International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Identifying Similarity Between Sentences Using Natural Language Processing

Medidi Jeneeth Venkata Sai Ajit Satya Dev

Student, Rajam, Vizianagaram, 532127, India.

ABSTRACT

Sentence Similarity may be a measure of how similar two pieces of text are, or to what degree they express the identical meaning. Word embeddings became widespread in language Processing (NLP). They allow us to simply compute the semantic similarity between two words, or to a target word. The lexical and semantic content of texts is captured by embeddings, which can be found by pre-trained encoders or bag-of-words techniques employing the embeddings of component words. However, often were more fascinated by the homogenity between two sentences. while performing various kinds of NLP tasks like text summarization and text retrieval the concept of sentence similarity critical aspect.As already many traditional approaches are in practice for making this Siamese neural network perform comparatively great. The various contributions of different words to the general phrase semantics are ignored in the sentence semantic representation that is produced by sharing weights within the SNN with none attention mechanism. The diverse contributions of different words to the phrase semantics are ignored by the sentence semantic representation, which is produced by sharing weights inside the SNN without an attention mechanism.

Keywords: Natural Language Processing (NLP), Sentence Similarity, Siamese Neural Networks (SNN), semantic..

1. Introduction

The field of engineering referred to as "natural language processing" (NLP) is more particularly the sector of "artificial intelligence" (AI) that's concerned with providing computers the capacity to grasp written and spoken words in an exceeding manner like that of humans. NLP blends statistical, machine learning, and deep learning models with computational linguistics—rule-based modelling of human language. With the employment of those technologies, computers are now able to interpret human language within the type of text or audio data and fully "understand" what's being said or written, including the speaker's or writer's intentions and mood.

Sentence Similarity is the one of the most used applications of natural language processing. Finding how similar two texts are the job of sentence similarity. Sentence similarity models take input texts and turn them into vectors (embeddings) that capture semantic data and determine how similar (near) they are to one another. Clustering and grouping as well as information retrieval benefit greatly.You can extract information from documents using Sentence Similarity models. the primary step is to rank documents using Passage Ranking models. you'll be able to then get to the highest ranked document and search it with Sentence Similarity models by selecting the sentence that has the foremost similarity to the input query.

2. Literature Survey

In paper [1]Y. Yu, D. Qiu and R. Yan, The major goal of this paper is to use dimensionality reduction to enhance the quantum entanglement-based sentence representation. In this part, we examine various relevant efforts on language similarity and quantum computation.When two systems are in the quantum entangled state, no matter how far apart they are, learning something about one system immediately reveals something about the other.Two issues should be taken into account in comparison to the models given. First, just a word's semantic content is taken into account, with the effect of other words being completely disregarded(MODEL-I). Second, approaches based on dependency trees that involve complicated computational processing take the relationship between words into account(MODEL-II,MODEL-III).

In paper [2]Z. Liang and S. Zhang, Particularly difficult are the two challenges of finding a similar sentence for a given sentence and assessing the semantic similarity between two sentences. To solve the MSS problem a new algorithm is proposed and it is called as Syntactic and Semantic Short-Term Memory (SSLSTM). As there are logical connections between GSS and MSS algorithms, GAN framework is proposed and it is called Sentence Similarity Generative Adversarial Network (SSGAN). Three versions of GAN are proposed, they are, Classic GAN(C-SSGAN), Hybrid GAN(H-SSGAN), Black-Box GAN(B-SSGAN).

In paper [3] Z. Li, H. Lin, W. Zheng, M. M. Tadesse, Z. Yang and J. Wang, As the percentage of biomedical information is growing, the number of

medical texts is also increasing. However, a lot of the sentences in these massive amounts of data have semantically similar meanings but entirely different text descriptions, which causes medical research a lot of unnecessary trouble. In order to extract useful biomedical information, such as DDIs, it is crucial to assess the similarity presented in between biomedical texts. Siamese Neural Network (SNN) is one of the best approaches for this scenario

In paper [4]Li, Yuhua& McLean, David & Bandar, Zuhair & O'Shea, James & Crockett, Keeley.Sentence similarity has been shown to be one of the most successful methods for increasing the performance of website recovery, when labels are chosen to represent papers inside the named page searching task. The "bag of words" method is another term for word co-occurrence methods. They are frequently used in systems for information retrieval. The systems already have n words in their vocabulary. In order to include all expressions in a language, the value of n is typically in the thousands or many thousands. Based on semantic and order information, this research described a method for calculating the semantic similarity of sentences or very short texts. First, lexical cognitive content and a corpus contribute to semantic similarity. The lexical content stimulates general human understanding of terms used in a natural language; this understanding is occasionally consistent across a wide range of language application domains. A corpus reflects the specific ways in which language and words are used.

