Semantic Similarity Measure Using a Combination of Word2Vec and WordNet Models

Aissa FELLAH[1], Ahmed ZAHAF[2], Atilla ELçi[3]
[1,2]Department of Computer Science, University of Saida Dr. Moulay Tahar, Saida, Algeria
[3]Software Engineering Dept. Hasan Kalyoncu University, Gaziantep, Türkiye

Article Info

Article history:

Received Sep 22, 2023
Revised May 5, 2024
Accepted Jun 30, 2024

Keyword:

Word2Vec
WordNet
Semantic Similarity Relatedness
Word Embedding

ABSTRACT

The cognitive effort required for humans to perceive similarities and relationships between words is considerable. Measuring similarity and relatedness between text components such as words, texts, or documents is challenging, and it continues to be an active area of research across various domains. The complexity of language and the diverse factors that influence similarity and relatedness make this task an ongoing research focus. Researchers are exploring diverse approaches, to improve the accuracy and effectiveness of measuring similarity and relatedness in text. The utilization of knowledge sources, such as WordNet, has been a popular approach for modeling semantic relationships between words. However, Recently, distributional semantic models, such as Word2Vec, have demonstrated their ability to effectively capture semantic information and outperform lexicon-based methods in terms of unidirectional contextual similarity outcomes. In contrast to lexicon-based approaches, which rely on structure, distributional models leverage context to capture semantics. This study proposes a novel approach that linearly combines the lexical databases WordNet and Word2Vec to measure semantic similarity, focusing on improving upon previous techniques. The proposed approach is thoroughly detailed and evaluated using popular datasets to determine its effectiveness. The experimental results indicate that the proposed approach achieves highly satisfactory results and surpasses the performance of individual methods.

Corresponding Author:

Aissa FELLAH
Department of Computer Science
University of Saida Dr. Moulay Tahar
Saida, Algeria
ammfellah@gmail.com

1. INTRODUCTION

Finding semantic similarities between different text components, including words, texts, or documents, is arduous. Although it has become an essential part of a wide variety of applications, it is a hard problem to solve, old but unfortunately still very topical and it "is one of the open research problems" [1]. The objective is to measure the relationship between texts, sentences, and words to depict their degree of similarity or resemblance. Semantic Textual Similarity deals with determining how close two text components are. Semantic similarity algorithms conventionally provide a degree or rate of the text components' resemblance. Frequently, semantic similarity and semantic relatedness can be used interchangeably. Additionally, when evaluating semantic relatedness, the common semantic attributes of two words are also considered [2]. There is an overwhelming need for similarity assessment in various computer applications. Human evaluation of word likeness uses multiple cognitive abilities. The accomplishment of such actions by the machine is a tedious task. "The field of artificial intelligence, information retrieval, and natural language processing has seen a lot of research activity in the proposal of methods to estimate the degree of similarity and relatedness between words and concepts” [3]. References [4, 5, 6] are examples of works; an excellent survey is presented in [1]. The similarity calculation is the proposal of automatic methods converge to human appreciation.

Let us consider these two words: Word1= ‘Liquid’ and Word2= ‘Water’. Human’s estimate of the similarity between these words is 0.70625, according to the WS353-all [7] dataset.

We seek to propose a measure that converges to human evaluation. WordNet has been widely used as a knowledge ontology, by most semantic similarity algorithms [8, 9, 10, 11], due to its clear structure. Each word ‘liquid’ and ‘water’ has several synsets in WordNet. The maximum Wu & Palmer [8] similarity measure is 0.9333; that value is far from human estimation. Studies like Elekes et al. [12] have shown that modern distributional semantic models, namely the famous Word2Vec proposal from [13], can apprehend the sense and have shown promising results; however, they may not always outperform knowledge approaches in measuring semantic similarity. The measure of similarity between the words 'liquid' and 'water' by calculating the cosine of the vectors representing the two words in Google's news Word2Vec pretrained mode [14] is 0.3653; which is even less close to human estimation, according to [15]. Considering performance on standard similarity datasets, several techniques based on WordNet have been found to outperform the Word2Vec method. The major challenge for research on semantic similarity is to propose techniques to improve computational accuracy [16, 17, 18, 19, 20, 21]; a promising way is to exploit the advantages of WordNet and Word2Vec. However, several techniques have combined the best of each method to improve semantic similarity measurement. Our main motivation in this research is to combine semantic similarity estimation methods between words. To steer this research, the following two questions will be considered:

- Q1: What is the current standing of WordNet and Word2Vec methods in word similarity evaluation?
- Q2: Is it possible to surpass the individual achievement of Word2Vec and WordNet methods via a linear combination of these techniques?

This study presents a novel technique for calculating semantic similarity that combines word embeddings and a lexical database through a linear combination, which utilizes a dynamic weighting coefficient that considers the combined measurements. To assess the effectiveness of our technique, we use Spearman's and Pearson's correlation coefficients to compare our assigned semantic similarity scores against those of human judgments. The rest of the paper is organized thereby: the following section bestows a quick summary of the techniques employed in the present work; Section III Summarily presents related works; in Section IV, we introduce our approach; the experimental details and datasets used are described in Section V. Finally, we conclude the paper with findings and suggestions of certain indications for future works.

