Semantic Similarity Controllers: On the Trade-off between Accuracy and Interpretability

Jorge Martinez-Gil

Software Competence Center Hagenberg GmbH
Softwarepark 21, 4232 Hagenberg, Austria
jorge.martinez-gil@scch.at

Jose Manuel Chaves-Gonzalez

University of Extremadura - Department of Computer Systems Engineering
Avda. de la Universidad s/n Caceres, Spain
jm@unex.es

Abstract

In recent times, we have seen an explosion in the number of new solutions to address the problem of semantic similarity. In this context, solutions of a neuronal nature seem to obtain the best results. However, there are some problems related to their low interpretability as well as the large number of resources needed for their training. In this work, we focus on the data-driven approach for the design of semantic similarity controllers. The goal is to offer the human operator a set of solutions in the form of a Pareto front that allows choosing the configuration that best suits a specific use case. To do that, we have explored the use of multi-objective evolutionary algorithms that can help find break-even points for the problem of accuracy versus interpretability.

Keywords: Knowledge Engineering, Fuzzy Logic Controllers, Similarity Learning, Semantic Similarity Measurement

1. Introduction

In recent times, we have witnessed an absolute prevalence of computational solutions based on the architectures of a neural nature when it comes to automatically solve problems related to semantic similarity. The truth is that solutions of this kind are, without a doubt, the ones that manage,

time and again, to overpass the state-of-the-art in a wide range of computational tasks and problems of both academic and business nature. However, by betting on this kind of solution, the research community runs the risk of making chronicle some of the traditional problems associated with the use of artificial neural networks (ANNs). Among these problems, three main ones stand out: the lack of interpretability, the large amount of data required to carry out training correctly, and the difficulty of transfer learning.

The problem of lack of interpretability is because it is humanly impossible to understand a model that is based on many hundreds or thousands of interconnected nodes. In this way, a human operator can define the outputs that correspond to certain inputs, and the ANN will be in charge of configuring a fairly accurate mapping function, but the human operator will not be able to obtain any clues about what happens within that model. For that reason, the models of this kind are known as black-boxes due to the characteristic that leads them to hide their operation insights from users.

Another problem presented by neuronal architectures is the large number of solved cases that they need to start giving good results. Fortunately, in some domains, there is a lot of data perfectly labeled and prepared to serve as training to ANNs, but in many other specific domains, the amount of existing data is not so voluminous and this kind of solution encounters problems to complete a training phase that might shed certain guarantees of success.

Last but not least, such solutions have to face the problem of transfer learning, i.e. how to transfer the knowledge learned through a training process to solve another problem of analogous nature but not completely similar. The complexity of this problem is closely related to the first point and lies in the fact that if we are not able to understand the model, we will probably not be able to apply it in a similar situation in the future.

With the idea of solving these problems, we have raised the idea of using semantic similarity controllers in the past [54]. These controllers are software artifacts that can be automatically designed to avoid the problems we have mentioned above. The idea of the semantic similarity controllers is

to be able to aggregate, both automatically and strategically, already existing methods for assessing semantic similarity. These methods must meet several characteristics, among which they must yield good results on an individual basis and must be easily interpretable by a human operator.

By being able to design these controllers automatically following some of the existing information fusion paradigms, we will be able to transfer desirable properties to these controllers. For example, in our previous work, we demonstrated how it was possible to automatically design semantic similarity controllers based on fuzzy logics that allow: the obtaining of results quite close to the state-of-the-art, without requiring large amounts of labeled data for training and with the possibility of transferring the captured knowledge simply. All this was due to the fact that fuzzy logics allows the formulation of aggregation problems through concepts and rules that are similar to natural language. In addition, we imposed conditions to facilitate, even more, the interpretability. For example, we made strictly obligatory that a reduced set of concepts and rules might be used, and that these rules must be simple, with at most two antecedents.

In such work, we use an evolutionary learning strategy to automate the design phase. We did not influence the design of this strategy because it was outside the scope of that work. However, we now wonder what might be the most promising strategy for the automatic and efficient design of these software artifacts with a special focus on the accuracy versus interpretability trade-off [7], i.e. give the human operator the possibility to decide what kind of configuration (e.g. more interpretability and less accuracy or vice versa) best fits a given situation. Therefore, the major contributions to the state-of-the-art of this work are as follows.

- We have conducted a massive analysis of a large number of state-of-the-art multiobjetive evolutionary learning strategies in order to determine which one is best for building semantic similarity controllers based on fuzzy logic paying special attention to model an appropriate interaction between the predictive accuracy and its associated interpretability.
- We have achieved results that represent a new state-of-the-art in terms of the calculation of

semantic similarity using highly interpretable strategies, without the need for large volumes of data for training and with the possibility of simple and efficient transfer learning.

The rest of this work is structured as follows: In section 2, we present the state-of-the-art concerning the design of interpretable solutions for the calculation of semantic similarity. In section 3, we develop our contribution to the state-of-the-art with special focus on explaining the need of building solutions that can correctly handle a trade-off between accuracy and interpretability. In section 4, we report the results obtained after an extensive empirical evaluation of the evolutionary strategies mentioned above. And finally, we extract the lessons learned from this work and set the guidelines for future work.

2. State-of-the-art

The problem of automatically assessing the semantic similarity between words, sentences, text paragraphs, or even documents is widely assumed to be a research challenge that tries to address one of the aspects of artificial intelligence that will allow computers to perform routine and tedious tasks [16, 50, 53]. This field has attracted a lot of attention in recent times due to its relevance to both industry and academia. The reason for this is that having computer systems that can correctly assess the likeness of two pieces of text might bring a window of opportunities to achieve an impact from the most basic research to the most advanced business models [66].

