


A BERT-GRU Model for Measuring the Similarity of Arabic Text


Rakia Saidi

(LIMTIC Laboratory, UTM University, Tunis, Tunisia)

 <https://orcid.org/0000-0003-0798-4834>, saidi.rakya@gmail.com


Fethi Jarray

(ISI Medenine, Gabes University, Medenine, Tunisia)

 <https://orcid.org/0000-0003-2007-2645>, fjarray@gmail.com

Didier Schwab

(LIG Laboratory, Univ. Grenoble Alpes, France)

 <https://orcid.org/0000-0002-2462-8148>, schwabd@univ-grenoble-alpes.fr

Abstract: Semantic Textual Similarity (STS) aims to assess the semantic similarity between two pieces of text. As a challenging task in natural language processing, various approaches for STS in high-resource languages, such as English, have been proposed. In this paper, we are concerned with STS in low resource languages such as Arabic. A baseline approach for STS is based on vector embedding of the input text and application of similarity metric on the embedding space. In this contribution, we propose a cross-encoder neural network (Cross-BERT-GRU) to handle semantic similarity of Arabic sentences that benefits from both the strong contextual understanding of BERT and the sequential modeling capabilities of GRU. The architecture begins by inputting the BERT word embeddings for each word into a GRU cell to model long-term dependencies. Then, max pooling and average pooling are applied to the hidden outputs of the GRU cell, serving as the sentence -pair encoder. Finally, a softmax layer is utilized to predict the degree of similarity. The experiment results show a Spearman correlation coefficient of around 0.9 and that Cross-BERT-GRU outperforms the other BERT models in predicting the semantic textual similarity of Arabic sentences. The experimentation results also indicate that the performance improves by integrating data augmentation techniques.

Keywords: Semantic Similarity, Cross-Encoder, Data augmentation, Arabic text, GRU, BERT, Backtranslation

Categories: H.3.1, H.3.2, H.3.3, H.3.7, H.5.1

DOI: 10.3897/jucs.111217

1 Introduction

Semantic Textual Similarity (STS) is a crucial natural language processing task dedicated to gauging the similarity between two given texts or sentences [Hliaoutakis et al. 2006]. Its practical applications span various domains, including information retrieval [Hliaoutakis et al. 2006], semantic web, plagiarism detection [Al-Shamery and Ghenni 2016], machine translation [Wieting et al. 2019], document clustering, word sense disambiguation [Murad et al. 2010], and question answering [Almiman et al. 2020]. The common thread among these applications is the computation of textual document similarity. Moreover, STS

serves as a metric for automatic summarization by gauging the similarity between generated and reference summaries.

At the heart of an STS system lies sentence embedding, with early methods predominantly grounded in traditional machine learning and manually engineered features [Hliaoutakis et al. 2006]. However, these approaches faced challenges, yielding suboptimal performance due to the inherent language ambiguity and intricate sentence structures. This challenge, often termed the gap between low-level features and high-level semantics, has seen a transformative shift with the advent of word embedding and the ascendancy of deep neural networks in STS and broader natural language processing (NLP) contexts. This paper delves into a specific category of deep neural networks known as cross-encoders known for their adeptness in learning semantic similarity, particularly in STS applications.

The essence of sentence similarity manifests as a multiclassification problem, in which each class denotes a score or level of similarity [Almiman et al. 2020]. While numerous traditional algorithms have been employed to train classifiers on manually annotated corpora, their effectiveness is overshadowed by the consistent outperformance of deep neural networks in various NLP tasks. Notably, deep contextualized models such as BERT (Bidirectional Encoder Representations from Transformers) have demonstrated prowess in tasks like semantic similarity of Arabic questions [Sa'ad et al. 2021, Al-Bataineh et al. 2019]. This paper explores the application of BERT to assess the semantic similarity of Arabic sentences, leveraging prominent Arabic BERT models [Sa'ad et al. 2021, Al-Bataineh et al. 2019].