2. Background

In this section, we present a basic review of the WordNet and Word2Vec techniques used in this research.

2.1. WordNet and Lexical Semantic Similarity Measures

WordNet, created by Princeton University, is a significant lexical resource for various uses [22, 23]. It is a lexical database that consists of hundreds of thousands of English concepts; it “can be represented as a graph where the nodes correspond to the meanings of words or concepts, where the edges signify the connections between them” [24]. The fundamental component of WordNet is the synset or set of synonyms, which is a collection of interchangeable words that denote a particular meaning. The path distance is the basis of the similarity computation [25, 19], commonly using topological similarity existing within the ontology, which in this case, is WordNet. The method used in our work is Wu and Palmer[8], which exploits WordNet. This method provides the similarity value between 0 and 1.

Let’s consider the two words, $\mathrm{W_1}$ and $\mathrm{W_2}$, their Least Common Subsumer (LCS) is denoted by $\mathrm{W_{lcs}}$. The Wu and Palmer similarity measure (Sim_WP) is then computed using Formula 1.

$$\mathrm{Sim\_WP}(\mathrm{W_1},\mathrm{W_2}) = \frac{2*\mathrm{depth}(\mathrm{W_{lcs}})}{\mathrm{depth}(\mathrm{W_1})+\mathrm{depth}(\mathrm{W_2})} \tag{1}$$

where, $\mathrm{depth}\,(\mathrm{W_1})$ is the number of arcs between the concept of term $\mathrm{W_1}$ and the ontology (that is, WordNet) root.

2.2. Word2Vec and the Contextual Similarity Measures

Word Embedding aims to enable machines to better understand words by providing vector representations of words that capture the relationships between them. These vectors are obtained using various methods, including neural networks such as Word2Vec [13, 26]. Word2Vec is a popular neural network model that generates a distributed vector representation of words based on a given corpus. It has two variations: (1) Continuous Bag-of-Words and (2) SkipGram, consisting of a single hidden layer, making them computationally efficient during training. Continuous Bag-of-Words predicts the central word based on a nearby window of words, while SkipGram predicts the context based on the central word. According to [1], Word2Vec models effectively represent words as vectors while maintaining contextual similarity and providing accurate semantic similarity predictions. Word2Vec [14] is one of the most widely used pre-trained word embeddings, containing vector representations of around 3 million words and phrases, developed from the Google News dataset.

3. Related Works

We distinguish among four main methods to calculate semantic similarity, namely, corpus-based, knowledge-based, deep neural network-based, and hybrid approaches; these are briefly introduced below.

Corpus-based approaches: Methods that rely on investigation into huge corpus to calculate the semantic similarity of term pairs. The basic idea is that words that appear in interchangeable contexts probably tend to have the same meanings. The principal techniques are Point Mutual Information [27], Latent Semantic Analysis [28], Word-alignment models [29], Explicit Semantic Analysis [30], Normalized Google Distance [31], Kernel-based models [32], and Word-attention models.

Knowledge-based approaches: To compute the semantic similarity between words, knowledge sources are used, including general-purpose ontologies such as WordNet [22], SENSUS [33], Cyc [34], BabelNet [35], and domain-based ontologies such as UMLS [36] and MeSH [37]. Ontology-based approaches are broadly classified into edge-based [8, 9], information content-based [10], and feature-based approaches [11].

Deep neural network-based approaches [38, 39] have been motivated by the latest progress in neural networks; they have good outcomes. Non-exhaustively, the most used techniques are Long Short Term Memory [40], Bidirectional Long Short Term Memory [41], Convolutional Neural Networks [42], and BERT [43].

Hybrid approaches [44] are based on a combination of the methods mentioned above. Each technique has unique advantages and disadvantages that will make it preferable for some cases but not others. The emergence of hybrid techniques provides the possibility of combining them to obtain the best of each of them to measure semantic similarity. Our proposed technique is classified in this category, that is, in the class of methods that combine WordNet and Word2Vec.

