Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 26 (2023), No. 3, pp. 475–482
DOI : 10.47974/JIM-1675

Word sense disambiguation: Mathematical modelling of adaptive word embedding technique for word vector

Chandrakant D. Kokane *
Sachin D. Babar †
Department of Computer Engineering
Savitribai Phule Pune University
Smt. Kashibai Navale College of Engineering
Pune
Maharashtra
India

Parikshit N. Mahalle §
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Information Technology
Pune
Maharashtra
India

Shivprasad P. Patil ‡
Savitribai Phule Pune University
NBN Sinhgad School of Engineering
Pune
Maharashtra
India

Abstract

Word embedding is the method of representing ambiguous words into word vectors. The existing methods of word embedding are applicable for homonymous words. Constructing word vector of polysemous words is the challenge. The word vector of polysemous words are made by considering context information. The proposed adaptive word embedding technique is discussed in this article. The adaptive word embedding

* *E-mail:* **cdkoakne1992@gmail.com (Corresponding Author)**
† *E-mail:* **sdbabar@sinhgad.edu**
§ *E-mail:* **aalborg.pnm@gmail.com**
‡ *E-mail:* **principal.nbnsscoe@sinhgad.edu**

technique is applicable for both homosemous and polysemous words. While representing ambiguous word into word vector the context information is considered. The adaptive word embedding technique generates dynamic word vector for ambiguous word. The word vector with dimension size 198 is created here. There are 198 features are considered in the discussed model. The countable nouns are used as features in adaptive word embedding.

Subject Classification: *Primary 93A30, Secondary 49K15.*

Keywords: *Context information, Semantic similarity, Polysemous words, Adaptive word embedding.*

1. Introduction

Word embedding is the process of representing the words into word vector in such a way that machines can understand the meaning of word [1]. Machine understands only binary language so in adaptive word embedding the textual data is represented in binary form so that machine will understand [2]. The ambiguous word is represented in vector by considering features. The word vector is further used for the training Recurrent Neural Network(RNN) model so that machines will get an exact and accurate meaning of the word. As shown in figure-1, words in Machine Translation/Natural Language Processing(NLP) words are categorized into two categories as,

Polysemous words have multiple meanings but have the same spelling and pronunciations [3]. Homonymy words are easy to disambiguate because the word vectors of such words are static. Disambiguation of polysemous words is very difficult because the same words may have multiple vector representations [4]. The main role of the disambiguation process with respect to context information: consider available context information, construct the word vector by considering context and use the same word vector for the training [8,9].

The article is subdivided into sections as Section-I explains the introduction of adaptive word embedding. Section- II describes the state of

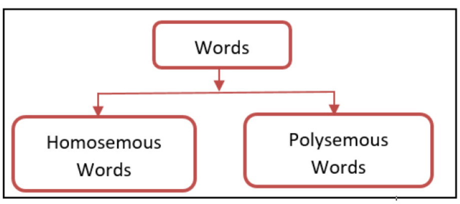

Figure 1

Classification of words

the art word embedding techniques and gaps identified by existing word embedding techniques. Section-III elaborates proposed adaptive word embedding models. Section-VI elaborates Mathematical foundations. Section- V describes the result of adaptive word embedding technique

2. Simple Word Embedding Models

The literature review is done with respect to available word embedding techniques. Figure-2 shows the classification of word embedding techniques as context dependent and context independent described below,

a. *One Hot Encoding*

One Hot encoding technique is context dependent word embedding technique. The vector of the word is constructed by marking the index position of the word as 1 and the rest of the index is marked as 0 [2].

MAHASHRATRA= [1,0,0,0,0,0,0,….]. The word GOA is present at the 1st index so the word vector for GOA is GOA= [0,1,0,0,0,0,0,…].

b. *TF-IDF*

This technique of word embedding works with respect to the statistical analysis of the word to determine the mathematical significance in the document. The construction of word vectors is quite similar to the one hot encoding technique but the TF-IDF values of words are used instead of 0 and 1[3].

First TF (Term Frequency) value is calculated as, Ratio of number of ambiguous words in the sentence/total number of words in the sentence (1)

TF[Bank] = [0.33 0.33 0.33], TF[is]= [0.33 0.33 0.33] and TF[or]= [0.20 0.20 0.20 0.20 0.20] [3,4]

c. *IDF - Inverse document frequency:*

IDF is the logarithmic ratio of Log (Total number of sentences in which the ambiguous word present) to the total number of documents [5]. The IDF values for "bank", "is" and "or" are calculated as,

bank=log (3/3) =0 [1.,1.1761,1.4771,0,0,0,0] (2)

is=log (3/2) =0.1761 [1.,1.1761,0.,1.4771,0,0,0,0] (3)

or=log (3/1) =0.4771 [1.,0.,0.,0.,1.4771,1.4771,1.4771,1.4771] (4)

"Bank is open", "Bank is open working" "Bank is open working isn't functioning or closed".

d. *Word2Vec:*

This technique first scans the whole document and base on the features the word vector for the target word is prepared. More than 800 features are considered and used for word Embedding [6].