For this reason, there are numerous solutions to the problem of semantic similarity measures [44]. Traditionally, most techniques have been based on the development or refinement of natural language techniques using some kind of manually compiled resources such as dictionaries, taxonomies, etc [31]. However, in recent times both academia and industry are turning more to neural network-based solutions inspired in the seminal work of Mikolov et al.[56]. For example, BERT [24] and ELMo [59]. These neural network-based techniques require a lot of training time, but once trained, are very accurate. The problem is their associated lack of interpretability [55], i.e. the limited capacity a

person might have to understand why the methods work so well as well as the number of resources required for training and their adaptation to working with new cases.

However, without detracting from the merit of neural solutions, many professionals from the most diverse application domains (legal, medicine, pharma, etc.) are usually not satisfied with just an answer to their question and demand much more. In fact, they demand to understand why the model has opted for such an output and not for another of the alternatives. This brings up one of the ghosts that have always been associated with neural computing, i.e. its behavior is similar to that of a black-box, since it is possible to give it an input and obtain an output, but it is humanly impossible to understand how the connections of thousands of neurons have worked to give rise to such an output, or even what is the real meaning behind the feature vectors that have been obtained after subjecting the model to the training of a neuronal nature. For example, the base configuration of BERT [24] requires a configuration consisting of 12 layers of neurons and 12 windows of attention. Therefore, there are usually issues related to the interpretability of the resulting model. This is where our research comes in. In fact, we have set out to explore one of the most popular methods in communities working on methods to automate information fusion: fuzzy logics.

While fuzzy logics is now well-established and has a strong community behind it that has been studying it for several decades, there is a lack of work on its application to the semantic similarity problem [51]. We believe that this gap should be explored since fuzzy logics allow the construction of rule-based models, in many cases, very close to natural language [4], making them suitable candidates to implement solutions where interpretability is a real requirement [19].

For the design of accurate and interpretable fuzzy rule-based systems, evolutionary multiobjective optimization methods have traditionally been used. Multiobjective genetic fuzzy systems is a term used to describe a research topic in which evolutionary algorithms are employed to find non-dominated fuzzy rule-based systems that are accurate and interpretable. Some of the classic works in this area include [64], [36], [2], and [38].

In addition to evolutionary strategies, there are other approaches that seek the intelligent generation of solution fronts. For example, an outstanding approach consists of the active learning of Pareto fronts [14]. This approach enables an analytical model of the Pareto front to be built whereby an active learning strategy reduces the computational effort in generating the necessary information. The model is learned from a set of informative training objective vectors. The training objective vectors are approximated Pareto-optimal vectors obtained by solving different scalarized problem instances. Moreover, Aggarwal [1] proposes the algorithm PLEMOA to automate the decision processes through the generation of a vast number of solutions. PLEMOA uses pair-wise comparisons to gather information to learn an ideal solution corresponding to a decision-maker's preferences for different conflicting objectives under consideration.

In the rest of this paper, we present our research around the proper trade-off between accuracy and interpretability on semantic similarity controllers using different evolutionary strategies so that they can contribute to facilitate the work of human operators, reduce the number of training resources, and facilitate the transfer learning processes. This is particularly interesting in an application domain such as semantic similarity measurement that is currently dominated by black-box solutions.

3. On the Trade-off between Accuracy and Interpretability

Although much of the current research focuses on getting word embeddings that work best for a given domain, our approach is radically different. We focus on the reuse of simple but highly interpretable techniques that already exist in the literature. Our concept consists of given a set of existing semantic similarity measures trying to calculate a scoring function by their automatic strategic aggregation.

The problem is that designing that scoring function is far from being trivial. Currently, there are many proposals to automatically assess semantic similarity [43]. Some of these proposals are based on the exploitation of taxonomies [62], variations of the concept of web distance [17], others are based

on the distributional assumption terms that appear in a similar context base [23], the calculation of synonyms [46], others are based on the co-occurrence of words in the same textual corpus [32], etc. In principle, it is very difficult to discern which approach could perform better than the others. This always depends on the use case and the context in which they are applied. For this reason, our research does not presuppose any method in advance and tries to build a model that gives each one of them a chance.

3.1. Fuzzy Logics

Fuzzy logics have been already used in many application domains in a successful way. In our specific scenario, we use semantic similarity 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 of the database, 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 $Q(x_i)$.

Finally, it is necessary to define the terms on the basis of membership functions so that: $\widetilde{T} = \left\{ \left(x, \mu_{\widetilde{T}}(x)\right) \mid x \in U \right\}$. Although it is possible to use many points to define those functions, in practice, a wide range of membership functions can be defined by just making use of a limited number of points which represents an advantage for us when coding possible solutions in the form of individuals from a population as we will see later.

We focus here on Mamdani fuzzy systems [49] what 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 variable is a real value representing the result of the process of aggregating the existing methods. The advantage of Mandami's models in relation to others that are also quite popular, e.g. Tagaki-Sugeno's [65], is that they facilitate interpretability. This is because the Mamdani inference is well suited to human input while the Tagaki-Sugeno inference

system is well suited to mathematically analysis [19].