The literature contains several studies utilizing WordNet and word embedding models to measure semantic similarity. In one such study, Lee et al. [17] calculate the semantic relatedness score of two words by combining the cosine similarity between their embedding vectors and the path distance between their corresponding SynSets in WordNet. The authors determine the weighting coefficient through a heuristic search with a step size of 0.05 over the parameter range of 0 to 1. Although this method is like our approach, we differ in two key factors, namely, we utilize Wu and Palmer's measure for the WordNet component, and our weighting coefficient is dynamically calculated for each case. In another research, Qu et al. [15], first generated continuous representations for each word sense, they then computed the similarity between two given words by comparing the sense embedding vectors, which were obtained using BabelNet [35] as the knowledge base and the September 2014 English Wikipedia dump corpus. For Word Sense disambiguation, they used Babelfy4. Finally, they utilized Word2Vec to create continuous representations for word senses. In the study by Rothe and Schütze [18], they introduced AutoExtend, a method that merges word embeddings with semantic resources. AutoExtend accomplishes this by learning embeddings for synsets, entities, and words that integrate semantic information from various sources such as WordNet, GermaNet, and Freebase. Unlike other methods, AutoExtend employs tensors without any other knowledge resource. Sugathadasa et al. [19] employed a domain-specific semantic similarity measure that combines Word2Vec, a word embedding technique for computing semantic similarity, with lexicon-based (lexical) semantic similarity methods. They argue that using a combination of word vector embedding and lexical semantic similarity measures provides a more precise assessment of the degree to which two words are semantically similar within the particular domain being studied. Lee et al. [45] present three distinct techniques to assess the semantic relationship of a pair of words. Firstly, they enhance the performance of the GloVe word-embedding model by either transforming or removing abnormal dimensions. Secondly, they combine the information extracted from WordNet and word embeddings using a linear approach. In the final analysis, they use word embeddings in conjunction with WordNet-extracted linguistic features to do a Vectorial Regression. Li et al. [20] proposed a semantic similarity method that combines the "is-a" semantics of WordNet with the link semantics of Wikipedia using new aggregation schemas. In contrast, Orkphol and Yang [46] used Word2Vec to generate context sentence vectors and sense definition vectors for each word sense. They then assigned a score to each sense using cosine similarity and expanded the sense definition by retrieving sense relations from WordNet. If the score did not exceed a certain threshold, they combined it with the probability distribution of that sense, which was learned from a large sense-tagged corpus. Hussain, Bai, and Jiang [21], introduce a novel method for measuring semantic similarity. The approach is based on multiple inheritances, and it defines the category semantic space, from the Wikipedia graph, by using its neighborhood. Next, the semantic value of a category is calculated by combining the inherent semantic contributions based on the information content of its multiple semantically relevant ancestors.

The common principle among all these approaches is to enhance the results of existing methods by combining different techniques to take advantage of their unique abilities to measure semantic similarity. However, there are differences in their test conditions and approaches to integration. It is important to note that in all the works the results are inconclusive and the problem of semantic similarity measurement remains open and relevant. Future contributions in this field are highly sought after.

4. Methods

Our study aims to propose a new approach to determine semantic similarity using Word2Vec and WordNet through a linear aggregation. To explain this approach, we first provide a simple example. Additionally, we elaborate further on our proposal in detail. Our preliminary results indicate that our method outperforms using only one of the contextual and structural similarity approaches.

4.1. Example

As mentioned in the introduction, according to the WS353-all [7] dataset, Human's estimate of the similarity between these two words Word1= ‘Liquid’ and Word2= ‘Water’ is 0.70625.

The two words, ‘liquid’ and ‘water’, each has several synsets in WordNet. The maximum Wu & Palmer [8] similarity measure is 0.9333. The measure of similarity by calculating the cosine of the vectors representing the two words in Google’s news Word2Vec pre-trained model is 0.3653.

The results obtained from both measures are not accurate enough when compared to human evaluation. Intuitively, by combining the first (structural) and the second (contextual) measurements with an appropriate weighting coefficient, it is possible to obtain a similarity value closer to the human estimate!

In the next section, we will explain our suggested solution the Aggregated Semantic Similarity Measure (ASSM), surpassing the individual achievement of Word2Vec and WordNet techniques and computing closer to the human estimate.

4.2. Our Proposed Approach

To capture the strength of the contextual link between two words, say W_1 and W_2, each is represented by a dense vector in the Word2Vec model, firstly, we calculate the similarity between them. Incidentally, it is possible to train Word2Vec on any corpus of text, however, this is not done here. Rather than developing our word embeddings, we utilize the widely popular Google's Word2Vec Pretrained Word Embedding [14], which was trained on the vast Google News dataset, consisting of approximately 100 billion words. The Word2Vec model has numerous applications, such as recommendation engines, knowledge discovery, and text classification tasks. Then, we exploit WordNet to compute the structural link between two words with Wu & Palmer similarity (Equation 1).

Finally, we compute a linear combination of the two measurements. Several experiments have led us to choose the arithmetic mean between the two measurements (i.e., Word2Vec measurement and Wu & Palmer on WordNet) as a weighting coefficient ($\boldsymbol{\alpha}$) in (4) for the two measurements. Formally, the proposed Aggregated Semantic Similarity Measure (ASSM) of two words using semantic similarity measures that combine Word2Vec and WordNet is defined as follows:

$$\mathbf{ASSM}(\mathbf{W_1}, \mathbf{W_2}) = \mathrm{Max}\begin{pmatrix}\mathrm{Sim\_W2V}(\mathrm{W_1}, \mathrm{W_2}) \\ \beta\end{pmatrix} \tag{2}$$

Where:

$$\boldsymbol{\beta} = \alpha\, \mathrm{Sim\_WP}(\mathrm{W_1}, \mathrm{W_2}) + (1-\alpha)\, \mathrm{Sim\_W2V}(\mathrm{W_1}, \mathrm{W_2}) \tag{3}$$

