EAI Endorsed Transactions
on Energy Web

Research Article EAI.EU

Sentence Semantic Similarity Model Using Convolutional Neural Networks

Karthiga M[1], Sountharrajan S[2], Suganya E [3] and Sankarananth S[4]

[1]Bannari Amman Institute of Technology, Tamilnadu, India
[1]karthigam@bitsathy.ac.in
[2]VIT Bhopal University, Madhya Pradesh, India
[3]Anna University, Tamilnadu, India
[4]Excel College of Engineering and Technology, Tamilnadu, India

Abstract

In Natural Language Processing, determining the semantic likeness between sentences is an important research area. For example, there exists many possible semantics for a word (polysemy), and the synonym of the word differs. Double LSTM (Long Short Term Memory) working at same time on double phrase sequences model is projected to overcome the solitary sequence problem. Furthermore, with the goal of overcoming the second issue, as indicated by the qualities of English dialect, we utilized the British corpus semantic similarity datasets structured by specialists to prepare, and validate the technique. During the training process the stop words were reserved for use. Convolution Neural Network and Semantic Likeness model based on grammar are used to compare the results of our projected representation. The outcomes demonstrate that the proposed methodology is more prominent than the previous approaches by means of precision, recall rate, accuracy etc., along with the enhanced generalization potential of the neural network.

Received on 05 November 2020; accepted on 07 December 2020; published on 25 January 2021

Keywords: Double sequence, deep learning, convolution neural network, Semantic similarity

doi:10.4108/eai.25-1-2021.168226

1. Introduction

Semantic closeness detection is a quantifiable measure that illustrates the similarity in their meanings, when various sequence of text are provided without projecting on their nature of occurrence in the text. One of the major challenges in measuring semantic closeness in a natural language lies in the way of expressing the same information in a variety of manners when analysing the text context [1]. Thus, by utilizing the semantic closeness model, users are allowed to less repetitively reveal the same information meaningfully when evaluating the text messages thereby avoiding redundancy. When numerous text sequences in the same document are formed with corresponding representation, explaining the same thought in various ways throughout the text is considered textual redundancy. Unavoidable increase in volume of text, thereby enhancing the vagueness and uninteresting nature of the text, will occur due to the textual redundancy in spite of their usage in deriving better conclusions in the text.

Deep Learning is a prominent area of research, and a large number of short content semantic likeness detection models have been proposed based on it. We separate this into two divisions: a prototype comprising of single-granularity tiny content semantic similarity and a prototype comprising of multi-granularity tiny content semantic similarity. Words and phrases are expressed as vectors in the single-granularity tiny content semantic similarity prototype and the sentence likeness esteem is obtained by determining vector similarity using Deep Structured Semantic Model [2], Convolutional Latent Semantic Model [3], and LSTM-Recurrent Neural Network. The prototype on multi-granularity text similarity [4] model is based upon the single-greyscale textual semantic closeness, and not only words/ phrases but also its combinations

*Corresponding author. Email: karthigam@bitsathy.ac.in are considered for text representation such as Multi-Granularity Convolution Neural Network model [5] , and the Multi Variable-Long Short Term Memory technique [6]. Semantic likeness detection in English sentences using the above two models results in two issues: 1) Single grouping methodology cannot manage the problem of polysemous words and synonymously equivalent words. 2) These models do not consider the structural dependency in the characteristics of English language. Stop words in sentences give logical meaning, as well as semantic data. So these stop words are also needed to be considered for understanding the logical flow of the text.

With the goal to take care of the principal issue, we suggest a semantic content likeness prototype for twofold short English sentences. Two LSTMs with similar structure and features are utilized for processing the textual pair to prevail over the drawbacks of the single sequence technique. Accuracy of the model is further en-hanced by increasing the text semantic differences. Text semantic differences are enhanced by including the product variance details from the results of training. To handle the next issue, the dataset is created and the stop words are utilized during the preparation of the prototype (the prevailing ones more often expel Ignorable words). The proposed model depicts the use of convolution neural networks to train the ma-chine in overcoming the polysemy problem and considering the English semantics of the terms in grouping the sentence. Convolution Neural Network and Semantic Like-ness model based on grammar are used to compare the results of our projected repre-sentation. The outcomes demonstrate that the proposed methodology is more promi-nent than the previous approaches by means of precision, recall rate, accuracy etc.., along with the enhanced generalization potential of the neural network.

