A Novel Neurofuzzy Approach for Semantic Similarity Measurement

Jorge Martinez-Gil[1], Riad Mokadem[2], Josef Küng[3], Abdelkader Hameurlain[2]

[1]Software Competence Center Hagenberg GmbH
Softwarepark 21, 4232 Hagenberg, Austria
[2]IRIT Toulouse
Route de Narbonne 118, 31062 Toulouse Cedex, France
[3]Johannes Kepler University Linz
Altenbergerstraße 69, 4040 Linz, Austria
e-mail: jorge.martinez-gil@scch.at

Abstract. The problem of identifying the degree of semantic similarity between two textual statements automatically has grown in importance in recent times. Its impact on various computer-related domains and recent breakthroughs in neural computation has increased the opportunities for better solutions to be developed. This research takes the research efforts a step further by designing and developing a novel neurofuzzy approach for semantic textual similarity that uses neural networks and fuzzy logics. The fundamental notion is to combine the remarkable capabilities of the current neural models for working with text with the possibilities that fuzzy logic provides for aggregating numerical information in a tailored manner. The results of our experiments suggest that this approach is capable of accurately determining semantic textual similarity.

Keywords: Data Integration, Neurofuzzy, Semantic Similarity, Deep Learning Applications

1 Introduction

Data mining and knowledge discovery techniques have long been trying to boost the decision-making capabilities of human experts in a wide variety of academic disciplines and application scenarios. In addition, these techniques have greatly facilitated the development of a new generation of computer systems. These systems are designed to solve complex problems using expert-generated knowledge rather than executing standard source code. However, this notion has been evolving towards more effective and efficient models and offers more flexibility in supporting the human judgment in decision-making. In this work, we look at the field of semantic textual similarity. That is the possibility of boosting the capability of a human expert in deciding whether two pieces of textual information could be considered similar. In this way, automatic data integration techniques can be notably improved.

Over the last two decades, different approaches have been put forward for computing semantic similarity using a variety of methods and techniques [11]. In this way, there are already many solutions for automatically calculating the semantic similarity between words, sentences, and even documents. Currently, the solutions that can obtain the best results are those based on neural networks such as USE [4], BERT [9] or ELMo [25]. However, there is still much room for improvement since the development of these models is still in its infancy. Many of the functions they implement are trivial, and it is to be expected that as more sophisticated approaches are investigated, the results to be achieved could be better.

For these reasons, we have focused on a novel approach that is slowly making its way into the literature: neurofuzzy systems. Systems of this kind are built by a clever combination of artificial neural networks and fuzzy logic. These systems attract much attention because they can bring together the significant advantages of both worlds [27]. However, its application in the domain of semantic similarity remains unexplored.

We want to go a step further to design a new neurofuzzy approach that might be able to determine automatically and with high accuracy the degree of semantic similarity between pieces of textual information. To do that, we propose to follow a concurrent fuzzy inference neural network (FINN) approach being able to couple the state-of-the-art models from the neural side together with the state-of-the-art from the fuzzy side. This approach is expected to achieve highly accurate results as it brings together the computational power of neural networks with the capability of information fusion from fuzzy logics. Thus, the contributions of this research work can be summarized as follows:

- We propose for the first time a neurofuzzy schema for semantic similarity computation that combines the ability of neural networks to transform pieces of textual information into vector information suitable for processing by automatic methods with the advantages of personalized aggregation and decoding offered by fuzzy logics.
- We have subjected our proposal to an empirical study in which we compare it with state-of-the-art solutions in this field. The results obtained seem to indicate that our proposal yields promising results.

The rest of this paper is structured as follows: in section 2, we present the state-of-the-art concerning the automatic computation of semantic similarity when working with textual information and recent solutions based on neurofuzzy systems to solve practical application problems. In section 3, we provide the technical explanation on which our neurofuzzy approach is based. In section 4, we undertake an empirical study that compares our approach with those that make up the state-of-the-art. Finally, we highlight the lessons that can be drawn from the present work and point out possible future lines of research.

2 State-of-the-art

The semantic similarity field attracts much attention because it represents one of the fundamental challenges that can advance several fields and academic disciplines [17]. The possibility that a computer can automatically determine the degree of similarity between different pieces of textual information regardless of their lexicography can be very relevant. This means that areas such as data integration, question answering, or query expansion could greatly benefit from any progress in this area.

