers of dimensions, as shown in Figure 4. By reducing the dimensions in the vectors to 350 dimensions, the maximum Pearson correlation was obtained. It should be noted that the principal component analysis technique was used for dimension reduction, up to a maximum of 350 dimensions.

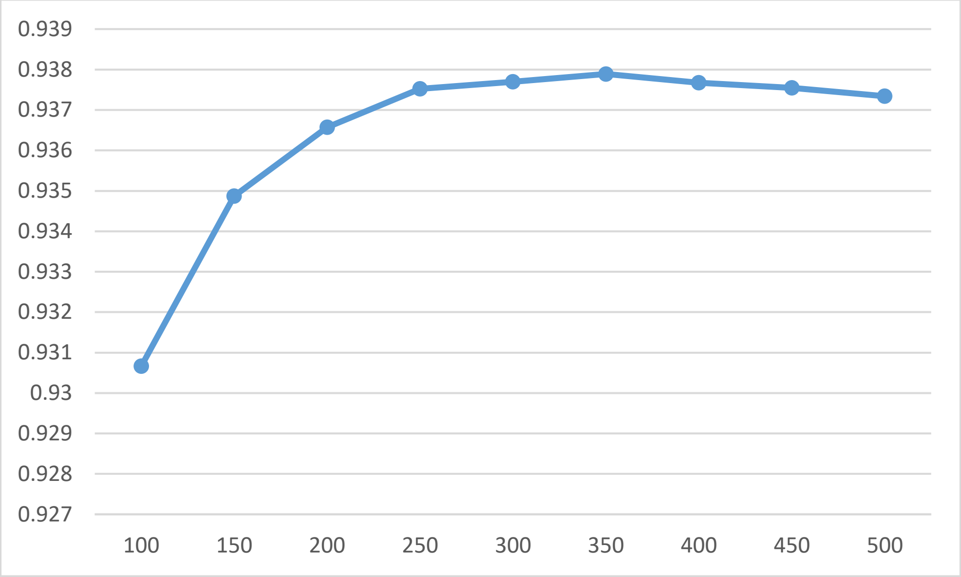

Fig.4 Pearson correlation in different dimensions of vectors

To adjust the number of layers in the neural network of combined Method 3, we measured the Pearson correlation in different modes, as shown in Figure 5, to determine the optimal number of layers. As we can see in Figure 5, increasing the complexity of the network to more than 8 layers did not result in a significant change in Pearson's correlation. Therefore, we selected 8 layers to avoid excessive complexity of the neural network and achieve the best performance.