3.2. Fuzzy aggregation strategies

Aggregation methods are very popular in various areas of computing and are often used in production environments, as they allow to blur the errors that a method makes between a set of methods that usually work well most of the time [52]. Only in the rare case whereby all the methods might make the same mistake at the same time, the aggregation techniques lose their effectiveness. Some of the most common aggregation operators are the arithmetical mean, the median, and the geometric and harmonic means. However, their aggregation strategy is usually short-sighted since it is not able to model adequate interaction between the input variables. This usually means that these strategies do not lead to optimal results. Therefore, researchers tend to look for more sophisticated operators.

3.3. On the importance of the interpretability

If we did not have as a fundamental requirement, the interpretability of the solution, training an ANN that might be capable of aggregating simple methods in terms of a training set would be the ideal solution to our problem. However, the capability to understand the model, which indeed is a really important aspect in many application domains, is not the only obstacle. For example, also finding large data sets that represent solved cases is very difficult or expensive, or the ANN model is usually difficult to export.

It is not easy to define interpretability. For example, Magdalena develops the notion of intepretability by stating that a model is interpretable in human terms if the language of the model can be translated into the language of the human interpreter [48]. While Bodenhofer and Bauer state that interpretability means the possibility to estimate the system's behavior by reading and understanding the rule base [12]. Magdalena also states that interpretability is not only a matter of the semantics of information, but it is also a matter of volume. For example, a large number of rules, rules

jointly considering many variables, or linguistic variables with large term sets, will negatively affect interpretability.

And this is just where our research contributes since when building a predictive model in the field of semantic similarity measurement, there are two important orthogonal features: accuracy and interpretability, which generally can be modeled through a trade-off relationship. In our specific case, we want to be able to offer the human operator the possibility to choose between accuracy and interpretability when configuring a semantic similarity controller. Accuracy is about to have a model being able to make correct predictions, while interpretability is about capability of the model to facilitate its human understanding. In the particular case of semantic similarity controllers, it is necessary to remark that Mamdani rules use variables in premise and consequent, and therefore, they are more interpretable than Tagaki-Sugeno's rules [65] that use functions in the consequent. Therefore, we will exclusively refer to the Mamdani model [49].

3.4. Semantic Similarity Controllers

In this work, we focus on the training of semantic similarity controllers that allows us to aggregate inputs of different nature. This approach is based on the idea that models which bring together a variety of methods are usually able to achieve better performance than the one that could be obtained from any of the methods alone [63]. A controller of this kind is characterized following a data-driven approach. There is always interdependency between terms and rules since the conditions and the consequences of the rules are associated with the aforementioned terms. The great advantage of this method is that a sufficiently trained human operator can simply observe the resulting model, understand how it works, and translate it into a form close to natural language.

Concerning the automatic derivation of the semantic similarity controller, it is necessary to remark that each of the inputs corresponds to the methods to be aggregated. It is important to emphasize that our approach is guided by a learning strategy that tries to find the best parameters, although without falling into over-fitting [6]. At present, there are numerous approaches based on a multiobjective

approach being able to do that. But as this is an eminently empirical field of study, it is difficult to discern which strategy best suits each specific scenario. For this reason, a rigorous analysis of each of the strategies is necessary in order to determine which one behaves better in our particular context.

3.5. Multiobjetive learning

Multiobjective learning is an approach involving more than one function to be simultaneously learned. This kind of learning is useful in scenarios whereby decisions need to be taken in the presence of trade-offs between more than one orthogonal objectives [28]. This is because no single solution exists simultaneously satisfying each objective. Therefore, without the external judgment from a human operator, all optimal solutions should be considered equally good [29]. We can express it more formally as follows:

$$\min\left(f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})\right) \text{s.t.} \vec{x} \in X,$$

In the field of multiobjective learning, there does not typically exist a solution for all objective functions at the same time. Therefore, the focus must be on solutions that cannot be improved in any of the objectives without worsening at least another one. More formally, a solution $\vec{x}_1 \in X$ is said to dominate another one $\vec{x}_2 \in X$, if

$$f_i(\vec{x}_1) \leq f_i(\vec{x}_2)$$

$$\text{for all } i \in \{1, 2, \ldots, k\}$$

$$f_j(\vec{x}_1) < f_j(\vec{x}_2)$$

$$\text{for at least } j \in \{1, 2, \ldots, k\}$$

A solution $\vec{x} \in X$ is called Pareto optimal if there does not exist any solution that might dominate it. An item of the search space x is said to dominate another item y if x is not worse than y concerning all objectives, and is strictly better than y for at least one.

The set of items of the search space that are not Pareto-dominated by any other element is called the *Pareto front* and it represents the best possible compromises to the orthogonal objectives. And it is the way we model an appropriate trade-off between accuracy and interpretability.

Unlike classical methods, which usually try to find one optimal solution, multiobjective strategies are the only approaches that can directly search for the whole Pareto front, allowing human operators to choose one of the solutions depending on the subjective information that they handle as well as defining what levels of accuracy and interpretability are tolerable [30].

Please note that according to [37], there two ways to improve interpretability in systems of this kind: either reduce the number of rules involved in the controller design or reduce the amount of antecedents in each of the rules considered. As the semantic similarity controllers already consider a maximum of two antecedents in each of the rules, for us the improvement of the interpretability will be given by a reduction of the set of rules.

4. Results

We present here the results of our experimental studies. To do that, we describe the experimental setup of our strategy including the benchmark datasets used, the goals to be achieved and the base configuration of the considered methods. After that, we perform an exhaustive analysis of the different multiobjetive strategies considered and the empirical results that we have been able to achieve. Moreover, we present a comparison with existing works including both those works that pay attention to the intepretability as well as those works that just focus on the accuracy. We also include a time analysis of the considered methods and we conclude with the discussion of the results that we have achieved.