To date, numerous solutions have been developed in this regard. These solutions range from traditional techniques using manually compiled synonym dictionaries such as Wordnet [15], to methods using the web as a large corpus such as Normalized Google Distance [5] through the classical taxonomy-based techniques [26] or the ones based on corpus statistics [13].

Besides, more and more solutions have been developed that are valid in a wide range of domains of different nature. Some of these solutions are based on the aggregation of atomic methods to benefit from many years of research and development in semantic similarity measures [21]. Moreover, there have been breakthroughs that have completely revolutionized the field of semantic similarity. One of the most promising approaches has been word embeddings [23]. Where solutions of a neural nature have been able to reduce pieces of text to feature vectors of a numerical nature so that they are much more suitable for automatic processing by computers [14]. These approximations are so robust that they can determine the degree of similarity of cross-lingual expressions [10].

Neurofuzzy systems have begun to be used in many application domains due to the versatility they offer. Neurofuzzy systems have the great advantage of combining the human-like reasoning of fuzzy systems through a linguistic model based on IF-THEN rules with the tremendous computational power to discover patterns of neural networks. Neurofuzzy systems are considered universal approximators because of their ability to approximate any mathematical function, which allows them to be highly qualified for problems related to automatic learning.

Concerning these neurofuzzy systems' neural side, some neural approaches have been recurrently used to work with text—for example, automatic translation or text auto-completion. The problem is that these models are usually not very good at capturing long-term dependencies. For this reason, transformer architectures have recently emerged [9]. This kind of architecture uses a particular type of attention known as self-attention [8].

However, all these architectures of neural nature that have been presented have only used simplistic ways of aggregation and decoding of the last neural layer to date. The operations that can be found recurrently in the literature are cosine similarity, manhattan distance, euclidean distance, or inner product. This is where our contribution comes into play since we propose a fuzzy logic-based solution that can model a much more sophisticated interaction between the numerical feature vectors generated by the neural part.

3 A novel neurofuzzy approach for semantic similarity

Fuzzy logic can offer computational methods that aim to formalize reasoning methods that are considered approximate. We are here considering Mamdani fuzzy inference [19] since it is considered an optimal method for developing control systems governed by a set of rules very close to natural language. In Mamdani fuzzy inference systems, the output of each rule is always a fuzzy set. Since systems of this kind have a rule base that is very intuitive and close to natural language, they are often used in expert systems to model the knowledge of human experts.

The reason to use fuzzy logics is that rules can also be derived analytically when it is impossible to count on the expert's help, as is the case in our approach. In our specific scenario, we use aggregation controllers. These controllers are usually divided into several components including a database of terms such as $\mu_{\widetilde{S}}(x)$ that states the membership of x in $\widetilde{S} = \left\{ \int \frac{\mu_{\widetilde{S}}(x)}{x} \right\}$ what is usually defined as $\mu_{\widetilde{S}}(x) \in [0, 1]$, and a non-empty set of rules. In this way, the terms associated with the database can be used to characterize the rules.

Moreover, the input values need to be encoded according to the terms from the controller, so that $\widetilde{I} = \mu_1 Q(x_1) + \mu_2 Q(x_2) + ... + \mu_n Q(x_n)$, whereby μ_i is the term associated with the transformation of x_i into the set $Q(x_i)$.

Last but not least, we need to define the terms on the basis of membership functions so that: $\widetilde{T} = \left\{ \left(x, \mu_{\widetilde{T}}(x)\right) \mid x \in U \right\}$. The great advantage of this approach is that a wide range of membership functions can be defined by just using a limited number of points which represents an advantage for us when coding possible solutions in the form of individuals from an evolving population.

Working with Mamdani fuzzy systems [19] also means that the result of the inference will be a set such as $\widetilde{O} = \left\{ \int \frac{\mu_{\widetilde{O}}(v)}{v} \right\}$. Therefore, the output might be a real value representing the result of aggregating the input values. One of the traditional advantages of Mandami's models concerning other approaches, e.g., Tagaki-Sugeno's [28], is that they facilitate interpretability. This is because the Mamdani inference is well suited to human input while the Tagaki-Sugeno inference is well suited to analysis [7].