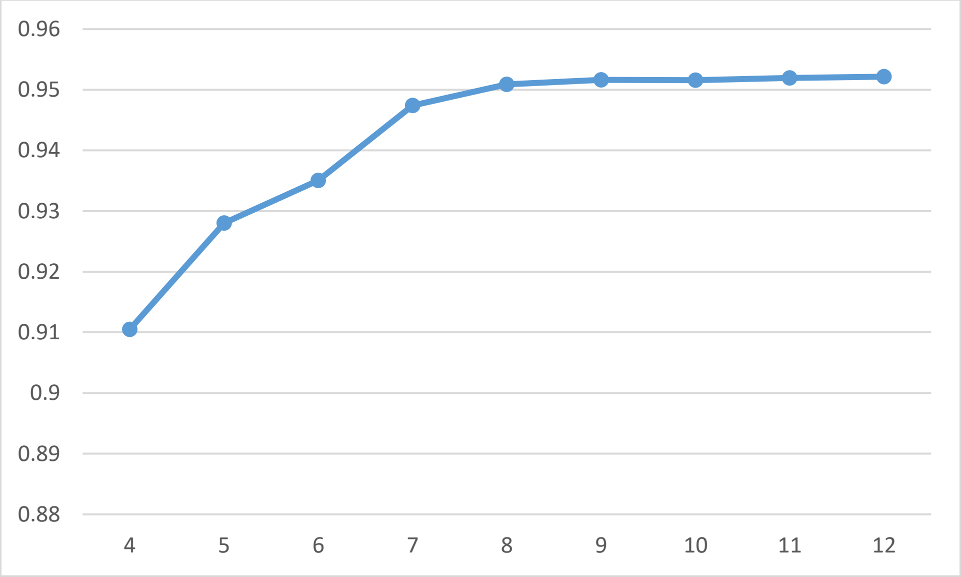

Fig.5 Adjusting the number of neural network layers in ensemble method 3

As we know, the input to the neural network in combined Method 3 is a vector with 4608 dimensions. Therefore, the first layer of the neural network will have 4608 neurons, and we want a single output neuron for similarity score. For the number of neurons in each hidden layer, we selected the optimal configuration based on Figure 6. It should be noted that the number of layers in the experiments of Figure 6 was set to 8, as determined from Figure 5. The number of neurons in each layer is shown in Table 3:

Table 3 number of neurons in different layers in different experiments

	Number of neurons							
Experiment No.	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5	Layer 6	Layer 7	Layer 8
1	4608	4608	2304	2304	1000	1000	700	1
2	4608	4608	2304	2304	2304	1000	700	1
3	4608	4608	2304	2304	1000	1000	32	1
4	4608	4608	2304	2304	1000	700	32	1
5	4608	4608	2304	1000	1000	700	32	1
6	4608	2304	2304	1000	1000	700	32	1
7	4608	2304	2304	1000	700	700	32	1
8	4608	2304	1000	1000	700	700	32	1

According to the results obtained from Figure 6, we selected Experiment 7 as the final neural network configuration. However, there was no significant difference between the experiments, with results ranging from 0.957 to 0.951 percent. Nevertheless, the maximum Pearson correlation was obtained by Experiment 7.

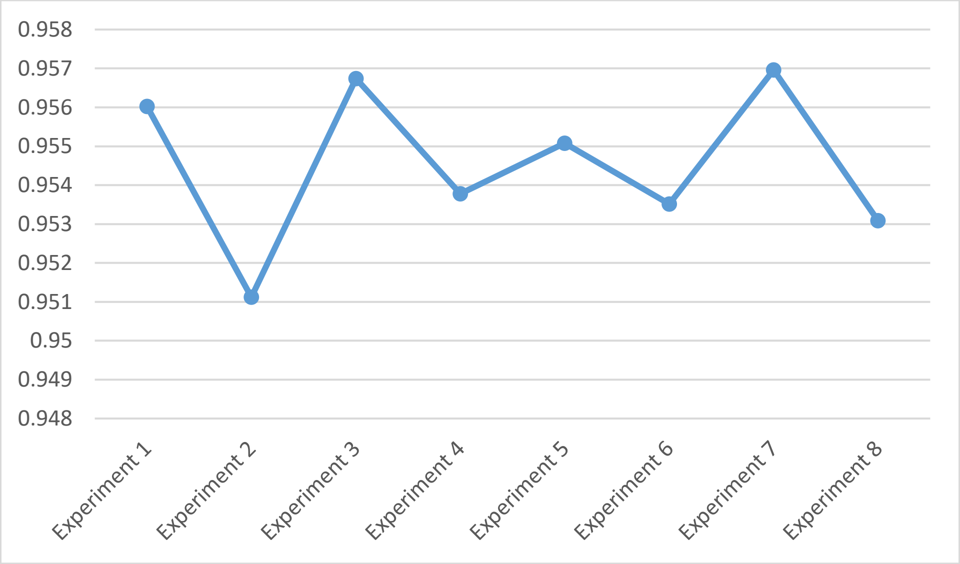

Fig.6 Adjusting the number of neural network neurons in the last phase of ensemble method 3

Although the main focus of this article is on Persian-English semantic sentence similarity, we also evaluated our method on the Arabic-English language pair using the Semeval2017 dataset, as shown in Table 4. Our method achieved the best performance among the previous methods evaluated on this dataset.

As shown in the comparison between Tables 2 and 4, the performance of the methods varies across different language pairs, and the quality and quantity of training data available for each language pair play an important role in the accuracy of the methods. In the case of Persian-English, we addressed this issue by providing a dataset, and this process can be applied to other language pairs as well.