2. Related Work

Current research in the area of Language Processing has given valuable results to semantic relations between the sentences and words. This part investigates the ad-vantages and limitations of existing methods to identify the semantic measures. It also strengthens the field of semantic web measures and relevance to Co-relations methods. [7]. Hanna Bechara et al [8] proposed a semantic text similarity model using the classifier, Support Vector Machine. Data sets are composed from the SemEval repository. The comparison is made between English and Spanish sentences. The system provides better results for English sentences compared to Spanish with an average ranking of 33 amongst 74 runs and a correlation result of 0.72 for Pearson using 5 different test data sets. Semantic similarity between two words using Support Vector machine is proposed by Karthiga et al in [9] [10]. Semantic text closeness model for Hindi language using supervised learning approach and rule based model was undergone by Darshan Agarwal [11]. For determining the semantic closeness among two sentences in Hindi language Paninian reliance grammar which depicts the orientation of correlated words has been employed. Data set has been gathered from Hindi newspapers and different Hindi essays. Supervised support vector machine classifier is utilized to conduct experiments by repeatedly adding features with best combination, which results in increased contexts. It is the first system to measure semantic evaluation on Hindi sentences and the test results provide about 79.9% correlation.

Recent machine learning algorithms using natural language processing are utilized to extract actual models of similarities between the sentences and words. Word co-occurrence methods are generally utilized in the Information Retrieval (IR) systems. It has a list of significant words and every query is considered as a document. A vector is framed for query sets of documents. The related documents are extracted based on the likeness between the query vector and text vector [12].

A dynamic methodology based on convolutional neural network has been proposed by Kalchbrenner et al [13] in their project wherein aspects such as changing maximum-pooling, without external sources and independent of programming language based similarity model is utilized. Features are extracted based on dynamic perspectives with the help of CNN and the model is evaluated using various similarity considering metrics for determining the similarity among the sentences by He at al in their work [14]. Combined methods of CNN and LSTM are utilized by He and Lin [15] to determine the semantic similarity among the words. Combined models of CNN and wordembeddings based technique is utilized by Haipeng Yao [16]. In our proposed model, combination of CNN and double LSTM based semantic likeness strategy has been utilized and also removal of stop words with effective transitive and replaceability based outcomes are projected in comparison with the existing CNN based models.

3. Proposed Model

3.1. Content preparing and training

The British corpus Wikipedia is pre-processed and pre-trained.

Pre-processing. The British corpus dictionary holds the URL and different identity sentences. The corpus dictionary is processed by using wiki-extractor tool, leaving just the English content and removing the URLs. Secondly, the British corpus dictionary comprises traditional English sentences and these are converted to normal English sentences. Third, before determining the text similarity, words in the text should be segmented and this is done by utilizing word embedding methodology.

Pre-Training. From literature reviews, it is understood that Word Embedding methodology is far better than word2vec model in determining the textual similarity. Se-quences of words are pre-trained to word-vector-sequences using word embedding technique. The proposed prototype uses double text sequences and LSTM technique for training these double sequences as represented in the structural design of the pro-posed prototype in Fig. 1. The structural design reveals the following explanation: (1) word sequences (w1, w2) are formed by training the textual inputs (t1 and t2) using word embedding technique. The transformed word sequences are mapped with the trained word-vector-sequences (vec1, vec2). (2) vec1 and vec2 are provided as inputs to the LSTM phase. Two feature text vectors (v1 and v2) are obtained as results from LSTM. The 'p' indicates the product of v1 and v2, 'q' indicates the variance of v1 and v2. After determining the product and variance, p, q, v1 and v2 are joined together. (3) The results from the connector are passed to different dense layers of neural network for obtaining the final semantic similarity outcome.

The product and variance results of the training word vectors along with double text sequences are combined to obtain the final score in our proposed model in contrast with existing approaches. The product enhances the same part of double sequences thereby improving the prototype sensitivity. Variance is used to determine the differences between the double sentences thereby enhancing the accuracy of the prototype.

4. Results and Analysis

4.1. Parameter background creation

The datasets for English tiny text semantic likeness is generally small. Wikipedia and British corpus informational indexes available are not paid and expensive. Our proto-type utilizes the Wikipedia's collection of English information up to 1.75GB. An English data up to 1GB is obtained after expelling irrelevant information. Word Embeddings are utilized to obtain the word-vector-sequence. (The obtained word-vector is 423698; the dimension is 400, covering the generally used English words).