On the other hand, the neural side will use transformers that are suitable models for translations between abstract representations. The transformer models consist of an encoder-decoder architecture. It is necessary to feed the encoder with the input textual information. From there, the encoder learns to represent the input information and sends this representation to the decoder. The decoder receives the representation and generates the output information to be presented to the user. The way of working is like this: for each attention item, the transformer model learns three weight matrices; the query matrix W_Q, the key matrix W_K, and the value matrix W_V. For each token i, the input *embedding* x_i is multiplied with each of the three matrices to produce a query array $q_i = x_i W_Q$, a key array $k_i = x_i W_K$, and a value array $v_i = x_i W_V$. Attention weights are calculated using the query and key arrays in the following way: the attention weight a_{ij} from token i to token j is the product between q_i and k_j. The attention weights are divided by the square root of the key arrays' dimension, $\sqrt{d_k}$, which stabilizes gradients during training, and it is processed by a function $softmax$, which normalizes the weights to sum to 1.

Figure 1 shows the mode of operation of a concurrent FINN architecture. The input data is first processed by the neural network (i.e., transformer), which is very good with the vectorization of the pieces of textual information into numerical feature vectors. These feature vectors will correspond to the membership functions of the fuzzy module at a later stage. The coupling of the modules of neural nature with that of fuzzy logic nature should obtain good results once a training phase has correctly calibrated all their parameters. This architecture is more complex and more powerful than the more advanced neural models because it adds the last layer that can process the results in a much more sophisticated way.

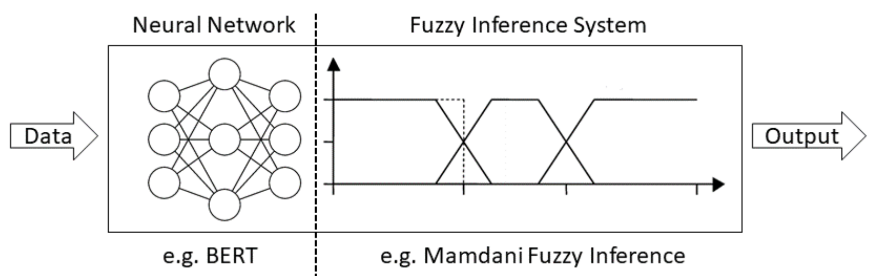

Fig. 1: Architecture of our neurofuzzy approach. The neural and fuzzy models are coupled together and then trained in order to be properly calibrated to solve the scenario that we wish

The appropriate combination of the two approaches gives rise to a concurrent FINN, which we intend to obtain good results with when trying to automatically determine the degree of semantic similarity between two textual expressions that are analogous but have been represented using different lexicographies. Also, another great advantage of these systems is that they can be trained separately or together. This allows them to benefit from great flexibility and versatility. For example, not everyone needs to develop a neural solution from scratch because the existing ones are of high quality and have been trained on hard-to-access corpora.

Finally, it is worth mentioning that our learning process is guided by an evolutionary strategy that tries to find the best parameters in the neurofuzzy model, although trying to avoid an over-fitting situation. We have opted for the classical solution for an elitist evolution model with a low mutation rate, which allows automatically exploring of the solution space, although it also allows a low rate of random jumps in search of better solutions [1]. The following pseudocode shows the rationale behind this approach.

Algorithm 1 Pseudo-code for the evolutionary strategy to obtain optimal FINN model

```
procedure CALCULATION OF THE BEST POSSIBLE FINN MODEL
    RandomIndividuals (population)
    calculateFitness (population)
    while (NOT stop condition) do
        for (each individual)
            parents ← selectionOfIndividuals ()
            offspring ← binCrossOver (parents)
            offspring ← randomMutation (offspring)
            calculateFitness (offspring)
            population ← updatePopulation (offspring)
        endfor
    endwhile
    return Model(population)
```

The evolutionary strategy allows us to optimally calibrate the following parameters: the transformer model to be used, how the operations in the last layer of the neural network will be computed, the fuzzy sets and membership functions, the IF-THEN rules that best fit the input data, as well as the defuzzification method.

4 Experimental Study

This section describes our strategy's experimental setup, including the benchmark dataset that we have used and the evaluation criteria that we are following, and the configuration of the considered methods. After that, we perform an exhaustive analysis of the different approaches considered and the empirical results we have achieved. Finally, we offer a discussion of the results we have obtained.

4.1 Datasets and Evaluation criteria

We have used the most widely used general-purpose benchmark dataset in this field to carry out our experiments. Our approach's behavior concerning this dataset will give us an idea of how our approach works. This benchmark dataset is the so-called MC30, or Miller & Charles dataset [24] that consists of 30-word pairs that everyone might use.