In the realm of sentence matching, two prominent architectures—Bi-encoder and cross-encoder—stand out [Humeau et al. 2019]. A Bi-encoder comprises two independent branches sharing the same infrastructure and weight, each processing sentences separately. Conversely, a Cross-encoder simultaneously processes both sentences, matching their tokens and aggregating results through an additional network for final decision-making. Notably, Humeau et al. [Humeau et al. 2019] substantiate the superior performance of cross-encoders over bi-encoders in sentence scoring tasks. In alignment with this evidence, our contribution adopts a cross-encoder approach.

The primary contributions of this paper are summarized as follows:

- Introduction of a cross-encoder neural network for predicting sentence pair similarity.
- Augmentation of the existing Arabic Semantic Text Similarity dataset through backtranslation techniques.
- Integration of GRU layer and BERT model
- Evaluation of the proposed approach performance across three benchmark datasets.

The subsequent sections are organized as follows: Section 2 provides a comprehensive survey of the current state of the art. Section 3 explains our approach, and Section 4 details results and discussions. The paper concludes by summarizing contributions and outlining potential avenues for future extensions.

2 Related work

Semantic similarity can be measured in documents, sentences, and words. Measurement of semantic similarity between Arabic sentences is a challenge in terms of human understanding. Related approaches for semantic similarity of Arabic sentences can be classified

into (1) classical approaches and (2) machine learning (ML) based approaches. Early works on STS rely on hand-crafted features. Alzahrani [Alzahrani 2016] addressed the problem of semantic similarity by examining the semantic similarities between Arabic and English in short phrases and sentences. From a monolingual perspective, dictionary and machine translation techniques were used to determine the relatedness of cross-lingual texts. Three algorithms for determining semantic similarity were created and deployed to the human-rated benchmark. Using the term sets generated by the dictionary-based approach, an averaged maximum translation similarity algorithm was presented. The semantic similarity could also be calculated using noun-verb and term vectors produced using the Machine Translation (MT) approach. The highest correlation ($r = 0.8657$) was found using the MT-based word vector semantic similarity method, which was followed by the averaged maximum translation similarity algorithm ($r = 0.7206$).

Li et al. [Li et al. 2006] used machine translation to determine the similarity of the phrase vectors, where the similarity score is derived from the cosine similarity between two term vectors. The entry value in the term vector is the maximal semantic similarity between the relevant word and other sentence words. The results reveal that the term vector similarity algorithm based on machine translation achieves a correlation of 0.86.

Abd Alameer [Abd Alameer 2017] used a hybrid similarity measures strategies to find similarities between two Arabic texts: semantic similarity measure, cosine similarity measure, and N-gram similarity measure (using the Dice similarity measure). To detect similar character sequences, the authors performed cosine and N-gram similarity measurements.

Recent methods employ textual features of deep learning models, such as recurrent neural networks. Nagoudi and Schwab [Nagoudi and Schwab 2017] investigated the semantic similarity of Arabic phrases using word embedding, a proposed technique for word alignment, and various weighting algorithms for vector words. The sum of the vectors of the sentence's content terms is the sentence's vector. They tested their method with Arabic-translated. Microsoft Research Video Definition Corpus (MSRvid).

Alian and Awajan [Alian, and Awajan 2021] proposed a method that uses lexical, semantic, and syntactic-semantic variables, as well as linear regression and support vector machine regression, to quantify sentence similarity. Experiments are carried out on two datasets from SemEval 2017 (Arabic paraphrasing benchmark, MSRvid, and SMTeuroparl) to evaluate the performance of this technique. This method achieves a correlation of 0.354 when applied to the Arabic paraphrasing benchmark, while it achieves 0.743 and 0.467 on the MSRvid and SMTeuroparl datasets, respectively.

The most recent approaches have taken advantage of modern contextualized word embeddings. Alsaleh et al. [Alsaleh et al. 2021] used AraBERT to categorize pairs of verses from the QurSim dataset as semantically related or unrelated. They preprocessed the QurSim dataset and divided it into three comparison data sets. To determine which version of AraBERT performs best with the specified datasets, they used mBERT and AraBERTv2. AraBERTv0.2 produced the best results, with 92% accuracy on a dataset that included label '2' and label '-1', so they used only two classes of similarity, the latter of which was constructed outside the QurSim dataset, but in this work they investigated only the AraBERT model, not the other versions of the Arabic BERT model.