4.1. Experimental setup

First, we explain the data set we have worked with, the objective functions that our strategy should pursue, and finally the base configuration that we have used in our tests and that should guarantee the repeatability of the experiments.

4.1.1. Datasets

Our experiments are based on a dataset that is de-facto standard to work with general-purpose oriented solutions. It is called the Miller & Charles [57] dataset. This dataset measures the likeness between textual information from several general-purpose scenarios, i.e., terms found in numerous situations. Therefore, it is formed by words that in principle one could expect to find in almost any database, document, map or even website. Please observe that we are working here with the version that contains 30 wordpairs (MC30), as many authors use shorter versions (e.g. 28 wordpairs) since their methods cannot work with words that are not covered in dictionaries such as Wordnet [1].

4.1.2. Goal

For the fitness function, we have two ways to guide our learning process. The Pearson Correlation Coefficient and the Spearman Rank correlation. The Pearson Correlation Coefficient is calculated between two vectors. It is defined as follows:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\ \sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

Being x the vector representing the results from the human, and y the vector from the solution

The other alternative is to use the Spearman Rank correlation which is a coefficient to measure the degree of association between the human-generated and the machine-generated vectors. The Spearman

[1] https://wordnet.princeton.edu/

Rank correlation is the appropriate goal when the results have to be compared on an ordinal scale. It is defined as follows:

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

Being $d_i = rg(X_i) - rg(Y_i)$ the difference between the two ranks of each vectors.

The difference between the two correlation coefficients is that the first is most appropriate in those scenarios involving an absolute scale, while the second is more suitable for scenarios involving ordinal scales. This means that when we want to produce absolute values, then we should train the model with Pearson. And if we want a model to produce relative ordering, then we should train the model using Spearman.

4.1.3. Configuration

Concerning the learning process, it is necessary to remark that in order for all experiments to be performed under equal conditions, we have searched for the optimal number of iterations. That is, the number of iterations beyond which most strategies do not achieve improvements over the previously obtained results. The result obtained is close to 10,000 iterations, so we have imposed this stopping condition to all the evaluated strategies.

The learning process is guided by the optimization of values in the training phase. However, the values reported are obtained by running the controller obtained by training on the data set in a blind manner in what is called the test phase. Almost certainly, there will be a gap between the results of both phases, being the value reported, the one obtained by the test phase, since it has not had access to the ground truth. Therefore, there is always certain degree of additional error associated. Since the methods we work with are stochastic strategies, this process is repeated by performing each experiment 10 times independently. Therefore, it must be taken into account that the reported values are the result of an average of 10 independent executions.

Moreover, it is necessary to point out that all the experiments have been performed on a regular PC holding Microsoft Windows 10 64-bit over a processor Intel Core i7-8700 at 3.20Ghz and 32 GB of main memory.

4.2. Analysis of strategies

As we have already mentioned, multiobjective learning aims to simultaneously learn a function that needs to meet several contradictory objectives simultaneously. For problems of this kind, there does not exist a single optimal solution. Therefore, in this work, our proposed approach is compared, using the ten most representative multiobjective strategies of the state-of-the-art in the area. Although a detailed description of how such techniques work is beyond the scope of this work, a detailed technical description can be found in [26] which is, in fact, the work that inspires the selection of our methods.

In our specific case, we seek to minimize the number of rules (what in practice means to increase interpretability [3]) and at the same time maximize the correlation between human judgment and the results of our proposal (increase accuracy). The problem is that both objectives are orthogonal or contradictory, so we need to resort to multiobjective approaches. In addition, it is convenient to highlight one important issue: the semantic similarity controller acts as a guide to achieve the best possible result on a set of training data. However, the results that we report are achieved on a blind data set (test set) since this is the way to verify that the controller has been able to learn a correct configuration capable of generalizing the results.

It is necessary to remark that our individuals support the possible encoding of up to 20 fuzzy rules. However, this is the maximum number, and the design process would rarely require such a large number of rules as it would cause over-fitting and would be detrimental to interpretability at the same time. The multiobjective strategies are designed to minimize the number of rules while attempting to maximize accuracy. Experimental results show that lower number rules can lead to the achievement of excellent accuracy values.

Our implementation is based on the Java programming language. Moreover, we rely on the frame-

work JFuzzyLogics [18] that is used as a virtual machine for the fitness assessment of the different semantic similarity controllers. And last, but not least, concerning the different multiobjetive approaches, we rely on the frameworks MOEA[2] and JMetal [9]. The different strategies studied following the alphabetical order are: CellDE [25], CMAES [35], DBEA [40], GDE3[42], MOEAD/D[67], MSOPS [34], NSGA-II [21], NSGA-III [22], PAES [41], and SMPSO [58]. These methods have been chosen as the most significant after a preliminary evaluation of a wide range of strategies. Thus, methods such as PESA2 [20], SPEA [11], SMS-EMOA [10] and IBEA [60] have been discarded because they performed worse in terms of Pareto front solutions found.

4.2.1. CellDE

CellDE algorithm stands for Cellular Genetic Algorithm with Differential Evolution [25]. This approach is a variation of the MOCell algorithm hybridized with differential evolution, that it is well-known for its good results for global optimization. In fact, CellDE also obtains very good results for a double reason: first, as mentioned, it is based on the highly efficient differential evolution, and second, CellDE takes from other multiobjective approaches the idea of best-solution archive, that store non-dominated solutions. This two design features makes the algorithm to obtain very good results when managing a wide range of multiobjective optimization problems of different domains.