4.2. Datasets used for training

The accuracy of training phase is purely dependent upon the quality of the dataset utilized. Highly efficient data with appropriate size is utilized most effectively in our training model. The training dataset are of three sections:

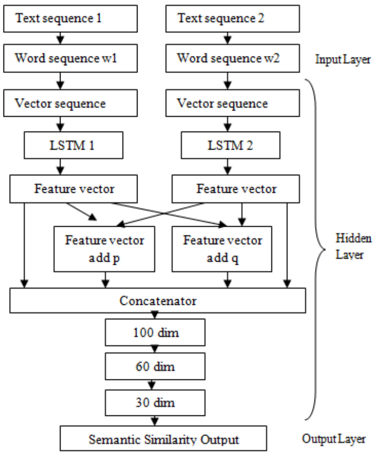

Figure 1. Proposed prototype structural representation.

Training data developed by specialists. The English dialect specialists are asked to choose 1000+ primary sentences using English grammar [8][9]. A pair among the sentence is made by combining the actual sentence and the newly created sentence. Specialists are asked to determine the semantic value for these pairs. Semantically similar sentences are ranked as '1' and completely irrelevant sentences are marked as '-1' and meaningfully opposite sentences are ranked as '0'. Finally, 2000+ sets of sentences as per English language structure rules are arranged. [8-9]

Training data developed by automation. From Wikipedia sentences, word-segment apparatus is utilized to remove sentences with similar watchwords. A sentence pair is formed with main sentence by choosing more than ten sentences for every main sentence and the similarity esteem of this pair is noted as 0.0. Finally, 10000+ sets of effective sentences are chosen automatically. As indicated by the English linguistic information, a similar sentence match is made out of 14500 sentences, and the closeness esteem is set apart as 5.0 [8][9].

Testing model Datasets. Transitivity and replaceability data sets are formed in the testing phase as follows: 1) 1700 sets of representative phrases chosen by specialists form the transitivity data. 2) Subject, object, and predicate are replaced as some other words leaving the rest of the sentences as it is in replaceability test data. An aggregate of 3000+ sets of sentences are chosen. These kind of datasets are freely accessible on GitHub.

4.3. Background processing

Dimension of the word-vectors is chosen as 400 in our proposed model. Dimension for the model has to be chosen with proper care as a smaller value of dimension will not provide the exact semantic information of the text whereas large dimension value may increase the calculation difficulty as well as diminish the speed of the training process. In order to progress the generalization of deep convolutional neural network, dropout should be in the range around 20% 40%. In our proposed work, dropout is chosen as 0.4. Smaller dropout value results in over fitting whereas larger dropout value will lead to inefficient learning. For sentences, 300 is set as the maximum length. The prototype utilizes 85% of the total data for training, and the staying 15% as the validation test data. The last emphasis count of the prototype is picked multiple. Total number of iterations handled is 50. In the tests, it is discovered that the accuracy is enhanced and the loss is minimized in prior cycles. The training phase keeps storing the weights of the phases for which accuracy is more. As an activation function, softmax is used and as a loss function, crossentropy is utilized in the final layer.

4.4. Analysis of Training result

The training accuracy is shown in Fig.2. and training loss in Fig. 3. The actual values are depicted as curves in the figures. When the curve in Fig.2. goes upward the curve in Fig. 3.slopes down. From Fig. 2. it is understood that the accuracy is improved more for previous iterations than that of the latter. From Fig. 3. it is understood that the training loss is decreased more in the previous iterations whereas in the latter iterations the changes become stable. After 42 iterations, there is no gradual change in accuracy and loss. So the number of iterations considered is fixed to 50 in our pro-posed work.

For validation, two models are compared. One is English text similarity model using Convolutional neural network readily available from GitHub and the other is the Grammar based semantic likeness detection model. Using the prototype depicted over, the consequences of the training results are depicted in below Tables 1-6.

Precision, recall and f1 score are utilized to depict the experiment outcomes, and number of sentences considered for testing is represented as support. By investigation, we obtained the resultant outcomes: 1) As recognized from Table 1. and Table 2., results of transitive assessment outcomes introduced here are superior to the semantic likeness estimation based on (CNN) Convolution Neural Network and accuracy obtained is maximum. 2) From Table 3 and Table 4, it is observed that the replaceability outcomes of the projected model are superior to the semantic likeness computation based on (CNN) Convolution Neural Network, and a 1% enhancement in the accuracy. 3) It is recognized from Table 1 and Table 5, the transitive outcomes of prototype projected lies superior to the grammar based semantic likeness estimation and the accuracy enhancement is more than 10%. 4) From Table 3 and Table 6 it is understood that the replaceability outcomes of the projected prototype are superior to grammar based semantic likeness estimation and the accuracy enhancement is more than 4%. Fig. 4. and Fig. 5. depicts the precision and recall score of the proposed and existing methodologies and from the figure it is understood that the proposed model outperforms the results of existing.