Gabr et al. [Gabr et al. 2023] proposed a approach that used BERT and marBERT model to calculate the measure of similarity for arabic short text.

Saidi et al. [Saidi et al. 2023] and Alshammeri et al. [Alshammeri et al. 2021] proposed a siamese networks for the arabic text semantic similarity. For [Saidi et al. 2023], The authors investigated the most available Arabic BERT models to embed the input sentences. they validated this approach via Arabic STS datasets [Arabic STS

corpus]. The araBERT-based Siamese Network model achieves a Pearson correlation of 0.925. The results obtained demonstrate the superiority of integrating the BERT embedding, the attention mechanism, and the Siamese neural network for the semantic textual similarity task. Also, in [Alshammeri et al. 2021] the proposed siamese araBERT system demonstrates the effectiveness of the BERT model.

The table below compares the state-of-the-art approaches.

Approach	Model	Dataset	Result
[Alzahrani 2016]	dictionary and MT	human-rated benchmark	0.8657
[Li et al. 2006]	cosine	Benchmark created by [Li et al. 2006]	0.86
[Abd Alameer 2017]	cosine and N-gram	NF	NF
		Arabic paraphrasing	0.182
[Nagoudi and Schwab 2017]	word embedding	MSRvid	0.691
		SMTeuropar	0.206
		Arabic paraphrasing	0.354
[Alian, and Awajan 2021]	LR and SVM	MSRvid	0.743
		SMTeuropar	0.467
[Alsaleh et al. 2021]	AraBERT	QurSim dataset	92%
[Saidi et al. 2023]	ArabicBERT	MSRParaphrase	0.188
		MSRvid	0.238
		SMTeuropar	0.371
	CaMelBERT	MSRParaphrase	0.325
		MSRvid	0.372
		SMTeuropar	0.201
	AraBERT	MSRParaphrase	0.154
		MSRvid	0.550
		SMTeuropar	0.220
[Alshammeri et al. 2021]	siamse araBERT	Quran	84.96%

Table 1: State-of-the-art approaches: machine translation (MT), linear regression (LR) and support vector machine regression (SVM), NF stands for Not Found

According to Alsaleh et al. (2023), the AraBERT model achieved the best results among the models evaluated in Table 1. This observation motivates the design of a Cross-Encoder model in this study, which would leverage the best Arabic BERT model and incorporate a GRU model.

We focused our literature review on the Arabic STS systems but we interested again on the recent English STS models. Table 2 summarizes the results obtained for each approach.

Approach	System	Metric	Result
[Jiao et al. 2019]	TinyBERT	Pearson Corr.	80.4
[Sanh et al. 2019]	DistilBERT	Pearson Corr.	86.9
[Sun et al. 2020]	ERNIE	Pearson Corr. Spearman Corr.	87.6 86.5
[Liu et al. 2019]	STS-B	Pearson Corr.	92.2
[Lan et al. 2019]	AI-BERT	Pearson Corr.	92.5
[Yang et al. 2019]	XLNet	Pearson Corr.	93
[Raffel et al. 2020]	T5-11B	Pearson Corr. Spearman Corr.	93.1 92.8

Table 2: English STS State-of-the-art approaches: STS-B [Cer et al. 2017] is used in all baselines.

Finally, Tokenization is the first step in any NLP model, and it involves splitting input text into tokens. The BERT tokenizer uses WordPiece tokenization, which means that a word can be broken down into smaller subwords. The BERT vector for a word depends on the entire sentence, so a word can have multiple vectors depending on the context. There are many different tokenizers available for the Arabic language. For example, AraBERT uses SentencePiece to train a BERT model from scratch on Arabic text that has been segmented with FARASA. The pre-trained mBERT tokenizer is character-based, which means that its vocabulary consists of individual characters.