As we can see in Figure 1, it is possible to get less error (which is equivalent to greater accuracy) by having more complex models and vice versa. It can also be observed that the results of the training phase are a little higher than those of the test phase as is logical since the test is executed over a blind sample. It would be at the expense of the human operator to choose the configuration that best fits the problem to be faced.

[2] http://moeaframework.org/

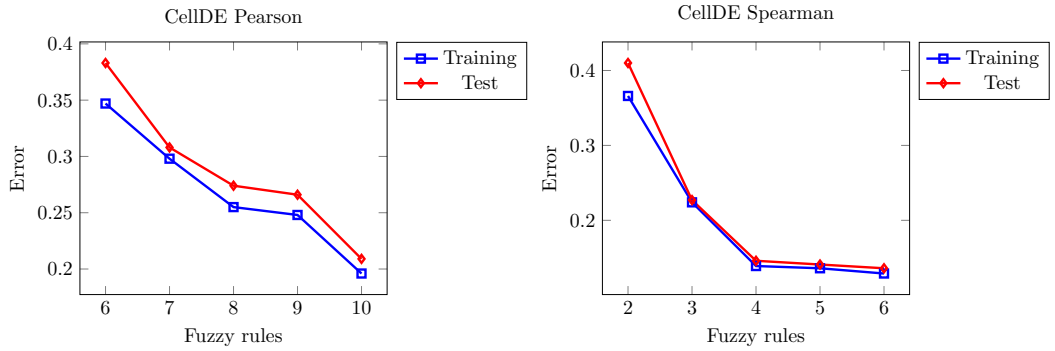

Figure 1: Pareto front of non-dominated solutions obtained using CellDE. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

4.2.2. CMAES

CMA-ES stands for Covariance Matrix Adaptation Evolution Strategy [35]. Evolution strategies (ES) are stochastic, derivative-free methods for numerical optimization of non-linear or non-convex continuous optimization problems. The results obtained with this approach are not as good as the results obtained with other classic MOEAs. Its design is based on two main ideas: the principle of maximum likelihood and the record of two evolution paths, which are compared to control the correlation between consecutive iterations and avoid local optima.

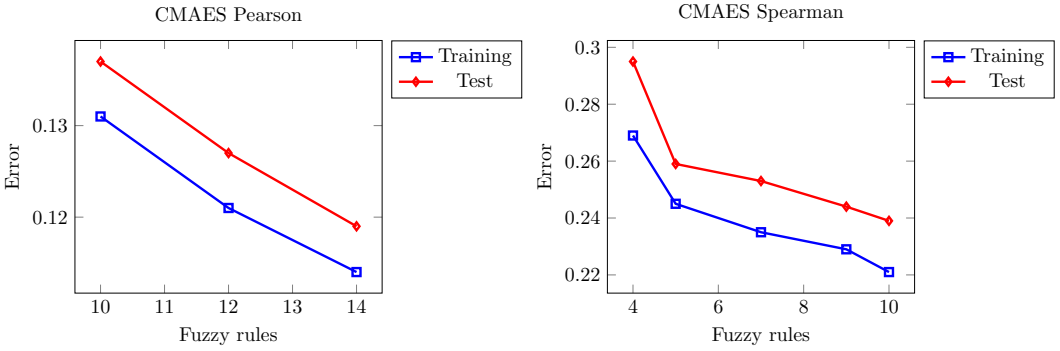

Figure 2: Pareto front of non-dominated solutions obtained using CMAES. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

In Figure 2, a classic Pareto front can be observed, where as a more simple to interpret model is

obtained, less accuracy is achieved and vice versa. Also, the training and test front never converge. It is also remarkable the disparity of accuracies obtained in relation to the Pearson correlation coefficient and Spearman rank correlation.

4.2.3. DBEA

DBEA stands for Improved Decomposition-Based Evolutionary Algorithm. While convergence-first sorting has continuously shown effectiveness for handling a variety of problems, it faces challenges to maintain well population diversity due to the overemphasis of convergence. DBEA is a general diversity-first sorting method for multiobjective optimization [40]. In general, decomposition based EAs perform better when applied on optimization problems involving many objectives (more than 3). DBEA depends on 3 key factors: first, the wisely distributed reference point generation of the model, second, the scheme definition to simultaneously deal with convergence and diversity in the search, and finally, the association between the solutions and the reference directions of the search strategy.

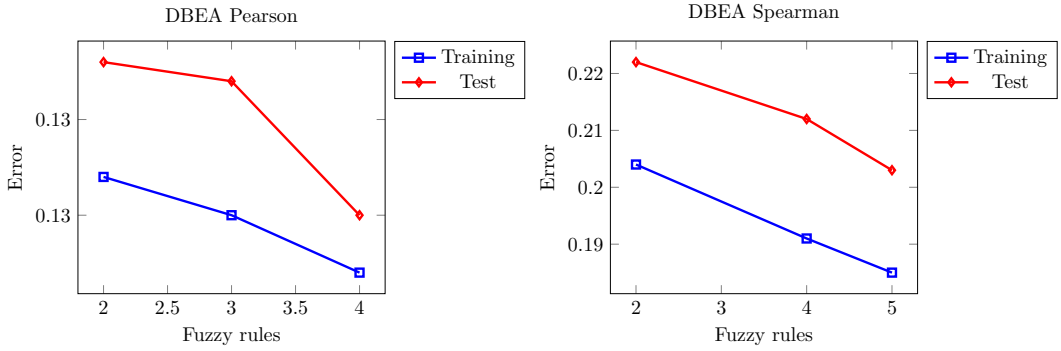

Figure 3: Pareto front of non-dominated solutions obtained using DBEA. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

As can be seen in Figure 3, the results show a Pareto front for both correlation coefficients. Although the results obtained for the Pearson correlation coefficient are, in general, better than those obtained for the Spearman Rank Correlation. Moreover, the training phase yields better accuracy results than the training phase as is obvious.