The reason for the weaker performance of combined Method 3 compared to the other two combined methods in the Arabic-English language pair is the lower quality and quantity of data available for this language pair, compared to the Persian-English language pair. This affects the neural network training operation in the Arabic-English language pair, which may not perform well in estimating the similarity score of sentence pairs.

Table 4. The evaluation results in Arabic – English

Year	Method	Pearson Correlation
2023	Ensemble method 1	**84.04**
2023	Ensemble method 2	83.16
2023	Ensemble method 3	83.02
2018	The best method of the past (ECNU) [10]	74.93
2020	ORT-OM-IDF [12]	68.5
2017	SEF@UHH [9]	53.84

For the fine-tuning operation, pairs of training sentences were used to update the weights of the last layer of transformers. The settings for this operation are listed in Table 5.

Table 5. Settings of Transformers Fine-tuning

Objective function	Cosine Similarity Loss
Epochs	4
Batch Size	32

It should be noted that for combined Method 3, a neural network is used to estimate the degree of similarity. The settings for the neural network are listed in Table 6, including hyperparameters such as the number of neuron units in each layer. In the experiments, the Adam optimization function was used for the neural network, but the results obtained from the RMSprop optimization function were slightly better. Therefore, we used the RMSprop function for the final evaluation. The mean square error between the estimated score and the golden score was used as the error function.

Table 6. Settings of Neural Network

Activation Function	Relu
Optimization function	RMSprop
Loss Function	Mean Squared Error
Epochs	20
Batch Size	50
Number of neurons in the input layer	4608
Number of neurons in the hidden layer	7040
Number of neurons in the output layer	1

5 Conclusion

As previously mentioned, the task of finding semantic similarity between sentences is language-dependent, and the available dataset for fine-tuning plays a crucial role in the performance of these methods.

By selecting and combining three transformer-based models, we were able to achieve better performance than the basic models. Furthermore, by limiting the number of models used, we controlled the possible complexity of the methods, which is another advantage compared to other combination methods. Based on evaluations in the Persian-English language pair, Method 3 showed the best performance among the other two methods, but its use of a forward neural network makes it more complex than the other two methods, which is a disadvantage.

By fine-tuning transformer-based models, we adjusted the weights of their last layers, which improved their performance in representing sentence vectors and finding similarity between cross-lingual sentences. The introduced combined methods were able to address the limitations of previous methods without requiring machine translation and reducing complexity compared to previous combination methods.

Considering that this article discusses sentences in two different languages, a challenge was the lack of a shared vector space between these languages. This issue was addressed by using models based on multilingual transformers and adjusting them for this purpose. Applications for cross-lingual sentence similarity include their use in machine translation and information retrieval systems and search engines. Future work can focus on applying the proposed method to improve performance, especially for the Persian language.

Statements and Declarations

Data availability

The experiments and results obtained in this paper are based on the PESTS dataset [2]. The PESTS dataset is available on GitHub(https://github.com/mohammadabdous/PESTS), which contains 5375 sentence pairs.

Funding

This research received no grant from any funding agency.

Competing interests

The authors declare no competing interests.

Authors

Poorya Piroozfar received his M.S. degree in computer Engineering from the University of Science and Technology, Iran, in 2022. His research interests include Natural Language Processing, Data Mining and Deep Learning. His supervisor is Dr. Behrouz Minaei Bidgoli in University of Science and Technology.

Google Scholar: https://scholar.google.com/citations?user=FAM6rOUAAAAJ&hl=en&oi=sra

Mohammad Abdous received his M.S. degrees from the University of Science and Technology, in 2016. From 2016 until now he was Researcher at mentioned Institute and worked on the high-level Natural Language Processing tools. Text mining and natural language processing are his interests.

Google Scholar: https://scholar.google.com/citations?user=3ZMgrdcAAAAJ&hl=en&oi=sra

Behrouz Minaei Bidgoli is a Professor and the head of the Computer Engineering School at Iran University of Science and Technology. He led the Data Mining Lab (DML) that does research on various areas in artificial intelligence and data mining, including text mining, web information extraction, and natural language processing.

Google Scholar: https://scholar.google.com/citations?user=M8tgU-wAAAAJ&hl=en&oi=sra

References

[1] Q. Le and T. Mikolov, “Distributed representations of sentences and documents,” in *International conference on machine learning*, 2014, pp. 1188–1196.