3 Proposed method

This work makes two contributions. First, we apply data augmentation techniques to the Arabic STS dataset. Second, we design a Cross-BERT-GRU model for the Arabic STS task.

3.1 Backtranslation for Arabic STS Data augmentation

Data augmentation is a procedure used to artificially generate new samples from the training data set by applying various techniques. It is mainly used to improve the performance of deep neural networks for small training data sets, as in the case of low-resource language. In this paper, we choose the backtranslation technique. It is the process of translating a sentence into English and then translating it back into Arabic. Back Translate is a well-known sense-preserving approach, so it is suitable for text classification [Ma and Li 2020] in general and for STS tasks in particular.

3.2 GRU-BERT model for Arabic STS

We cast STS as a multiclass classification problem where each pair of sentences (input) has a categorized score (class) ranging from 0 to 5. We propose a Cross-BERT-GRU model based on a cross-encoder network that learns semantic textual similarity between sentences. The general structure of the network is shown in Figure 3.2. It has two major components: a BERT word embedding layer for feature extraction and a GRU layer for long-term dependencies intra sentences.

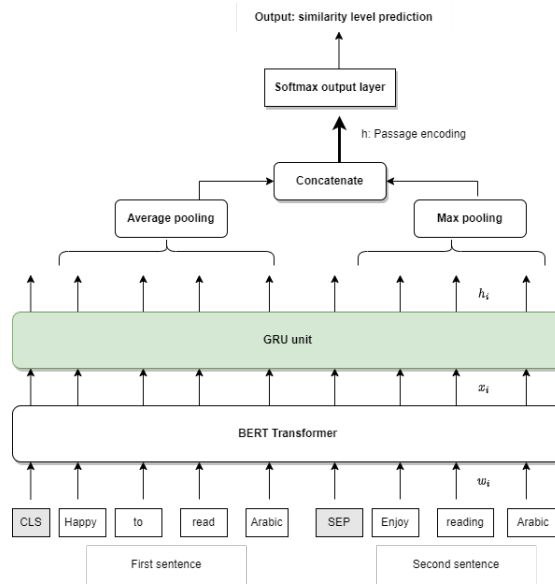


Figure 1: Cross encoder Cross-BERT-GRU architecture. It takes two sentences as input and produces an output indicating the degree of similarity between them.

To assess the similarity between two sentences, we create a passage by concatenating the sentences with a separator token, "SEP." This combined passage is then fed into the BERT-based embedding layer. We used the most available pre-trained BERT models for modern standard Arabic (MSA) including AraBERT [Antoun et al. 2020], Arabic-BERT [Safaya et al. 2020] and CAMEL-BERT [CamelBERT]. It is worth mentioning that the multilingual mBERT [Libovický et al. 2019] can also handle Arabic texts. The difference between these BERT versions lies in the internal architecture and the type of tokenizer adopted. We chose BERT because it is well known for being able to detect long-range dependencies. The output data from the BERT model is inputted to a subsequent GRU to extract the hidden representation for each word. We opt for GRU because it is able to capture long-term dependencies and has a performance similar to that of LSTM with less computation. The sentence pair representation is obtained by concatenating the results of mean-pooling and max-pooling of GRU output layer and exclude [SEP] embeddings. Finally, the output of the sentence representation vector is followed by a softmax layer to obtain the class probability distribution. The size of the output layer is 6 since we

have 6 classes. We use cross-entropy loss function to train the model and accuracy as an evaluation metric to assess the quality of the model.