In paper [5]AniketAnagawadi, Ram Mohana Reddy Guddeti, Saurabh Agarwala, Word embeddings are representations of words that are semantically significant based on their local co-occurrences in sentences. They are practically universal because they make determining the semantic similarity between two words easy and may be used to find terms that are similar to a particular phrase. The lexical and semantic characteristics of words are represented as continuous vectors in a low-dimensional space through embedding generation methods. The many methods now in use to acquire text embeddings are examined in this study, and an unique model to create an embedding and then estimate how similar two input texts are detailed. On the basis of the connection between the ground truth values provided in the dataset and the similarity values projected by the model, the techniques are also tested on the SICK dataset, and the experimental findings show that the innovative model outperforms the other models.

.

3. Data Collection

The below data is collected from the references.

4. Methodology

We conduct experiments using the SICK dataset1, which comprises human comparisons of sentence pairings. It has 10,000 English sentence pairings that are grouped according to how semantically connected they are. Through the use of crowdsourcing methodologies, each pair of sentences has their entailment graded on a scale of 0 to 5.

Taking the average of the word embeddings of the component words and comparing the resulting embeddings using a similarity function, such as cosine similarity, is the simplest way to assess two texts' similarity. To improve this method, it is first necessary to remove the stop words. All other models' performance is evaluated in relation to this standard operating procedure.

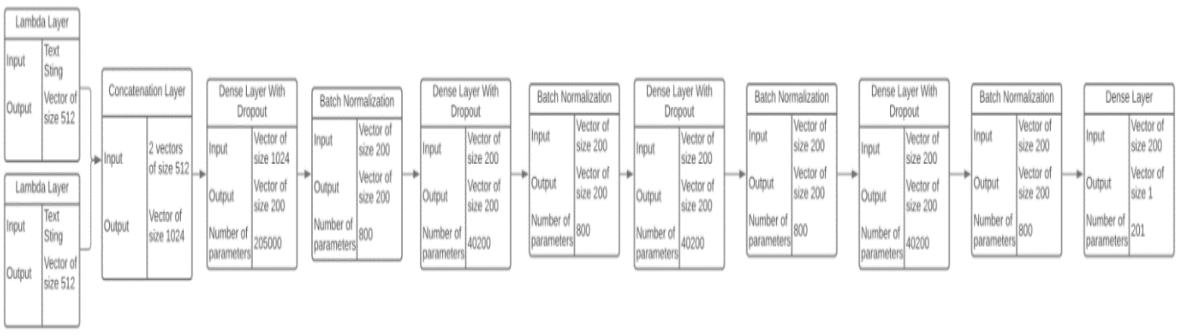


Fig. 1. Novel model architecture.

Novel model : To determine the semantic similarity between two texts, a brand-new deep learning model is created. The innovative model ages the input vectorization capability of the Universal Sentence Encoder. From the input sentences, the universal sentence encoder creates a fixed-length embedding of size 512. Following layers get these concatenated produced embeddings.

The layers present in our model works in this process. Firstly, input layer which takes two sentences as input and followed by lambda layer where the embeddings of the inputs are obtained and then these embeddings are combined in the concatenate layer. Batch normalization layer works to normalize inputs because the parameters at previous layers vary. Dense layer consists of many activated units which are interconnected and Dropout layer works for avoiding the overfitting. In this manner the entire process works.