This technique generates efficient word vectors for the words. "King", "MAN" and "KING" [9].

The man with the crown is king "KING" – "MAN" = "QUEEN". The gender of "Queen" as Gender = Female and Crown = 1 that is female with crown is "Queen".

This word embedding performs well for homonymous words to identify the synonyms and antonyms [10].

e. *BERT (Bidirectional Encoder Representations from Transformers)*

This technique generates two or multiple vectors for the same words and moves with both the directions [5]. The proposed methodology generates dynamic word vectors for the words [6]. BERT Generate word vector of 768-dimension Drawback it is very compute intensive at inference time [7,8].

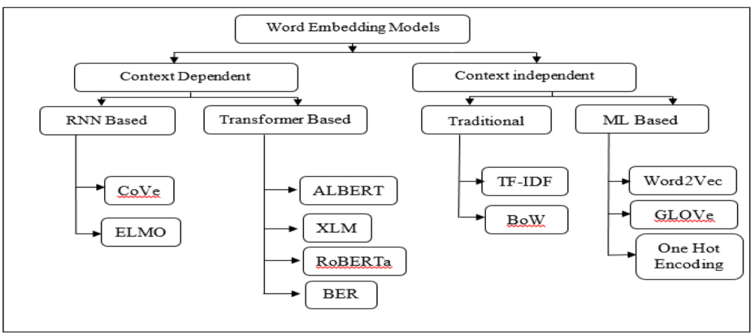

Figure 2

Word Embedding Models

2.1 *Analysis of word embedding techniques*

- The IF-IDF, Term frequency method generates the word vector by considering the statistical information.

- The IDF generates the logarithmic values for the word. The IF-IDF requires some normalisation burden for the word embedding.
- The Word2Vector method generates word2Vector by considering the features of the word.

3. Mathematical Foundation

In this section the mathematical relations are discussed. Adaptive word embedding technique builds HyperTree of ambiguous words with Lowest Common Subsumers (LCS) [9]. As shown in T1, the HyperTree represents the word with all possible meanings of the word "bank". Similarly, T2 shows the HyperTree for the word "saving". The Similarity score of the words "bank" and "saving" is 0.6 on the scale of 10. The similarity score represents the association of the words [10]. The Adaptive word embedding model has a similarity threshold T1 is 60% [11].

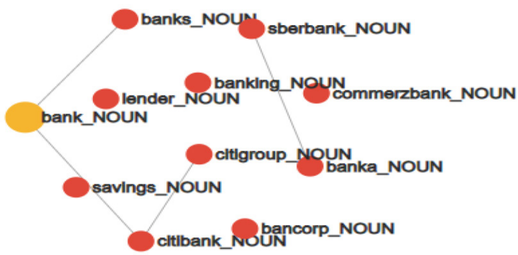

Figure 3

Representation of top similar words

Figure-3 shows the node presentation of words along with similarity score. The minimum edge distance describes the closest word whereas the longest distance describes the word is far closer. The adaptive word embedding model dynamically selected the features to represent the word in word vector.

4. Adaptive Word Embedding Model

As shown in figure-4, The proposed methodology accepts ambiguous documents as an input. The pre-processing for every individual sentence is performed. The output of the pre-processing step is the purest form of the sentence. While detecting ambiguous words the sense mapping mechanism is used the sense of individual words mapped with freely available lexical resource WordNet [12,13]. Semantic similarity of

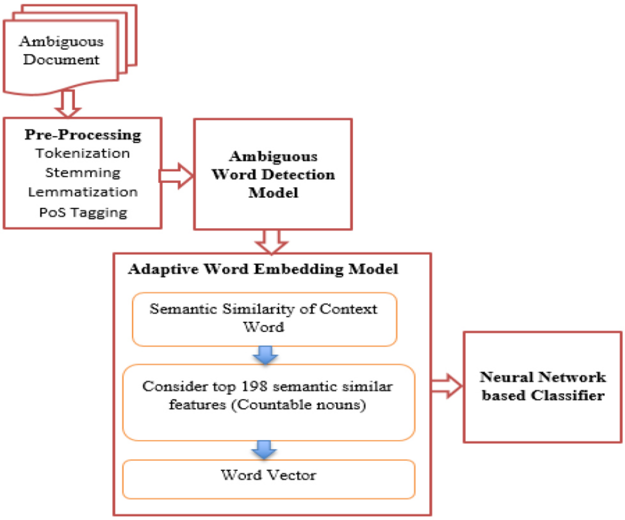

Figure 4

Adaptive WSD Framework

Ambiguous word and context information is calculated and based on the same 198 features from the countable noun are selected to generate word vectors.