Mathematically, let w_i represents the word i of the passage fed to BERT, $x_i = BERT(w_i)$ represents the encoding of the word w_i where *BERT* refers to the output of the embedding layer. x_i is then inputted into the GRU unit for processing, resulting in an output h_i . The GRU unit consists of several computations involving trainable parameters and gates. A GRU unit operates as follows.

$$\begin{aligned} z_i &= \sigma(W_z x_i + U_z h_{i-1}) \\ r_i &= \sigma(W_r x_i + U_r h_{i-1}) \\ \hat{h}_i &= \tanh(W x_i + U(r_i \odot h_{i-1})) \\ h_i &= (1 - z_i) \odot h_{i-1} + z_i \odot \hat{h}_i \end{aligned}$$

Where σ and \tanh are the sigmoid and tanh activation functions, \odot denotes element-wise multiplication and W_z, W_r, W, U_z, U_r, U are the weight matrices of the GRU. At step i , GRU inputs x_i and outputs h_i .

We denote by *MaxPool* and *AvgPool* be the element-wise max-pooling, respectively average pooling, of the passage over h_i . The global encoding of the passage $h = [MaxPool, AvgPool]$ is defined as the concatenation of two vectors *MaxPool* and *AvgPool*. Finally, the encoded representation h is inputted into a softmax layer to estimate the level of similarity between the sentences that make up the passage

We noticed that the Arabic-BERT model outperforms others in tokenization, thanks to its consideration of Arabic word proclitics and enclitics.

4 Results and Discussion

In this section, we conduct experiments to compare the performance of Cross-BERT-GRU with the state-of-the-art approaches on three STS datasets.

4.1 Datasets

For training, we used the Arabic Semantic Textual Similarity corpus [Arabic STS corpus]. This training data set is released for the SEMEVAL 2017 Multilingual Semantic Textual Similarity: Arabic subtask (Track 1). It contains three resources: Microsoft Research Paraphrase Corpus (MSR-Paraphrase) [MSR-Paraphrase Dataset], Microsoft Research Video Description Corpus (MSR-Video) [MSR-Video Dataset] and WMT2008 development dataset (SMTeuroparl) [SMTeuroparl Dataset].

The participating systems are given two sentences and instructed to report a continuous value similarity score on a scale of 0 to 5, with 0 representing perfect semantic independence and 5 indicating semantic equality. The Spearman's correlation between machine-assigned semantic similarity scores and human judgments is used to evaluate performance.

For both test and validation evaluations, we used the corpus of evaluation for Arabic STS corpus for all experiments (with the original data and the augmented), it contains 250 pairs (Arabic-Arabic).

Data	#Pairs	#Sentences
MSR-Paraphrase	510	1020
MSR-Video	368	736
SMTeuroparl	203M	406

Table 3: statistical information for the Arabic Semantic Textual Similarity (STS) task

4.2 Implementation details

For BERT fine-tuning, we used the pre-trained BERT-BASE model. All models are fine-tuned under the following hyperparameters. The total number of Transformer blocks is 12, 768 hidden layer blocks, and 12 self-attention heads. For the optimizer, we used Adam [Kingma and Ba 2014], a sequence length of 128, a batch size of 64 and a learning rate of 10^{-5} , the dropout probability is set to 0.1. We fine-tuned for 10 epochs, keeping the best model so far. We used the development set of evaluation for Arabic STS dataset, presented in Section 4.1, to fix the best parameters for our tests when fine-tuning. The size of the vocab and the total number of parameters of each pre-trained model are presented in Table 4.

Model	Ara-BERT	Arabic-BERT	CAMeL-BERT	mBERT
Parameters	135M	110M	108M	110M
Normalization	yes	no	yes	yes
Textual Data Size	27GB	95GB	167GB	61GB

Table 4: Characteristics of the existing Arabic BERT models.

4.3 Evaluation metric and results

We assess the predicted similarities against the provided annotated similarities using two metrics: MSE (Mean Squared Error) and Spearman’s correlation. Initially, we compute MSE by rescaling the golden similarities to fall within the range of $[0, 1]$. MSE (1) is defined as the average squared difference between the predicted values (\hat{y}_i) and actual values (y_i). It provides a quantitative assessment of the overall accuracy of the predictions.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (1)$$

The outcomes of the four Arabic BERT models are presented in Table 5. Notably, the *ArabicBERT* model demonstrates superior MSE scores, achieving 0.069 for the original data and 0.064 for the augmented data in the MSR-Video corpora. This success can be attributed to the ArabicBERT tokenizer’s adept handling of Arabic language nuances, specifically its consideration of morphological characteristics. Table 5 further supports the use of data augmentation techniques, even within a transformer-based approach, as a means to mitigate the risk of overfitting.