$\mathbf{Sim\_WP}(\mathbf{W_1}, \mathbf{W_2})$ is the WordNet Wu & Palmer similarity between two terms $\mathrm{W_1}$ and $\mathrm{W_2}$, in (1).
$\mathbf{Sim\_W2V}(\mathbf{W_1}, \mathbf{W_2}) = \mathrm{Cos}(\mathrm{EV_{W1}}, \mathrm{EV_{W2}})$ is the cosine similarity between words and represented by embedding vectors $\mathrm{EV_{W1}}$and $\mathrm{EV_{W2}}$, using Google’s Word2Vec. Instead of utilizing a constant parameter as the weight coefficient, we opt for a flexible parameter as follows. This flexible parameter, known as a dynamic weighting factor, is computed in the following manner:

$$\boldsymbol{\alpha} = (\mathrm{Sim\_WP}(\mathrm{W}_1,\mathrm{W}_2) + \mathrm{Sim\_W2V}(\mathrm{W}_1,\mathrm{W}_2))/2 \tag{4}$$

The algorithm Algo 1 shown below represents the pseudo-code of our semantic similarity calculation method.

Application to our example mentioned in Sections 1 and 4.1 is as follows:
$\mathrm{W}_1 = \text{'liquid'}$ and $\mathrm{W}_2 = \text{'water'}$
$\mathrm{Sim\_W2V} = (\mathrm{W}_1,\mathrm{W}_2) = 0{,}3653$
$\mathrm{Sim\_WP}(\mathrm{W}_1,\mathrm{W}_2) = 0{,}9333$
$\alpha = (0{,}9333 + 0{,}3653)/2 = 0{,}6493$
$\beta = 0{,}6493 * 0{,}9333 + 0{,}3507 * 0{,}3653 \;=\; 0{,}7341$
$\mathrm{ASSM}(\mathrm{W}_1,\mathrm{W}_2) = \mathrm{Max}\begin{pmatrix}0{,}3653\\0{,}7341\end{pmatrix} = 0{,}7341$
is our Aggregated Semantic Similarity Measure (ASSM).

According to the WS353-all [7] dataset, Human evaluation between the two words is 0.70625. Therefore, the result of ASSM exhibiting a higher correlation with human judgments is typically regarded as more accurate and reliable.

Algo 1 pseudo-code below spells out the detail and explanation of our similarity measurement algorithm. Our experiments with several test datasets to evaluate the effectiveness of the ASSM method are covered in the next section.

Algo. 1. The pseudo-code of our similarity measurement algorithm

```
   //Pseudo-code of the proposed semantic similarity calculation method
   Algorithm ASSM
   //Two words that need to be compared for semantic similarity
    Input: $W_1$ word, $W_2$ word
   // Two pre-trained models used to calculate the semantic similarity
    Parameters: WordNet, Word2Vec_Google : Models
            $\alpha$: Real
    //The final semantic similarity score that will be returned by the algorithm
    Output: ASSM  Aggregated Semantic Similarity Measure $\in [0,1]$
   //Check if both words are present in the Word2Vec model
    If  $W_1$ in Word2Vec_Google and $W_2$ in Word2Vec_Google:
   $\mathbf{EV_{W_1} = Word2Vec\_Google.Vec(W_1)}$
   $\mathbf{EV_{W_2} = Word2Vec\_Google.Vec(W_2)}$
   // Calculate semantic similarity using the Cosine_Similarity function
   $\mathbf{Sim2V(W_1,W_2) = Cos(EV_{W1},EV_{W2})}$
// Check if both words are present in the WordNet model
   If $W_1$ in WordNet and $W_2$ in WordNet :
   //Calculate semantic similarity using the Wu & Palmer Similarity function
   For each $\mathrm{Synset_i}$ in $\mathrm{Synsets}(W_1)$
           For each $\mathrm{Synset_j}$ in $\mathrm{Synsets}(W_2)$:
   // Calculate semantic similarity using the Wu & Palmer Similarity function between //every synsets couple
        $\mathbf{Sim\_Vect\_WP[k] = Sim\_WP(Synset_i\ ,Synset_j)}$
   //Calculate maximum semantic similarity using Wu & Palmer Similarity
      $\mathbf{Sim\_WP = Max(Sim\_Vect\_WP)}$
   //If one of the two words is not present in the WordNet model, Set the Wu & Palmer //Similarity score to 0
   Else   Sim_WP = 0
   //If one of the two words is not present in the Word2Vec model, Set the Word2Vec //Similarity score to 0
   Else Sim_W2V = 0
   //Calculate the average of the two similarity scores
   α = (Sim_W2V+Sim_WP) / 2
   //Calculate the final Aggregated Semantic Similarity Measure by taking the maximum
   //similarity score between Cosine_Similarity and a weighted sum of Cosine_Similarity and
   //Word2Vec Similarity
   ASSM = max(Sim_W2V , (Sim_W2V*(1-α)) + (Sim_WP*α))
   //Return the final Aggregated Semantic Similarity Measure
   Return(ASSM)
   End
```

5. Experiments

Next, we present the achievement of our ASSM algorithm and compare it against those of several other established approaches as reported in the literature.