[2] M. Abdous, P. Piroozfar, and B. Minaei, “*PESTS :Persian_English Corpus for Cross Language Semantic Textual Similarity*“. *arXiv Prepr. arXiv* 2305.07893.2023.

[3] G. Majumder, P. Pakray, A. Gelbukh, and D. Pinto, “Semantic textual similarity methods, tools, and applications: A survey,” *Comput. y Sist.*, vol. 20, no. 4, pp. 647–665, 2016.

[4] N. Limbasiya and P. Agrawal, “Semantic Textual Similarity and Factorization Machine Model for Retrieval of Question-Answering,” in *International Conference on Advances in Computing and Data Sciences*, 2019, pp. 195–206.

[5] E. Comelles and J. Atserias, “VERTa: A linguistically-motivated metric at the WMT15 metrics task,” in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 2015, pp. 366–372.

[6] Y. Chouni, M. Erritali, Y. Madani, and H. Ezzikouri, “Information retrieval system based semantique and big data,” *Procedia Comput. Sci.*, vol. 151, pp. 1108–1113, 2019.

[7] R. Rodrigues, P. Couto, and I. Rodrigues, “IPR: The Semantic Textual Similarity and Recognizing Textual Entailment Systems.,” in *ASSIN@ STIL*, 2019, pp. 39–48.

[8] A. Mahmoud and M. Zrigui, “Semantic similarity analysis for corpus development and paraphrase detection in arabic.,” *Int. Arab J. Inf. Technol.*, vol. 18, no. 1, pp. 1–7, 2021.

[9] M.-S. Duma and W. Menzel, “SEF@ UHH at SemEval-2017 Task 1: Unsupervised knowledge-free semantic textual similarity via paragraph vector,” in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 170–174.

[10] J. Tian, Z. Zhou, M. Lan, and Y. Wu, “ECNU at SemEval-2017 Task 1: Leverage Kernel-based Traditional NLP features and Neural Networks to Build a Universal Model for Multilingual and Cross-lingual Semantic Textual Similarity,” pp. 191–197, 2018, doi: 10.18653/v1/s17-2028.

[11] J. Ferrero, F. Agnes, L. Besacier, and D. Schwab, “CompiLIG at SemEval-2017 Task 1: Cross-language plagiarism detection methods for semantic textual similarity,” *arXiv Prepr. arXiv1704.01346*, 2017.

[12] T. Brychcín, “Linear transformations for cross-lingual semantic textual similarity,” *Knowledge-Based Syst.*, vol. 187, p. 104819, 2020.

[13] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 135–146, 2017.

[14] T. Brychcín and L. Svoboda, “UWB at SemEval-2016 Task 1: Semantic textual similarity using lexical, syntactic, and semantic information,” in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 588–594.

[15] C. Lo, M. Simard, D. Stewart, S. Larkin, C. Goutte, and P. Littell, “Accurate semantic textual similarity for cleaning noisy parallel corpora using semantic machine translation evaluation metric: The NRC

supervised submissions to the parallel corpus filtering task," in *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 2018, pp. 908–916.

[16] C. Lo and M. Simard, "Fully unsupervised crosslingual semantic textual similarity metric based on BERT for identifying parallel data," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019, pp. 206–215.

[17] M.-T. Luong, H. Pham, and C. D. Manning, "Bilingual word representations with monolingual quality in mind," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, 2015, pp. 151–159.

[18] M. Artetxe, G. Labaka, and E. Agirre, "Learning principled bilingual mappings of word embeddings while preserving monolingual invariance," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 2289–2294.

[19] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual BERT?," *arXiv Prepr. arXiv1906.01502*, 2019.

[20] A. Conneau *et al.*, "Unsupervised cross-lingual representation learning at scale," *arXiv Prepr. arXiv1911.02116*, 2019.

[21] K. Song, X. Tan, T. Qin, J. Lu, and T. Y. Liu, "MPNet: Masked and permuted pre-training for language understanding," *Advances in Neural Information Processing Systems*, vol. 2020-Decem. 2020.