Model	MSR-Para	MSR-Video	SMTeu-roparl	MSR-Parap (with)	MSR-Video (with)	SMTeu-roparl (with)
ArabicBERT	0.121	0.069	0.110	0.118	0.064	0.106
CAMeL-BERT	0.139	0.078	0.122	0.135	0.074	0.118
AraBERT-V2	0.144	0.086	0.132	0.138	0.081	0.127
mBERT	0.287	0.201	0.265	0.283	0.194	0.262
ArabicBERT ([Saidi et al. 2023])	0.188	0.238	0.371	-	-	-
CAMeL-BERT ([Saidi et al. 2023])	0.325	0.372	0.201	-	-	-
AraBERT ([Saidi et al. 2023])	0.154	0.550	0.220	-	-	-
mBERT ([Saidi et al. 2023])	0.285	0.489	0.382	-	-	-

Table 5: MSE of STS systems on the STS dataset (the lower, the better), (with) means with augmented data.

The second evaluation metric is the Spearman correlation coefficient. It is a measure of the rank correlation between two variables, such as the Golden semantic similarity and the Cosine similarity. It can be seen as a rank-based version of Pearson's correlation coefficient. Furthermore, it is more reliable against outliers and is suitable for non-normal distributions. The mathematical expression of the Spearman correlation coefficient, denoted as ρ , is as follows (2):

$$\rho = 1 - \frac{6 \sum_{i=1}^n (d_i)^2}{n(n^2 - 1)} \quad (2)$$

where d_i represents the difference in ranks between the two variables for each sample and n represents the number of sentence pairs.

For a given pair of sentences, golden semantic similarity ranges from 0.0 to 5.0, the higher the score, the higher semantic similarity. Spearman's correlation coefficient ranges from -1 to +1 where -1 shows a perfect negative correlation, 1 shows a perfect correlation, and 0 indicates that there is no relationship between the two variables.

Our primary experimental findings are presented in Table 6. Cross-BERT-GRU stands out as the top performer, exhibiting the highest average Spearman's correlation coefficient across all datasets. Table 6 provides a comparison with well-known baselines using the STS dataset. It highlights the superiority of all BERT-based models over traditional state-of-the-art approaches, underscoring the contextualized strength inherent in BERT. Furthermore, the table demonstrates that the application of data augmentation techniques enhances the performance of all BERT models.

Model	MSR-Para	MSR-Video	SMTeu-roparl	MSR-Para (with)	MSR-Video (with)	SMTeu-roparl (with)
Cross-BERT-GRU(ours)	0.831	0.936	0.854	0.836	0.941	0.857
CAMeL-BERT	0.813	0.929	0.825	0.817	0.932	0.829
AraBERT-V2	0.797	0.923	0.814	0.799	0.926	0.817
mBERT	0.669	0.871	0.702	0.672	0.874	0.705
[Alian, and Awajan 2021]	0.354	0.743	0.467	-	-	-
[Nagoudi and Schwab 2017]	0.182	0.691	0.206	-	-	-

Table 6: Evaluation of STS systems on the STS dataset using Spearman correlation coefficient (higher values indicate better performance). The last three columns correspond to the augmented dataset.

In conclusion, it's worth noting that both MSE and Spearman coefficient align in their evaluation, affirming the consistency of our results. The suggested Cross-BERT-GRU architecture exhibits superior performance, surpassing other BERT architectures, which, in turn, outperform traditional approaches.