Evaluating the techniques to discover semantic similarity using correlation coefficients can be done in two different ways. First, it is possible to use Pearson's correlation coefficient, which can measure the degree of correlation between the background truth and the machine results, whereby a and b are respectively the source and target pieces whose degree of semantic similarity is to be compared.

$$\sigma = \frac{n \sum a_i b_i - \sum a_i \sum b_i}{\sqrt{n \sum a_i^2 - (\sum a_i)^2} \sqrt{n \sum b_i^2 - (\sum b_i)^2}} \tag{1}$$

The second way to proceed is the so-called, Spearman Rank Correlation, whereby the aim is to measure the relative order of the results provided by the technique.

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

Being $d_i = rg(X_i) - rg(Y_i)$ the difference between the two ranks of each array, and n is the size of both arrays.

The significant difference between the two correlation methods is that while Pearson is much better at determining the total order of the dataset results, Spearman is more suitable for determining a partial order.

4.2 Configuration

As we have already mentioned, the training phase is performed by an evolutionary learning strategy. Therefore, we have had to perform a grid search to determine which are the best parameters for that strategy. Since the search space is really huge, we had to narrow down the search intervals. In this way, the identified parameters of our evolutionary strategy are the following:

- Representation of genes (binary, real): **real**
- Population size [10, 100]: **42**
- Crossover probability [0.3, 0.95]: **0.51**
- Mutation probability [0.01, 0.3]: **0.09**
- Stop condition: Iterate over (1,000 - 100,000): **100,000** generations

In addition, evolutionary learning techniques are non-deterministic in that they rely on randomness components to generate initial populations and search for model improvements through small mutation rates (9% in our case). Therefore, the results reported are always the average of several independent runs, as we will explain later.

It is necessary to note that the hardware used has been an Intel Core i7-8700 with CPU 3.20 GHz and 32 GB of RAM over Windows 10 Pro. Furthermore, we rely on the implementation of Cingolani's fuzzy engine [6] as well as the implementations of USE [4], BERT [9], or ELMo [25] that their authors have published initially. In that sense, training using different text corpora may indeed yield different results. However, such a study is outside the scope of this work and would be interesting future work. Finally, it is necessary to remark that the average and maximum training times are reported in the following section.

4.3 Results

In this section, we show the empirical results that we have obtained. In Table 1, it is possible to see a summary of the results we have obtained when solving the MC30 benchmark dataset using the Pearson correlation coefficient. This means that we are looking for the capability of the different solutions to establish a

Approach	Score	p-value
Google distance [5]	0.470	$8.8 \cdot 10^{-3}$
Huang et al. [12]	0.659	$7.5 \cdot 10^{-5}$
Jiang & Conrath [13]	0.669	$5.3 \cdot 10^{-5}$
Resnik [26]	0.780	$1.9 \cdot 10^{-7}$
Leacock & Chodorow [16]	0.807	$4.0 \cdot 10^{-8}$
Lin [18]	0.810	$3.0 \cdot 10^{-8}$
Faruqui & Dyer [10]	0.817	$2.0 \cdot 10^{-8}$
Mikolov et al. [23]	0.820	$2.2 \cdot 10^{-8}$
CoTO [20]	0.850	$1.0 \cdot 10^{-8}$
FLC [22]	0.855	$1.0 \cdot 10^{-8}$
Neurofuzzy (median)	0.861	$4.9 \cdot 10^{-9}$
Neurofuzzy (maximum)	0.867	$1.0 \cdot 10^{-9}$

Table 1: Results over the MC30 dataset using Pearson Correlation

total order. Please note that the results reported for our approach are based on ten independent executions due to the non-deterministic nature of the learning strategy. So we report the median value and the maximum value achieved.

Table 2 shows the results obtained for Spearman's correlation coefficient. This means that we evaluate the capability of the existing approaches when determining a partial order between the cases of the MC30 dataset. Once again, we report the median value and the maximum value achieved.