5.1. Description

We aim to compare the effectiveness of word similarity measurements using established evaluation benchmarks and datasets. The evaluation process entails computing similarity values for each word pair in each set and comparing them with human evaluations. In our experiments, we leveraged three powerful Python libraries: Gensim(Generate Similar)[50] for its efficient and fast vector embedding creation, NLTK(Natural Language Toolkit Library)[51] for natural language processing tasks, and SciPy(Scientific Python)[52] a free and open-source library for scientific computing. The findings of the experiments are detailed below.

5.2. Datasets

To evaluate the efficaciousness of our method, we conducted tests on widely utilized datasets commonly used as reference points in similar contexts. These evaluation benchmarks are listed in Table 1 for ready reference.

Table 1. Popular benchmark datasets for semantic similarity [1]

Name	Number of pairs	Scale	Year	Paper
R&G	65	0:4	1965	[54]
WS353-all	353	0:10	2002	[7]
WS353-Sim	203	0:10	2009	[49]
MC-30	30	0:4	1991	[53]
AG-203	203	0-10	2009	[49]

A brief introduction to those datasets is given below:

- The R&G-65 [54] is a test collection proposed in 1965; it contains 65 wordpairs. Each pair's similarity is rated on a scale ranging from 0 to 4 (a greater numerical value corresponds to a higher degree of similarity) [48]. The dataset's similarity values represent the average ratings provided by 51 human participants.
- WS353-all [7] is a dataset for the similarity or relatedness of words. The similarity of each pair is scored on a scale of 0 to 10 (the higher the "similarity in meaning", the higher the number) [16].
- The original WS-353 dataset [49] conflates similarity and relatedness; it is divided into two subsets, each containing pairs for just one type of association measure: similarity (the WS-Sim dataset) and relatedness (the WS-Rel dataset).
- MC-30 [53] is a dataset for the similarity or relatedness of words. Each pair's similarity is rated on a scale ranging from 0 to 4.
- AG-203 [49] is a collection of similarity or relatedness of words, developed and maintained by Eneko Agirre. The similarity of each pair is scored on a scale of 0 to 10 (the higher the "similarity in meaning", the higher the number).

5.3. Evaluation Metric

To measure the accuracy of semantic similarity measurements, we calculated the Spearman's (ρ) and Pearson (r) correlation coefficients between the similarity values (X) computed by algorithms on the benchmark datasets and the corresponding human judgment scores (Y). This evaluation process is outlined as follows:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{5}$$

$$d_i = \mathrm{rank}(X_i) - \mathrm{rank}(Y_i) \tag{6}$$

Where:

n : the number of word pairs of the benchmark.

X_i : is the value of human evaluation for the ith word pair in the benchmark dataset.
Y_i : is the value of the semantic measure for the ith word pair in the benchmark dataset returned by our method.
d_i: is the difference between the two ranks of X_i and Y_i.
$\mathrm{rank}(X_i)$: returns the rank of X_i in a list X.

And,

$$r = \frac{n\sum X_iY_i - \sum X_i \sum Y_i}{\sqrt{n\sum X_i^2 - (\sum X_i)^2}\sqrt{n\sum Y_i^2 - (\sum Y_i)^2} \tag{7}$$

n : is the number of word pairs in the benchmark dataset.

X_i: is the human judgment scores of the ith word pair in the benchmark dataset.

Y_i: is the value of the semantic measure for the ith word pair in the benchmark dataset returned by our method.

5.4. Results and Discussion

Several tests were performed. Our results are split into two tables to standardize and bring the comparisons into conformity with other published works according to their experiments and the data sets used. We have normalized the values of human judgments for each dataset employed in these tests. shown

Table 2. Spearman's (ρ) and Pearson (r) correlation of similarity measures on datasets (RG-65), (WS353-all) and (WS353-sim).

	RG-65	WS353-all	WS353-sim
Method	ρ/r	ρ/r	ρ/r
Word2Vec	0.760/0.772	0.693/0.686	0.778/0.770
Our approach	**0.865/0.895**	**0.695/0.708**	0.793/0.814
[17]	0.873/0.830	0.707/0.656	0.812/0.793
[16]	0.871/None	0.714/None	0.756/None

Table 3. Pearson (r) correlation of similarity measures on datasets RG-65), (MC-30), and (AG-203).

Method	RG-65	MC-30	AG-203
WordNet [11]	0,87	0,85	0,63
Wordnet [47]	0,86	0,84	0,63
Word2Vec	0.772	0.786	0.770
Our approach	**0.895**	**0,926**	**0.814**
[21]	0.688	0.778	-
[20]	0,82	0,88	0,720

Table 2 illustrates the results of the first part of the experiments in comparison with [16] and [17] approaches. We evaluated the results using Spearman's (ρ) and Pearson (r) correlation coefficients. It can be observed that our method improves the Word2Vec results and outperforms both [16] and [17] approaches.

Table 3 illustrates the results of the second part of the experiments compared with [20] and [21] approaches. We evaluated the results using only the Pearson (r) correlation coefficient. Pearson's coefficient of our approach (indicated by the bold typeface) clearly shows superior performance.

A linear combination of WordNet and Word2Vec methods with a good intuition of choice of the weighting coefficient allowed us to obtain very satisfactory results and consequent improvement over each method alone.

The performance is due to the measure that exploits both the power of Word2Vec to capture the contextual aspect and versatility of WordNet for structural quality.