5 Conclusion

In this paper, we present a novel methodology that combines the BERT and GRU models to accurately predict the semantic similarity between Arabic texts. In our study, we leverage multiple versions of Arabic BERT as the word embedding layer. Additionally, we employ back-translation techniques to augment the datasets. The experimental results demonstrate also that our proposed approach outperforms other transformer based models for the Arabic language. It achieves a Spearman correlation coefficient of 0.9 between the golden and predicted similarities. As a future extension of this research, we intend to explore the combination of BERT with other deep learning architectures. Additionally, we aim to incorporate more advanced data augmentation techniques, such as those based on large language model [Saidi et al. 2022]. One limitation of Cross-BERT-GRU is its necessity for fine-tuning when applied to a domain different from the corpora on which BERT was originally trained.

References

- [Abd Alameer 2017] A. Q. Abd Alameer, "Finding the similarity between two arabic texts." *Iraqi Journal of Science*, 152-162, 2017.
- [Almiman et al. 2020] A. Almiman, N. Osman, and M. Torki. "Deep neural network approach for Arabic community question answering." *Alexandria Engineering Journal*, 59(6), 4427-4434, 2020.

- [Alian, and Awajan 2021] M. Alian, and A. Awajan. "Arabic sentence similarity based on similarity features and machine learning". *Soft Computing*, 25(15), 10089-10101, 2021.
- [Alshammeri et al. 2021] Alshammeri, M., Atwell, E., Alsalka, M. A. (2021). A Siamese Transformer-based Architecture for Detecting Semantic Similarity in the Quran. In The International Journal on Islamic Applications in Computer Science And Technology-IJASAT (Vol. 9, No. 4). Design For Scientific Renaissance.
- [Antoun et al. 2020] W. Antoun, F. Baly, and H. Hajj. "Arabert: Transformer-based model for arabic language understanding." *arXiv preprint arXiv:2003.00104*, 2020.
- [Arabic STS corpus] <https://alt.qcri.org/semeval2017/task1/index.php?id=data-and-tools>. Last accessed 05 November 2022.
- [Alsaleh et al. 2021] A. N. Alsaleh, E. Atwell, and A. Altahhan. "Quranic Verses Semantic Relatedness Using AraBERT". In *Proceedings of the Sixth Arabic Natural Language Processing Workshop* (pp. 185-190). Leeds, 2021.
- [Alzahrani 2016] S. Alzahrani, "Cross-Language Semantic Similarity of Arabic-English Short Phrases and Sentences". *J. Comput. Sci.*, 12(1), 1-18, 2016.
- [Al-Bataineh et al. 2019] H. Al-Bataineh, W. Farhan, A. Mustafa, H. Seelawi, and H. T. Al-Natsheh. "Deep contextualized pairwise semantic similarity for arabic language questions." In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)* (pp. 1586-1591). IEEE, 2019.
- [Al-Shamery and Ghenni 2016] E. S. Al-Shamery and H. Q. Ghenni. "Plagiarism detection using semantic analysis." *Indian Journal of Science and Technology*, 9(1), 1-8, 2016.
- [Bromley et al. 1993] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'siamese' time delay neural network." *Advances in neural information processing systems*, 6, 1993.
- [CamelBERT] CAMEL-BERT, <https://huggingface.co/CAMEL-Lab/bert-base-arabic-camelbert-ca>. Last accessed 10 March 2023.
- [Cer et al. 2017] Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. arXiv preprint arXiv:1708.00055.
- [Gabr et al. 2023] Gabr, M. A. E. I., Badr, A. Z., Mahdi, H. M. (2023). EXPLORING STRATEGIES FOR MEASURING SEMANTIC SIMILARITY IN SHORT ARABIC TEXTS.
- [Hliaoutakis et al. 2006] A. Hliaoutakis, G. Varelas, E. Voutsakis, E. G. Petrakis, E. Milios. "Information retrieval by semantic similarity". *International journal on semantic Web and information systems (IJSWIS)*, 2(3), 55-73, 2006.
- [Ho et al. 2010] C. Ho, M. A. A. Murad, R. A. Kadir, and S. C. Doraisamy. "Word sense disambiguation-based sentence similarity". In *Coling 2010: Posters* (pp. 418-426), 2010.
- [Humeau et al. 2019] S. Humeau, K. Shuster, M. A. Lachaux, and J. Weston. "Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring". *arXiv preprint arXiv:1905.01969*, 2019.
- [Jiao et al. 2019] Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., ... Liu, Q. (2019). Tinybert: Distilling bert for natural language understanding. arXiv preprint arXiv:1909.10351.
- [Kingma and Ba 2014] D. P. Kingma, and J. Ba. "Adam: A method for stochastic optimization". *arXiv preprint arXiv:1412.6980*, 2014.
- [Lan et al. 2019] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.