Approach	Score	p-value
Jiang & Conrath [13]	0.588	$8.8 \cdot 10^{-3}$
Lin [18]	0.619	$1.6 \cdot 10^{-4}$
Aouicha et al. [2]	0.640	$8.0 \cdot 10^{-5}$
Resnik [26]	0.757	$5.3 \cdot 10^{-7}$
Mikolov et al. [23]	0.770	$2.6 \cdot 10^{-7}$
Leacock & Chodorow [16]	0.789	$8.1 \cdot 10^{-8}$
Bojanowski et al. [3]	0.846	$1.1 \cdot 10^{-9}$
Neurofuzzy (median)	0.851	$1.0 \cdot 10^{-9}$
Neurofuzzy (maximum)	0.868	$4.6 \cdot 10^{-9}$
FLC [22]	0.891	$1.0 \cdot 10^{-10}$

Table 2: Results over the MC30 dataset using Spearman Rank Correlation

Finally, we offer a study of the time it takes to train our neurofuzzy systems. Figure 2 shows the training that has been performed to obtain the Pearson correlation coefficient. As we are working with non-deterministic methods, the results presented are equivalent to an average of ten independent runs of which we plot the minimum (red), the median (blue) and the maximum (black).

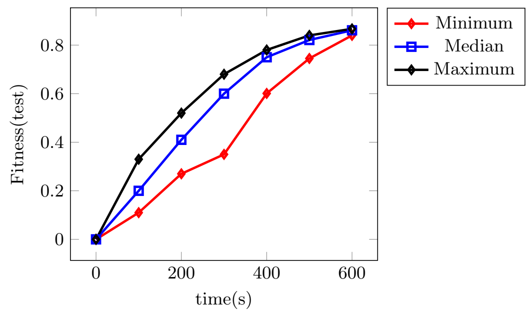

Fig. 2: Fitness evolution for the Pearson correlation. The red, blue, and black plots represent the worst, median and best cases respectively

While Figure 3 shows the time taken to properly set up our approach to solve the Spearman Rank Correlation. As in the previous case, the results we can see in the plot are the result of ten independent runs of which we plot the minimum, the median and the maximum values that we have achieved.

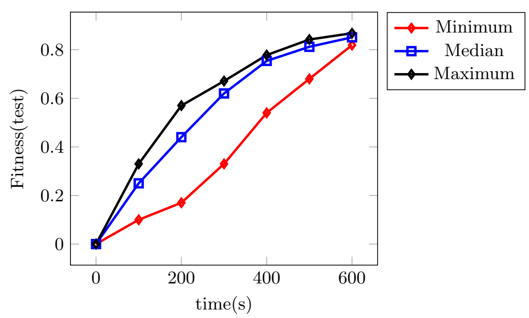

Fig. 3: Fitness evolution for the Spearman Rank correlation. The red, blue, and black plots represent the worst, median and best cases respectively

As it can be seen, our approach presents several significant advantages concerning the works that make up the state-of-the-art. As for techniques based on neurofuzzy hybridization, no work has yet been done in this area to the best of our knowledge. However, our previous experience designing solutions based on fuzzy logic led us to think that combining the human-like reasoning of fuzzy logics with neural networks' learning capability would yield quite good results. Training a neurofuzzy system indeed involves a significant consumption of resources in the form of time, but it is also true that once trained, the results are pretty good. Moreover, the model can be reused. Even mature transfer learning techniques can facilitate its application in analog-nature problems in different scenarios.

5 Conclusions and Future Work

We have presented a novel approach for the automatic computation of the degree of similarity between textual information pieces. Our approach is novel because it is the first time that a neurofuzzy system is proposed to deal with the problem. We think that a neurofuzzy system is appropriate in this situation since it can combine the high capabilities of neural nature solutions to extract and convert features associated with text expressed in natural language with the ability of fuzzy logics to aggregate and decode in a personalized way information of numerical nature.

The results we have obtained show that our approach can achieve results in line with the state-of-the-art, even without being specifically trained. These promising results rely on solutions of neural nature whose accuracy is highly contrasted together with fuzzy logic system, which has a great capacity to aggregate intermediate results and decode them into the values for which it has been trained.

Besides, the two parts that make up our system can be trained separately. For example, we can use a highly constrained accuracy model such as BERT combined with a classical Mamdani inference model, which usually gives outstanding results. In this way, our system is built based on building blocks that give it flexibility and versatility not known so far in semantic similarity measurement.

As future work, we plan to explore other approaches to assess similarity automatically. We have worked here with monolingual semantic similarity, i.e., all pieces of textual information were expressed in English. However, a pending issue is to study the problem from a cross-lingual perspective. The other pending issue would be how to improve the interpretability of the resulting system. Due to the black-box model that is implemented in the neural part, a solution must be found so that people can understand this model from the beginning to the end.