4.2.4. GDE3

Generalized Differential Evolution (GDE3) algorithm [42]. Differential Evolution optimizes a problem by maintaining a population of candidate solutions and creating new candidate solutions by combining existing ones according to its simple formulae. Differential evolution metaheuristics are simple, efficient and, in general, obtain very competitive results due to the design of the evolution strategy, which is based on a controlled differential mutation operation over the best solutions found.

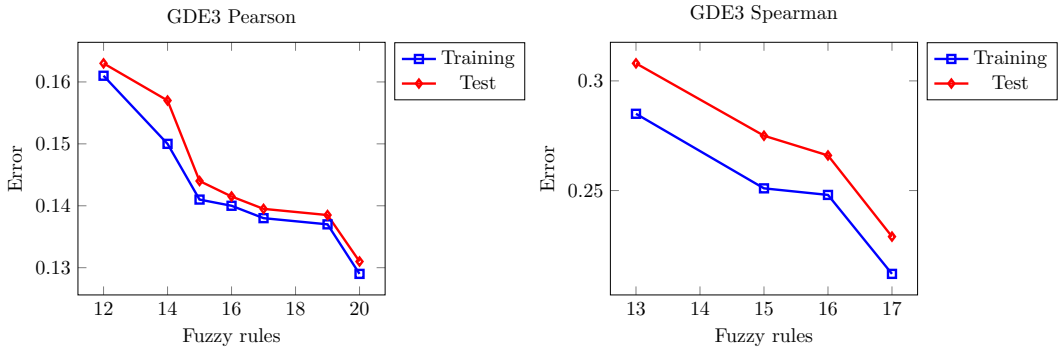

Figure 4: Pareto front of non-dominated solutions obtained using GDE3. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

From the experiments, it is possible to see that one of the most successful approaches is GDE3. In fact, in Figure 4, it is possible to see the great results that GDE is able to achieve for the Pearson correlation coefficient since the error is quite low. However, as we have also observed in the rest of the experiments, the results achieved for Pearson are better than those obtained for Spearman.

4.2.5. MOEA/D

MOEA/D stands for MultiObjective Evolutionary Algorithm with Decomposition [67]. MOEA/D is a relatively new optimization algorithm based on the concept of decomposing the problem into many single-objective formulations. This algorithm performs better than the classic DBEA, but as occurs with other decomposition based MOEAs, MOEA/D works much better with multiobjective problems involving more than 3 conflicting objectives.

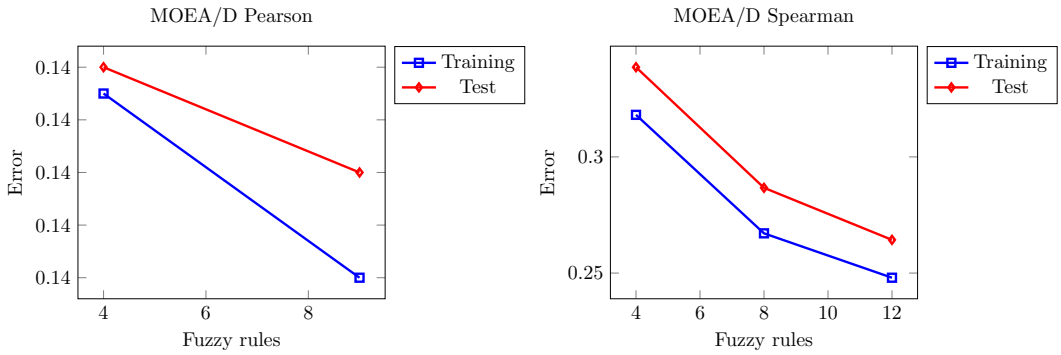

Figure 5: Pareto front of non-dominated solutions obtained using MOEA/D. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

As we can see in the Figure 5, we have obtained a Pareto front for the two cases (Pearson and Spearman). This is due to the orthogonality between the accuracy and the number of rules needed for the model. Of course, once again, it can be seen that the training values are slightly higher than the test values.

4.2.6. MSOPS

MSOPS stands for Multiple Single-Objective Pareto Sampling. Implementation of the Multiple Single Objective Pareto Sampling (MSOPS) algorithm [34], which performs a parallel search of multiple single objective optimizations. The strategy allows bounds and discontinuities of the Pareto front to be identified, and it works with few and many objectives.

Again, Figure 6 shows us that we have obtained the Pareto front in which it can be clearly observed that higher complexity equals better results in both the training and test phases (and vice versa). Again the results seem to be better for the Pearson correlation coefficient than for the Spearman Rank Correlation.