- [Li et al. 2006] Y. Li, D. McLean, Z. A. Bandar, J. D. O'Shea, and K. Crockett. "Sentence similarity based on semantic nets and corpus statistics". *IEEE transactions on knowledge and data engineering*, 18(8), 1138-1150, 2006.
- [Libovický et al. 2019] J. Libovický, R. Rosa, and A. Fraser. "How language-neutral is multilingual BERT?". *arXiv preprint arXiv:1911.03310*, 2019.
- [Liu et al. 2019] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- [Ma and Li 2020] J. Ma, and Li Li. "Data Augmentation For Chinese Text Classification Using Back-Translation." *In Journal of Physics: Conference Series* (Vol. 1651, No. 1, p. 012039). IOP Publishing, 2020.
- [Murad et al. 2010] C. Ho, M. A. A. Murad, R. A. Kadir, and S. C. Doraisamy. "Word sense disambiguation-based sentence similarity". *In Coling 2010: Posters* (pp. 418-426), 2010.
- [MSR-Paraphrase Dataset] <http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042/>. Last accessed 30 November 2022.
- [MSR-Video Dataset] <http://research.microsoft.com/en-us/downloads/38cf15fd-b8df-477e-a4e4-a4680caa75af/>. Last accessed 30 November 2022.
- [Nagoudi and Schwab 2017] D. Schwab. "Semantic similarity of arabic sentences with word embeddings." *In Third arabic natural language processing workshop*, (pp. 18-24), 2017.
- [Raffel et al. 2020] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1), 5485-5551.
- [Sa'ad et al. 2021] M. Hammad, M. Al-Smadi, Q. B. Baker, and A. Sa'ad. "Using deep learning models for learning semantic text similarity of Arabic questions." *International Journal of Electrical and Computer Engineering*, 11(4), 3519, 2021.
- [Saidi et al. 2022] R. Saidi, F. Jarray, J. Kang, D. Schwab. (2022, October). GPT-2 Contextual Data Augmentation for Word Sense Disambiguation. In PACIFIC ASIA CONFERENCE ON LANGUAGE, INFORMATION AND COMPUTATION.
- [Saidi et al. 2023] Saidi, R., Jarray, F., Alsuhaibani, M. (2023). SiameseBERT: A Bert-Based Siamese Network Enhanced with a Soft Attention Mechanism for Arabic Semantic Textual Similarity. In ICAART (3) (pp. 146-151).
- [Safaya et al. 2020] A. Safaya, M. Abdullatif, and D. Yuret. "BERT-CNN for Offensive Speech Identification in Social Media." *In Proceedings of the Fourteenth Workshop on Semantic Evaluation, Barcelona (online), International Committee for Computational Linguistics*, 2020.
- [Sanh et al. 2019] Sanh, V., Debut, L., Chaumond, J., Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
- [SMTeuroparl Dataset] <http://www.statmt.org/wmt08/shared-evaluation-task.html>. Last accessed 30 November 2022.
- [Sun et al. 2020] Sun, Y., Wang, S., Li, Y., Feng, S., Tian, H., Wu, H., Wang, H. (2020, April). Ernie 2.0: A continual pre-training framework for language understanding. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 05, pp. 8968-8975).
- [Wieting et al. 2019] J. Wieting, T. Berg-Kirkpatrick, K. Gimpel, and G. Neubig. "Beyond BLEU: training neural machine translation with semantic similarity". *arXiv preprint arXiv:1909.06694*, 2019.
- [Yang et al. 2019] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.