6. Conclusion

The approach we proposed is an innovative technique for calculating the semantic similarity of word pairs: linearly combining the outcomes of Word2Vec and WordNet models we can effectively capture the nuances and complexities of word meanings, resulting in more precise similarity measurement prediction. We believe this method has the potential to be used in a wide variety of applications. We evaluated our approach on the popular datasets RG-65, WS353-all, WS353-sim, MC-30, and AG-203. The experimental results show that aggregating the contextual dimension using the WordNet model and the structural dimension using the Word2Vec model is beneficial in measuring semantic similarity. For our research questions, thus we have covered the research question Q1 in sections 2 and 3.

We can now answer our research question Q2: Is it possible to surpass the individual achievement of Word2Vec and WordNet methods via a linear combination of the outcomes of these techniques? Apparently, a linear combination of the outcomes of the two models significantly boosts the resultant performance. The absence of some words in WordNet and the Word2Vec models posed a problem and can be considered a natural limitation of our approach.

Our work will likely see further advancements soon, as we explore using other embedded vector models such as GloVe, and investigate the potential benefits of incorporating additional lexical databases and ontologies. A key strength of our approach is its flexibility, as it can easily adapt to different word embeddings, whether unidirectional like GloVe, or bidirectional like BERT. Additionally, it may be possible to develop similar hybridization approaches that combine corpus knowledge and deep neural network-based methods, potentially leading to even more impressive results than those achieved by our current approach. Our research demonstrated the potential for continued innovation and highlighted the importance of exploring new techniques and models to improve our understanding and analysis of human language.

Leveraging our novel measurement technique, we are embarking on developing an API and applications that address critical challenges in data management: Semantic Service Discovery, Semantic Data Integration, and Ontology Matching.

REFERENCES

[1] D. Chandrasekaran and V. Mago, “Evolution of Semantic Similarity—A Survey,” ACM Computing Surveys, vol. 54, no. 2, pp. 1–37, Apr. 2021, doi: https://doi.org/10.1145/3440755.

[2] M. A. Hadj Taieb, T. Zesch, and M. Ben Aouicha, “A survey of semantic relatedness evaluation datasets and procedures,” Artificial Intelligence Review, vol. 53, no. 6, pp. 4407–4448, Dec. 2019, doi: https://doi.org/10.1007/s10462-019-09796-3.

[3] J. J. Lastra-Díaz, Josu Goikoetxea, M. Ali, A. García-Serrano, Mohamed Ben Aouicha, and E. Agirre, “A reproducible survey on word embeddings and ontology-based methods for word similarity: Linear combinations outperform the state of the art,” Engineering Applications of Artificial Intelligence, vol. 85, pp. 645–665, Oct. 2019, doi: https://doi.org/10.1016/j.engappai.2019.07.010.

[4] A. Fellah, M. Malki, and A. Elci, “A Similarity Measure across Ontologies for Web Services Discovery,” International Journal of Information Technology and Web Engineering, vol. 11, no. 1, pp. 22–43, Jan. 2016, doi: https://doi.org/10.4018/ijitwe.2016010102.

[5] Celik, Duygu, and Atilla Elçi. "Towards a Semantic Based Workflow Model for Composition of OWL-S Based Atomic Processes." Journal of internet Technology 12.1 (2011): 153-170. https://doi.org/10.6138/JIT.2011.12.1.15

[6] D. Çelik and A. Elçi, “A broker-based semantic agent for discovering Semantic Web services through process similarity matching and equivalence considering quality of service,” Science China Information Sciences, vol. 56, no. 1, pp. 1–24, Oct. 2012, doi: https://doi.org/10.1007/s11432-012-4697-1.

[7] Finkelstein, Lev, et al. "Placing search in context: The concept revisited." Proceedings of the 10th international conference on World Wide Web. 2001. https://doi.org/10.1145/371920.372094

[8] Wu, Zhibiao, and Martha Palmer. "Verb semantics and lexical selection." arXiv preprint cmp-lg/9406033 (1994). https://doi.org/10.3115/981732.981751

[9] R. Rada, Hafedh Mili, E. J. Bicknell, and M. Blettner, “Development and application of a metric on semantic nets,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 19, no. 1, pp. 17–30, Jan. 1989, doi: https://doi.org/10.1109/21.24528.

[10] D. Sánchez and M. Batet, “A semantic similarity method based on information content exploiting multiple ontologies,” Expert Systems with Applications, vol. 40, no. 4, pp. 1393–1399, Mar. 2013, doi: https://doi.org/10.1016/j.eswa.2012.08.049.

[11] Jiang, Jay J., and David W. Conrath. "Semantic similarity based on corpus statistics and lexical taxonomy." arXiv preprint cmp-lg/9709008 (1997).

[12] A. Elekes, M. Schaeler, and K. Boehm, “On the Various Semantics of Similarity in Word Embedding Models,” 2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL), Jun. 2017, doi: https://doi.org/10.1109/jcdl.2017.7991568.

[13] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013a). https://doi.org/10.48550/arXiv.1301.3781

[14] “Google Code Archive - Long-term storage for Google Code Project Hosting.,” code.google.com. https://code.google.com/archive/p/word2vec (accessed May 11, 2023).