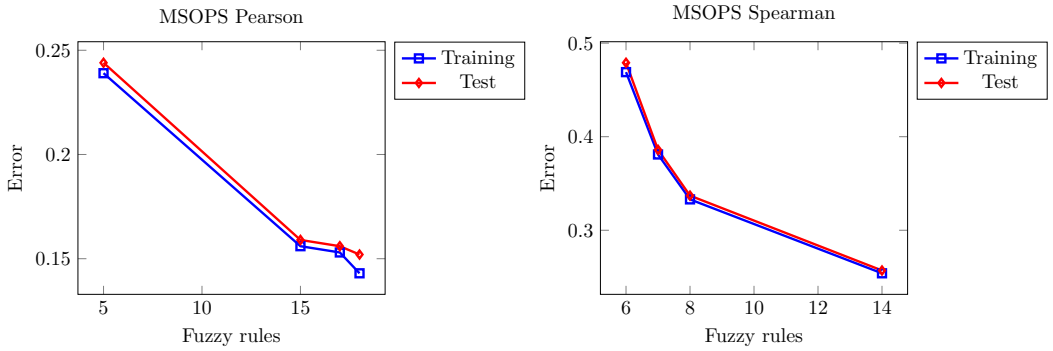

Figure 6: Pareto front of non-dominated solutions obtained using MSOPS. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

4.2.7. NSGA-II

NSGA-II stands for Non-dominated Sorting Genetic Algorithm II [21]. NSGA-II simultaneously optimizes each objective without being dominated by any other solution. It is one of the best approaches to work when the number of goals is not too high, in fact, NSGA-II is one of the most successful MOEAs in the specific bibliography. It is the main standard in multiobjective optimization. The algorithm works with the core concept of dominance, i.e. solutions which are better than others because they improve one of the objectives and they are not worse in any of the other objectives.

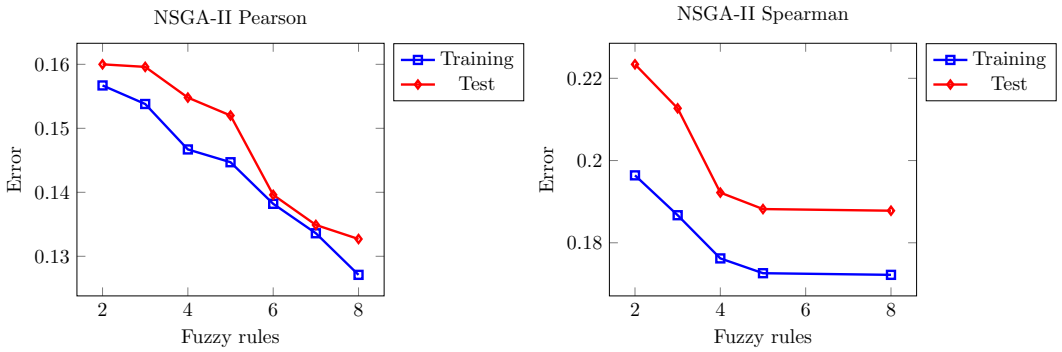

Figure 7: Pareto front of non-dominated solutions obtained using NSGA-II. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

NSGA-II based solutions are generally quite good. It is true, that this strategy does not manage

to be the best in either case, but it is no less true that it is in the top 3 of best approaches for both cases at the same time. Something that no other strategy studied has achieved. This is because this strategy has proven its effectiveness when working with few simultaneous objectives, as in our case. Figure 7 shows us the results obtained by using NSGA-II.

4.2.8. NSGA-III

NSGA-III stands for Reference-Point Based Non-dominated Sorting Genetic Algorithm [22]. Reference Directions which need to be provided when the algorithm is initialized. It is a improvement over its parent strategy NSGA-II, but the strategy is specially designed for many objectives optimization problems (more than 3 or 4), so it does not work better than NSGA-II with problems involving only 2 objectives.

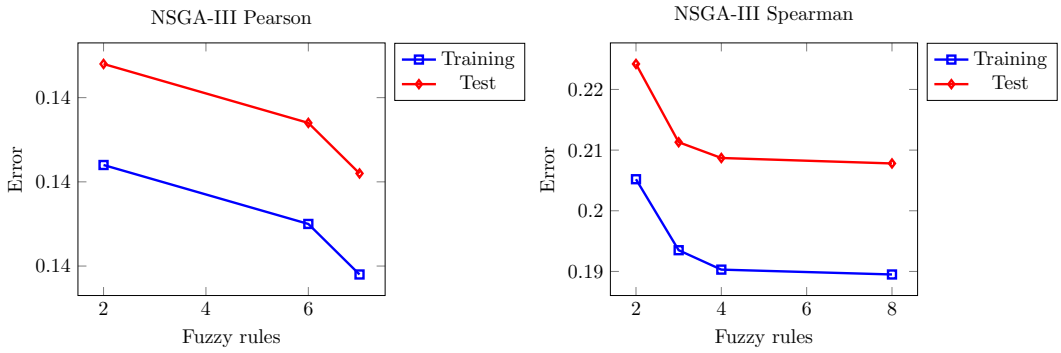

Figure 8: Pareto front of non-dominated solutions obtained using NSGA-III. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

In Figure 8, we can again see the generated Pareto front. NSGA-III is usually a method that works quite well when working with many targets, but in our case, we only have two targets, so the results are not spectacular. In fact, they are slightly worse than those obtained with its previous version NSGA-II which works much better for few targets.

4.2.9. PAES

PAES stands for Pareto Archived Evolution Strategy. This method uses an adaptive grid archive to maintain a diverse set of solutions [41]. This classic algorithm is usually surpassed by other multiobjective approaches, like CellDE, which takes one of the best design strength from PAES: the archive of solutions.