5. Discussion and Conclusion

Table 1

Comparative result of word embedding techniques

Word2vector	One Hot Encoding	Skip Grams	BERT	Adaptive Word Embedding
Applicable for Homonymy	Applicable for Homonymy	Applicable for Homonymy	Applicable for Polysemous	Applicable for both Homonymy and Polysemous word
Static vector is generated	Static vector is generated	Static vector is generated	Dynamic vector generated	Dynamic vector generated
Processing time is average	Processing time is adequate	Processing time is adequate	Processing time is more	Processing time is Efficient

Table-1 shows the comparative result analysis of the word embedding techniques. The proposed adaptive word embedding model generates multiple representations of word vectors with respect to semantic similarity based context information. In the future this adaptive model of word embedding can be suitable for WSD problems in large documents.

References

[1] D. Park, S. Kim, J. Lee, J. Choo, N. Diakopoulos and N. Elmqvist, "Concept Vector: Text Visual Analytics via Interactive Lexicon Building Using Word Embedding," in IEEE Transactions on Visualization and Computer Graphics, vol. 24, no. 1, pp. 361-370 (2018).

[2] M. Berger, K. McDonough and L. M. Seversky, "cite2vec: Citation-Driven Document Exploration via Word Embeddings," in IEEE Transactions on Visualization and Computer Graphics, vol. 23, no. 1, pp. 691-700 (2017).

[3] G. Iovane, L. Sensini, P. Di Gironimo, A. Rapuano & A. Briscione. TES: Token evaluation system for block chain contexts, *Journal of Interdisciplinary Mathematics*, 25:8, 2093-2111 (2022), DOI: 10.1080/09720502.2021.1932852.

[4] Q. Jiao and S. Zhang, "A Brief Survey of Word Embedding and Its Recent Development," 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 1697-1701 (2021).

[5] S. Liu et al., "Visual Exploration of Semantic Relationships in Neural Word Embeddings," in IEEE Transactions on Visualization and Computer Graphics, vol. 24, no. 1, pp. 553-562 (2018).

[6] W. Ding, C. Lin and P. Ishwar, "Node Embedding via Word Embedding for Network Community Discovery," in IEEE Transactions on Signal and Information Processing over Networks, vol. 3, no. 3, pp. 539-552 (2017).

[7] Ramesh Chandra Poonia, Joseph Varghese Kureethara, Kashinath Chatterjee, Vijander Singh, Linesh Raja & Md. Ashraful Babu (2022) Guest Editors, *Journal of Interdisciplinary Mathematics*, 25:7, ii, DOI: 10.1080/09720502.2022.2156736.

[8] C. D. Kokane, S. D. Babar and P. N. Mahalle, "Word Sense Disambiguation for Large Documents Using Neural Network Model,"

2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, pp. 1-5 (2021), doi: 10.1109/ICCCNT51525.2021.9580101.

[9] Chang-Jin Guo, Feng Pan & Yi Zuo, "Word similarity algorithm based on multi-features, *Journal of Interdisciplinary Mathematics*", 21:5, 1067-1072 (2018), DOI: 10.1080/09720502.2018.1493031

[10] Tung-Ju Wu, Rui Sun & Zi-Fen Lin, "The empirical research of the relationship between internal social capital and relational psychological contract and employee' innovation behavior", *Journal of Interdisciplinary Mathematics*, 19:4, 823-841 (2016), DOI: 10.1080/09720502.2016.1225933

[11] Medhat EI-Gharib Metaweh, "On the Art of Modeling", *Journal of Interdisciplinary Mathematics*, 4:2-3, 165-177 (2001), DOI: 10.1080/09720502.2001. 10700298.

[12] Vishal Pradhan, Ajay Kumar & Joydip Dhar, "Enhanced growth model of software reliability with generalized inflection S-shaped testing-effort function", *Journal of Interdisciplinary Mathematics*, 25:1, 137-153 (2022), DOI: 10.1080/09720502.2021.2006329.

[13] Kokane, C.D., Babar, S.D., Mahalle, P.N. "An Adaptive Algorithm for Lexical Ambiguity in Word Sense Disambiguation". In: Patil, V.H., Dey, N., N. Mahalle, P., Shafi Pathan, M., Kimbahune, V.V. (eds) Proceeding of First Doctoral Symposium on Natural Computing Research. Lecture Notes in Networks and Systems, vol 169 (2021). Springer, Singapore. https://doi.org/10.1007/978-981-33-4073-2_11