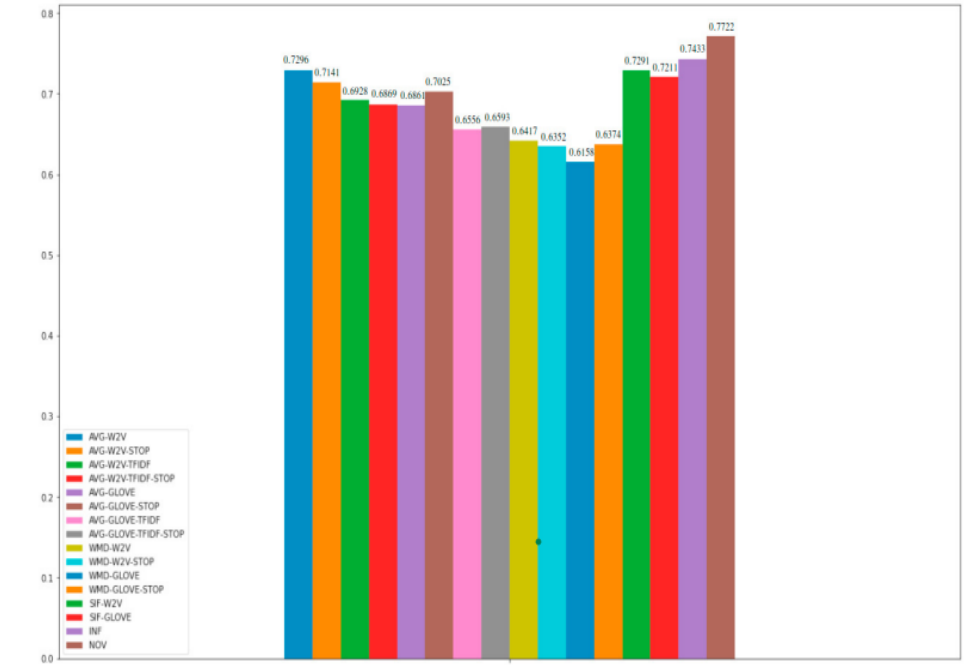
The model is trained using a combination of the development and training sets from the SICK dataset. The dataset is divided 80:20 into a train set and a test set, with the ground truth similarity scores scaled to suit the range [0,1]. For quicker convergence, the Adam optimizer is employed, while the hidden layers use the ReLU (Rectified Linear Unit) activation function. With 512 batches, the model is trained across 1000 epochs.

5. Results and Discussion

The correlation between the predicted similarity values and the human determined (ground truth) similarity values for the identical pairs of phrases presented in the dataset is calculated using the Pearson, Spearman, and Kendall's Tau correlation coefficients.

The results show that a simple baseline may be achieved by utilising just the unweighted average of the word embeddings of the sentence's component words, AVG-W2V and AVG-GLOVE. GLOVE embeddings fall short when compared to Word2vec embeddings, which are employed in bag-of-words-based approaches. Additionally, it has been discovered that WMD-based approaches perform poorly when compared to baseline approaches, whereas InferSent slightly outperforms SIF-based techniques. It is also clear that the new model outperforms all earlier methods when compared to all three criteria when compared side by side. This demonstrates how the Novel model can commonly build exact embeddings from input text that properly capture semantic information.