Acknowledgements

This work has been supported by the Austrian Ministry for Transport, Innovation and Technology, the Federal Ministry of Science, Research and Economy, and the State of Upper Austria in the frame of the COMET center SCCH. By the project FR06/2020 by International Cooperation Mobility (ICM) of the Austrian Agency for International Cooperation in Education and Research (OeAD-GmbH). We would also thank 'the French Ministry of Foreign and European Affairs' and 'The French Ministry of Higher Education and Research' which support the Amadeus program 2020 (French-Austrian Hubert Curien Partnership – PHC) Project Number 44086TD.

References

1. Plamen P. Angelov and Richard A. Buswell. Automatic generation of fuzzy rule-based models from data by genetic algorithms. *Inf. Sci.*, 150(1-2):17–31, 2003.
2. Mohamed Ben Aouicha, Mohamed Ali Hadj Taieb, and Abdelmajid Ben Hamadou. LWCR: multi-layered wikipedia representation for computing word relatedness. *Neurocomputing*, 216:816–843, 2016.
3. Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguistics*, 5:135–146, 2017.
4. Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. Universal sentence encoder for english. In Eduardo Blanco and Wei Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 169–174. Association for Computational Linguistics, 2018.
5. Rudi Cilibrasi and Paul M. B. Vitányi. The google similarity distance. *IEEE Trans. Knowl. Data Eng.*, 19(3):370–383, 2007.
6. Pablo Cingolani and Jesús Alcalá-Fdez. jfuzzylogic: a java library to design fuzzy logic controllers according to the standard for fuzzy control programming. *Int. J. Comput. Intell. Syst.*, 6(sup1):61–75, 2013.
7. Oscar Cordón. A historical review of evolutionary learning methods for mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *Int. J. Approx. Reason.*, 52(6):894–913, 2011.
8. Biyun Dai, Jinlong Li, and Ruoyi Xu. Multiple positional self-attention network for text classification. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7610–7617. AAAI Press, 2020.
9. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
10. Manaal Faruqui and Chris Dyer. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 462–471, 2014.
11. Sébastien Harispe, Sylvie Ranwez, Stefan Janaqi, and Jacky Montmain. *Semantic Similarity from Natural Language and Ontology Analysis.* Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2015.
12. Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. Improving word representations via global context and multiple word prototypes. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 873–882, 2012.
13. Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th Research on Computational Linguistics International Conference, ROCLING 1997, Taipei, Taiwan, August 1997*, pages 19–33, 1997.

14. Juan J. Lastra-Díaz, Ana García-Serrano, Montserrat Batet, Miriam Fernández, and Fernando Chirigati. HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset. *Inf. Syst.*, 66:97–118, 2017.
15. Juan J. Lastra-Díaz, Josu Goikoetxea, Mohamed Ali Hadj Taieb, Ana García-Serrano, Mohamed Ben Aouicha, and Eneko Agirre. A reproducible survey on word embeddings and ontology-based methods for word similarity: Linear combinations outperform the state of the art. *Eng. Appl. Artif. Intell.*, 85:645–665, 2019.
16. Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283, 1998.
17. Yuhua Li, Zuhair Bandar, and David McLean. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Trans. Knowl. Data Eng.*, 15(4):871–882, 2003.
18. Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27, 1998*, pages 296–304, 1998.
19. E. H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Hum.-Comput. Stud.*, 51(2):135–147, 1999.
20. Jorge Martinez-Gil. Coto: A novel approach for fuzzy aggregation of semantic similarity measures. *Cognitive Systems Research*, 40:8–17, 2016.
21. Jorge Martinez-Gil. Semantic similarity aggregators for very short textual expressions: a case study on landmarks and points of interest. *J. Intell. Inf. Syst.*, 53(2):361–380, 2019.
22. Jorge Martinez-Gil and José M. Chaves-González. Automatic design of semantic similarity controllers based on fuzzy logics. *Expert Syst. Appl.*, 131:45–59, 2019.
23. Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119, 2013.
24. G. Miller and W. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.
25. Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237, 2018.
26. Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.*, 11:95–130, 1999.
27. Leszek Rutkowski and Krzysztof Cpalka. Flexible neuro-fuzzy systems. *IEEE Trans. Neural Networks*, 14(3):554–574, 2003.
28. Tomohiro Takagi and Michio Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man Cybern.*, 15(1):116–132, 1985.