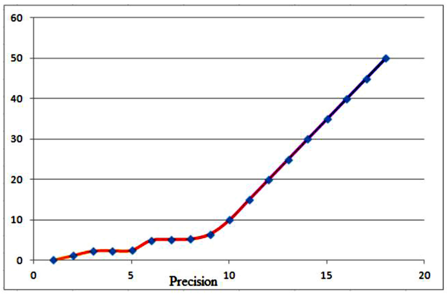

Figure 2. Accuracy obtained from Training.

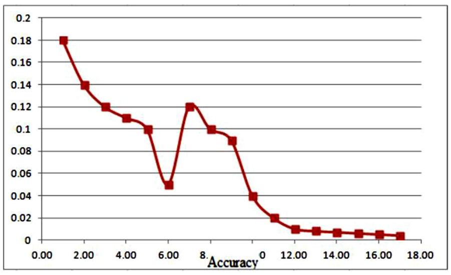

Figure 3. Loss curve.

Table 1. Transitivity Outcomes

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0	0.95	0.85	0.83	75
1	0.00	0.00	0.00	1
2	1.01	1.00	1.00	1
3	0.64	0.75	0.95	14
4	0.81	0.89	0.87	58
5	0.99	0.99	0.99	1499

5. Conclusion and Future Research

Semantic likeness estimation is an important research area and the proposed work aims in determining

Table 2. Transitivity Test Results based on Convolution Neural Networks

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0	0.52	0.51	0.51	75
1	0.00	0.00	0.00	1
2	0.00	0.00	0.00	1
3	0.00	0.00	0.00	14
4	0.70	0.20	0.32	58
5	0.99	0.98	0.97	1541

Table 3. Replaceability Outcomes

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0/1/2	0.00	0.00	0.00	0
3	0.90	0.93	0.90	21
4	0.76	0.77	0.77	125
5	0.99	0.99	0.98	3268

Table 4. Replaceability Test Results based on Convolution Neural Networks

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0/1/2	0.00	0.00	0.00	0
3	0.00	0.00	0.00	25
4	0.83	0.65	0.70	121
5	0.97	0.94	0.92	3268

Table 5. Transitivity Outcomes of Existing Prototype

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0	0.00	0.00	0.00	74
1	0.00	0.00	0.00	1
2	0.00	0.00	0.00	1
3	0.06	0.05	0.06	14
4	0.04	0.14	0.06	55
5	0.91	0.96	0.90	1550

Table 6. Replaceability Outcomes of Existing Prototype

Similarity Values	Prediction			
	Recall	F1Score	Precision	Support
0/1/2	0.00	0.00	0.00	0
3	0.00	0.00	0.02	21
4	0.03	0.01	0.01	121
5	0.97	0.94	0.94	3255

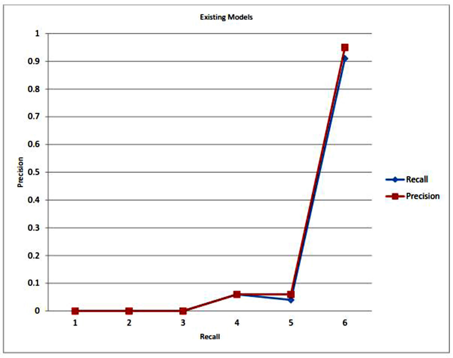

Figure 4. Comparison of Transitivity Outcomes in existing models.

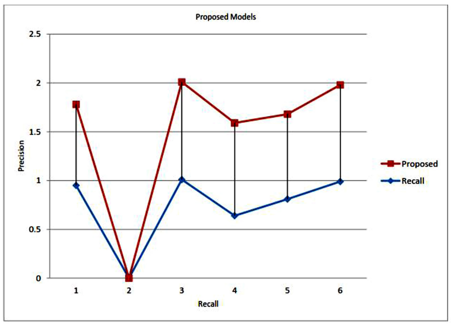

Figure 5. Comparison of Transitivity Outcomes in proposed models.

the same using double-LSTM and dynamic CNN techniques. Semantic likeness estimation between double English sentence sequences are calculated based on the following aspects: First, double LSTM (Long Short Term Memory) working at the consecutively on double English sequences model is project-ed to trounce the single sequence problem of handling many possible semantics for a word (polysemy) and word synonym. Second, stop words are utilized and reserved for determining the similarity among English short texts during training. Third, the replaceable and transitive outcomes of the prototype are determined by using two different test datasets. The outcomes demonstrate that, proposed prototype has a greater enhancement in transitive outcomes and a specific enhancement in replacement outcomes, that in turn enhances the generalization capability of neural networks. In future, additional improvement in semantic datasets and the generalization capability of the network will be focused.