Fig:2 pearson correlation



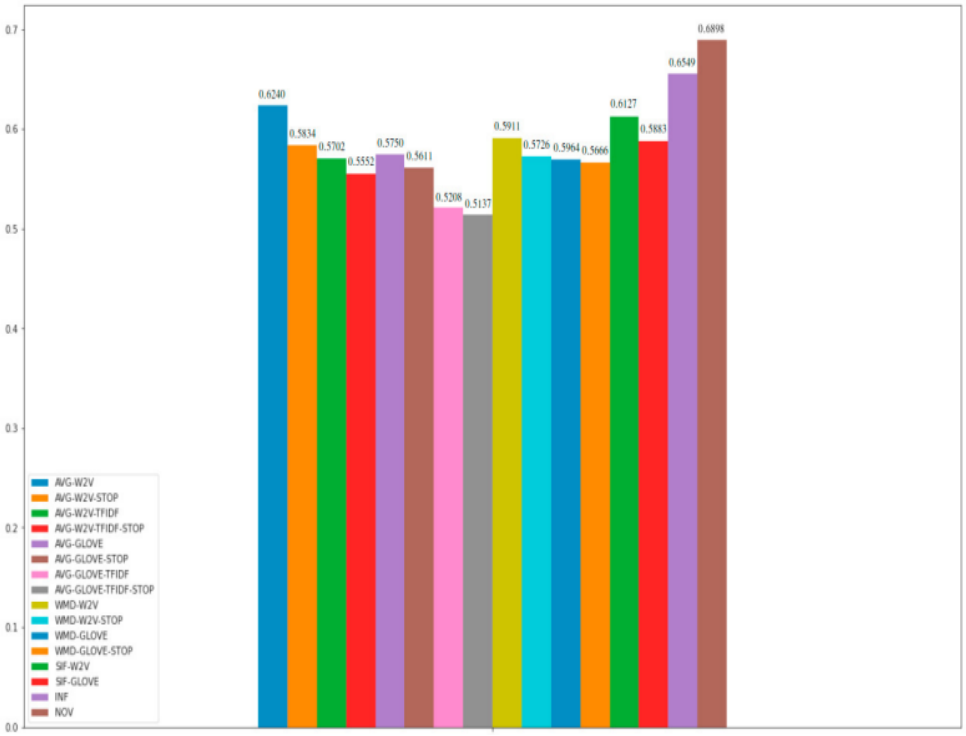


Fig 3: spearmen correlation

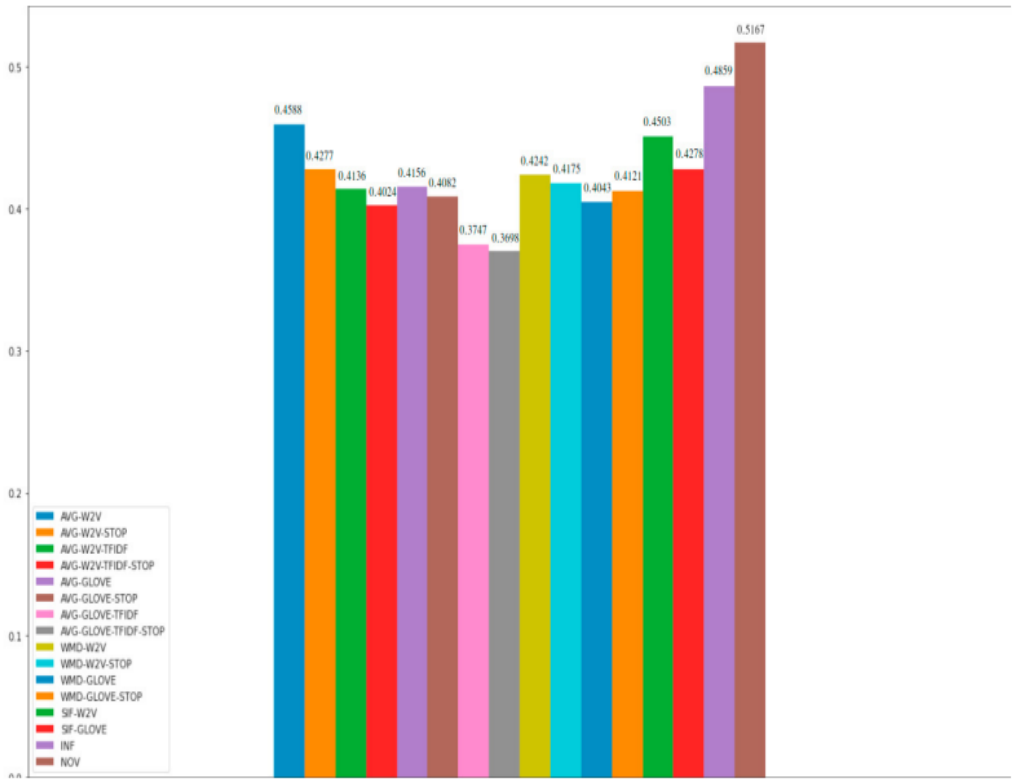


Fig 4 : Kendall's Tau Correlation

6. Conclusion

There are several uses for the semantic similarity score. This semantic similarity score is calculated by comparing the two texts' embeddings for similarity. This study uses Word2vec and GloVe word embedding techniques to build a variety of unique bag-of-words based methodologies to measure semantic similarity. The bag-of-words based techniques additionally include concepts like TF-IDF, Word Mover's distance, and Smooth Inverse Frequency to produce better embeddings from the text. The embeddings are also created using encoder models that have already been trained, such as Infersent. Using a Keras ensemble of Universal Sentence Encoder as a basis and different layers growing into a deep learning model trained on the SICK-train and SICK-dev datasets to determine the semantic similarity score between two texts, a new model to produce embeddings from input text is even created. The cosine similarity function between the obtained embeddings for text pairs is used to compare the various methods on the SICK-test dataset. By comparing the predicted similarity values to the ground truth values, Pearson, Spearman, and Kendall's Tau correlation metrics are used.

The new model outperforms every other model, according to experimental results, and may thus be used to effectively extract semantic data from the input text. Many situations call for semantic similarity. The best performing technique is used to develop a proof-of-concept application to detect semantic similarity between a collection of texts (the novel model). Literature, medicine, and general academics are just a few of the subjects to which the recommended work might be applied. Finally, it would be fascinating to put techniques in place in the future to determine how similar photo collections in various periodicals are to one another.

REFERENCES