[15] R. Qu, Y. Fang, W. Bai, and Y. Jiang, “Computing semantic similarity based on novel models of semantic representation using Wikipedia,” Information Processing & Management, vol. 54, no. 6, pp. 1002–1021, Nov. 2018, doi: https://doi.org/10.1016/j.ipm.2018.07.002.

[16] Iacobacci, Ignacio, Mohammad Taher Pilehvar, and Roberto Navigli. "Sensembed: Learning sense embeddings for word and relational similarity." Proceedings of the 53rd Annual Meeting of the Association for Computational

Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). 2015. https://doi.org/10.3115/v1/P15-1010

[17] Lee, Yang-Yin, et al. "Combining word embedding and lexical database for semantic relatedness measurement." Proceedings of the 25th international conference companion on world wide web. 2016. https://doi.org/10.1145/2872518.2889395

[18] S. Rothe and H. Schütze, "AutoExtend: Combining Word Embeddings with Semantic Resources," Computational Linguistics, vol. 43, no. 3, pp. 593–617, Sep. 2017, doi: https://doi.org/10.1162/coli_a_00294.

[19] K. Sugathadasa et al., "Synergistic union of Word2Vec and lexicon for domain specific semantic similarity," 2017 IEEE International Conference on Industrial and Information Systems (ICIIS), Dec. 2017, doi: https://doi.org/10.1109/iciinfs.2017.8300343.

[20] F. Li, L. Liao, L. Zhang, X. Zhu, B. Zhang, and Z. Wang, "An Efficient Approach for Measuring Semantic Similarity Combining WordNet and Wikipedia," IEEE Access, vol. 8, pp. 184318–184338, 2020, doi: https://doi.org/10.1109/access.2020.3025611.

[21] M. J. Hussain, H. Bai, and Y. Jiang, "Wikipedia bi-linear link (WBLM) model: A new approach for measuring semantic similarity and relatedness between linguistic concepts using Wikipedia link structure," Information Processing & Management, vol. 60, no. 2, p. 103202, Mar. 2023, doi: https://doi.org/10.1016/j.ipm.2022.103202.

[22] G. A. Miller, "WordNet: a lexical database for English," Communications of the ACM, vol. 38, no. 11, pp. 39–41, Nov. 1995, doi: https://doi.org/10.1145/219717.219748.

[23] 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," Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 2017, doi: https://doi.org/10.18653/v1/s17-2028.

[24] G. Zhu and C. A. Iglesias, "Computing Semantic Similarity of Concepts in Knowledge Graphs," IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 1, pp. 72–85, Jan. 2017, doi: https://doi.org/10.1109/tkde.2016.2610428.

[25] A. Pawar and V. Mago, "Challenging the Boundaries of Unsupervised Learning for Semantic Similarity," IEEE Access, vol. 7, pp. 16291–16308, 2019, doi: https://doi.org/10.1109/access.2019.2891692.

[26] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems 26 (2013b).

[27] Rush, Alexander M., Sumit Chopra, and Jason Weston. "A neural attention model for abstractive sentence summarization." arXiv preprint arXiv:1509.00685 (2015). https://doi.org/10.48550/arXiv.1509.00685

[28] B. Altınel and M. C. Ganiz, "Semantic text classification: A survey of past and recent advances," Information Processing & Management, vol. 54, no. 6, pp. 1129–1153, Nov. 2018, doi: https://doi.org/10.1016/j.ipm.2018.08.001.

[29] M. A. Sultan, S. Bethard, and T. Sumner, "DLS$@$CU: Sentence Similarity from Word Alignment and Semantic Vector Composition," Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015, doi: https://doi.org/10.18653/v1/s15-2027.

[30] Gabrilovich, Evgeniy, and Shaul Markovitch. "Computing semantic relatedness using Wikipedia-based explicit semantic analysis." IJcAI. Vol. 7. 2007.

[31] R. L. Cilibrasi and P. M. B. Vitanyi, "The Google Similarity Distance," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 3, pp. 370–383, Mar. 2007, doi: https://doi.org/10.1109/TKDE.2007.48.

[32] Shawe-Taylor, John, and Nello Cristianini. Kernel methods for pattern analysis. Cambridge university press, 2004. https://doi.org/10.1017/CBO9780511809682

[33] K.Knight, and K. L.Steve, "Building a large-scale knowledge base for machine translation." AAAI. Vol. 94. 1994.

[34] S. L.Reed, and B. L.Douglas "Mapping ontologies into Cyc." AAAI 2002 Conference workshop on ontologies for the semantic Web. 2002.

[35] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," Artificial Intelligence, vol. 193, pp. 217–250, Dec. 2012, doi: https://doi.org/10.1016/j.artint.2012.07.001.

[36] B. L. Humphreys, "The 1994 Unified Medical Language System Knowledge Sources," Health Libraries Review, vol. 11, no. 3, pp. 200–203, Sep. 1994, doi: https://doi.org/10.1046/j.1365-2532.1994.11301972.x.