References

[1] Lecun Y., Bengio Y. Convolutional networks for images, speech, and time series [J]. The handbook of brain theory and neural networks, 1995, 3361(10): 1995.

[2] Huang P-S., He X., Gao J., et al. Learning deep structured semantic models for web search using click through data[C]. Proceedings of the 22nd ACM international

conference on Conference on information and knowledge management. Amazon, India, 2013: 2333-2338.

[3] Shen Y., He X., Gao J., et al. A latent semantic model with convolutional-pooling structure for information retrieval[C]. Proceedings of the 23rd ACM international conference on Conference on information and knowledge management. New York, USA, 2014: 101-110.

[4] Palangi H., Deng L., Shen Y., et al. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval[J]. IEEE/ACM Trans-actions on Audio, Speech, and Language Processing, 2016, 24(4): 694-707.

[5] Sountharrajan, S., et al.Dynamic Recognition of Phishing URLs Using Deep Learning Techniques. Advances in Cyber Security Analytics and Decision Systems. Springer, Cham, 2020. 27-56.

[6] Wan S., Lan Y., Guo J., et al.: A deep architecture for semantic matching with multiple positional sentence representations[C]. Proceedings of the 30th AAAI Conference on Arti-ficial Intelligence. Phoenix, USA, 2016: 2835-2841.

[7] Abney S., Light M. Hiding a Semantic Class Hierarchy in a Markov Model. Proc. ACL Workshop Unsupervised Learning in Natural Language Processing, pp. 1-8, 1999.

[8] Hanna Bechara et al. MiniExperts: An SVM Approach for Measuring Semantic Textual Similarity. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 96–101,Denver, Colorado, June 4-5, 2015. c 2015 Association for Computational Linguistics.

[9] Karthiga, M., Priyadharsini M.: A Semantic Search Engine using Semantic Similarity Measure between Words. In: International Journal of Scientific and Engineering Research. 4.5 (2013): 379-384.

[10] Karthiga M., Kalaivaani PC., Sankarananth S.A semantic similarity approach based on web resources. In2013 International Conference on Information Communication and Embedded Systems (ICICES).IEEE. 2013. Feb 2: 226-231.

[11] Darshan Agarwal et al. Semantic Textual Similarity For Hindi. Proceedings of 18th Inter-national Conference on Computational Linguistics and Intelligent Text Processing (CICLing-2017), Budapest, Hungary.

[12] Taieb M A H., Aouicha M B., Hamadou A B. A New Semantic Relatedness Measurement Using WordNet Features. Knowledge and Information Systems, vol. 41, no. 2, pp. 467-497, 2014.

[13] Kalchbrenner., Nal., Edward Grefenstette ., and Phil Blunsom.: A convolutional neural network for modelling sentences. arXiv preprint arXiv.2014:1404.2188.

[14] He., Hua ., Kevin Gimpel., and Jimmy Lin.: Multi-perspective sentence similarity modeling with convolutional neural networks. Proceedings of the 2015 conference on empirical methods in natural language processing. 2015.

[15] He., Hua., and Jimmy Lin.:Pairwise word interaction modeling with deep neural networks for semantic similarity measurement.Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016.

[16] Yao., Haipeng., Huiwen Liu., and Peiying Zhang.: A novel sentence similarity model with word embedding based on convolutional neural network. Concurrency and Computation: Practice and Experience 30.23 (2018): e4415.

[17] Srihari R K., Zhang Z F., Rao A B. Intelligent Indexing and Semantic Retrieval of Multi-modal Documents. Information Retrieval, vol. 2, pp. 245-275, 2000.

[18] Miller, G. A, Bechwith, R., Felbaum, C., Gross, D. And Miller, K. Introduction to WordNet: an on-line lexical database. International Journal of Lexicography, 3(4):235-244.(1990).

[19] Smeulders., Worring M., Santini S., Gupta A., Jain. Content-Based Image Retrieval at the End of the Early Years. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 12, pp. 1349-1380, Dec. 2000.