[1] Y. Yu, D. Qiu and R. Yan, "A Quantum Entanglement-Based Approach for Computing Sentence Similarity," in IEEE Access, vol. 8, pp. 174265-174278, 2020, doi: 10.1109/ACCESS.2020.3025958.

[2] Z. Liang and S. Zhang, "Generating and Measuring Similar Sentences Using Long Short-Term Memory and Generative Adversarial Networks," in IEEE Access, vol. 9, pp. 112637-112654, 2021, doi: 10.1109/ACCESS.2021.3103669.

[3] Z. Li, H. Lin, W. Zheng, M. M. Tadesse, Z. Yang and J. Wang, "Interactive Self-Attentive Siamese Network for Biomedical Sentence Similarity," in IEEE Access, vol. 8, pp. 84093-84104, 2020, doi: 10.1109/ACCESS.2020.2985685..

[4]Li, Yuhua& McLean, David & Bandar, Zuhair & O'Shea, James & Crockett, Keeley. (2006). Sentence Similarity Based on Semantic Nets and Corpus Statistics. IEEE Transactions on Knowledge and Data Engineering. 18. 1138-1150. 10.1109/TKDE.2006.130.

[5] Saurabh Agarwala, AnikethAnagawadi, Ram Mohana Reddy Guddeti,Detecting Semantic Similarity Of Documents Using Natural Language Processing,Procedia Computer Science,Volume 189,2021,

[6] Farouk, Mamdouh. (2019). Measuring Sentences Similarity: A Survey. Indian Journal of Science and Technology. 12. 10.17485/ijst/2019/v12i25/143977.

[7] Wang, Yue & Di, Xiaoqiang& Li, Jinqing& Yang, Huamin& Bi, Lin. (2018). Sentence Similarity Learning Method based on Attention Hybrid Model. Journal of Physics: Conference Series. 1069. 012119. 10.1088/1742-6596/1069/1/012119.

[8] L. Wang, S. Liu, L. Qiao, W. Sun, Q. Sun and H. Cheng, "A Cross-Lingual Sentence Similarity Calculation Method WithMultifeature Fusion," in IEEE Access, vol. 10, pp. 30666-30675, 2022, doi: 10.1109/ACCESS.2022.3159692.

[9] Xiaofei Sun, Yuxian Meng, Xiang Ao, Fei Wu, Tianwei Zhang, Jiwei Li, Chun Fan; Sentence Similarity Based on Contexts. Transactions of the Association for Computational Linguistics 2022; 10 573–588. doi: https://doi.org/10.1162/tacl_a_00477

[10] FEKI, Wafa&Gargouri, Bilel& Ben Hamadou, Abdelmajid. (2018). Sentence Similarity Computation based on WordNet and VerbNet. Computación y Sistemas. 21. 10.13053/cys-21-4-2853.