[37] S. O. Nelson, W. Douglas Johnston, and B. L. Humphreys, "Relationships in Medical Subject Headings (MeSH)," Information science and knowledge management, pp. 171–184, Jan. 2001, doi: https://doi.org/10.1007/978-94-015-9696-1_11.

[38] K. S. Tai, , R. Socher, , and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks"arXiv preprint arXiv:1503.00075.2015 . https://doi.org/10.48550/arXiv.1503.00075

[39] N. H. Tien, N. M. Le, Y. Tomohiro, and I. Tatsuya, "Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity," Information Processing & Management, vol. 56, no. 6, p. 102090, Nov. 2019, doi: https://doi.org/10.1016/j.ipm.2019.102090.

[40] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. https://doi.org/10.1162/neco.1997.9.8.1735

[41] P. Zhou et al., "Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification," ACLWeb, Aug. 01, 2016. https://www.aclweb.org/anthology/P16-2034 (accessed May 30, 2020).

[42] J. Gu et al., "Recent advances in convolutional neural networks," Pattern Recognition, vol. 77, pp. 354–377, May 2018, doi: https://doi.org/10.1016/j.patcog.2017.10.013.

[43] J. Devlin, M. W. Chang, , K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding,", arXiv preprint arXiv:1810.04805 ,(2018). https://doi.org/10.48550/arXiv.1810.04805

[44] J. Camacho-Collados, M. T. Pilehvar, and R. Navigli, "Nasari : Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities," Artificial Intelligence, vol. 240, pp. 36–64, Nov. 2016, doi: https://doi.org/10.1016/j.artint.2016.07.005.

[45] Y. Lee, H. Ke, T. Yen, H. Huang, and H. Chen, "Combining and learning word embedding with WordNet for semantic relatedness and similarity measurement," Journal of the Association for Information Science and Technology, vol. 71, no. 6, pp. 657–670, Jul. 2019, doi: https://doi.org/10.1002/asi.24289.

[46] K. Orkphol and W. Yang, "Word Sense Disambiguation Using Cosine Similarity Collaborates with Word2vec and WordNet," Future Internet, vol. 11, no. 5, p. 114, May 2019, doi: https://doi.org/10.3390/fi11050114.

[47] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," in Proceedings of the 14th International Joint Conference on Artificial Intelligence, arXiv preprint cmp-lg/9511007, 1995.

[48] Y. Bai, L. Zhao, Z. Wang, J. Chen, and P. Lian, "Entity Thematic Similarity Measurement for Personal Explainable Searching Services in the Edge Environment," IEEE Access, vol. 8, pp. 146220–146232, 2020, doi: https://doi.org/10.1109/access.2020.3014185.

[49] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa, "A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches," Association for Computational Linguistics, 2009. Accessed: May 11, 2023. [Online]. Available: https://aclanthology.org/N09-1003.pdf https://doi.org/10.3115/1620754.1620758

[50] R. Rehurek, and P. Sojka, "Software framework for topic modelling with large corpora". In In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks., 2010.

[51] S. Bird, E. Klein, and E. Loper, Natural language processing with Python. Beijing Etc.: O'reilly, 2009.

[52] P. Virtanen et al., "SciPy 1.0: fundamental algorithms for scientific computing in Python," Nature Methods, vol. 17, no. 3, pp. 261–272, Feb. 2020, doi: https://doi.org/10.1038/s41592-019-0686-2.

[53] G. A. Miller and W. G. Charles, "Contextual correlates of semantic similarity," Language and Cognitive Processes, vol. 6, no. 1, pp. 1–28, Jan. 1991, doi: https://doi.org/10.1080/01690969108406936.

[54] H. Rubenstein and J. B. Goodenough, "Contextual correlates of synonymy," Communications of the ACM, vol. 8, no. 10, pp.627–633, Oct. 1965, doi: https://doi.org/10.1145/365628.365657.

BIOGRAPHY OF AUTHORS

Aissa FELLAH he is currently Associate Professor at Computer Science Department in Tahar Moulay University of Saida. received his M.Sc. degree and Ph.D In computer science from Sidi Bel Abbes University, Algeria. His academic interests include semantic Web services and ontology matching.

Ahmed ZAHAF received his Engineer degree in computer science from Oran University, Algeria, and M.Sc. And Ph.D In computer science from Sidi Bel Abbes University, Algeria. Currently, he is Associate Professor at Computer Science Department in Tahar Moulay University of Saida, Algeria. His research interests include semantic web, Linked data, ontology engineering, knowledge management and information systems.

Atilla Elçi is the full professor in the Software Engineering Department at Hasan Kalyoncu University (Aug 2020-). He is the retired chairman of Electrical and Electronics Engineering at Aksaray University, Turkey (August 2012 - September 2017). He served as full professor of computer engineering at several universities in Turkey. He has organized or served in the committees of numerous international conferences. He has published over a hundred journal and conference papers; He is an associate editor of Expert Systems: The Journal of Knowledge Engineering and editorial board member of several other journals. He obtained B.Sc. in Computer/Control Engineering at METU, Ankara, Turkey (1970), M.Sc. & Ph.D. in Computer Sciences at Purdue University, USA (1973, 1975).