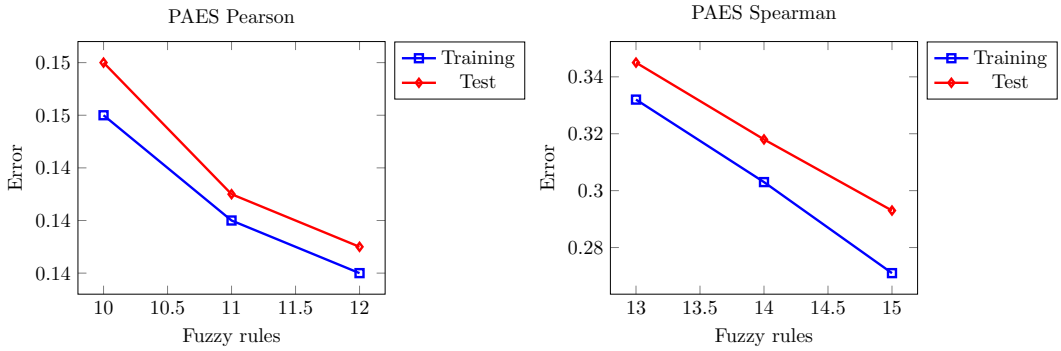

Figure 9: Pareto front of non-dominated solutions obtained using PAES. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

Figure 9 shows us the results yielded by PAES. As can be seen, the Pareto fronts obtained are not very populated. Or at least, they are less populated than with other solutions.

4.2.10. SMPSO

SMPSO stands for Speed-Constrained Multiobjective Particle Swarm Optimization [58]. SMPSO is a multiobjective particle swarm optimization algorithm. These algorithms create artificial particles and move them in the search space using some formulas based on position and velocity. Thus, the algorithm is a multiobjective version of a swarm intelligence approach. In general, swarm intelligence metaheuristics achieve very competitive results for a wide range of optimization problems, so it also obtains very good results applied to multiobjective problems.

As in the previous cases, the Pareto fronts obtained for both the Pearson and Spearman cases can be seen. In Figure 10, we can see how the red line marks the training results while the blue line

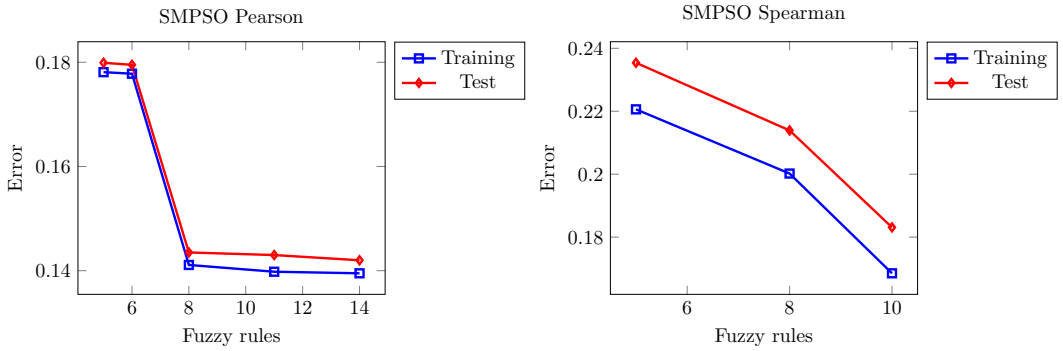

Figure 10: Pareto front of non-dominated solutions obtained using SMPSO. First plot represents the Pearson Correlation Coefficient, second plot represents the Spearman Rank Correlation

corresponds to the test results. As can be seen, the error (which is equivalent to the accuracy) is higher when there are fewer rules (more interpretability) and vice versa.

4.3. Comparison with existing works

Now we are going to proceed to compare the best solutions obtained using the multiobjective learning strategy. Since our proposal is the first to study the trade-off between accuracy and interpretability in the field of semantic similarity, we cannot directly compare it with any other proposal. For this reason, we will first compare the results with other techniques that also give importance to interpretability. And secondly, we compare with the wide spectrum of existing methods for automatic calculation of semantic similarity.

4.3.1. Comparison with interpretable methods

The first comparison we wish to make is with existing proposals that focus on interpretability. These approaches are mainly based on fuzzy logic or symbolic regression methods as mentioned in [55]. Figure 11 shows the boxplots corresponding to the results. In the first study, we have focused on the performance of the best approximations guided towards the optimization of the Pearson correlation coefficient.

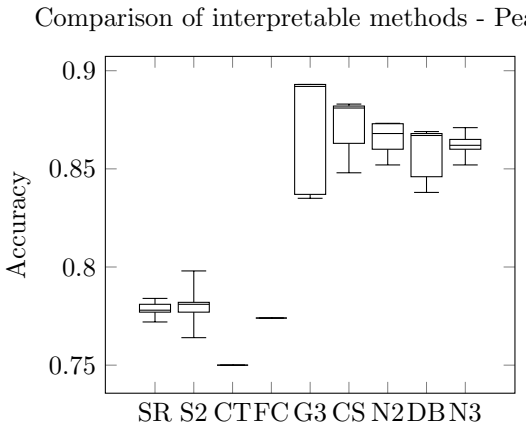

Figure 11: Results of the interpretable approaches for the calculation of the **Pearson Correlation Coefficient** over the MC30 benchmark dataset. SR = Symbolic Regression, S2 = Symbolic Regression with optimization of the Operator Precedence, CT = Consensus or Trade-Off, FC = Fuzzy Logic Controller, G3 = GDE3, CS = CMAES, N2 = NSGA-II, DB = DBEA, and N3 = NSGA-III

In Figure 12, we can see the results obtained by the classical techniques in the form of a boxplot. The methods that we can see are those that try to optimize the interpretability of the learned models plus our five best approximations. As it can be seen, although the variance is high, our approaches are able to obtain higher median results, being CellDE, SMPSO, and NSGA-II the ones capable of achieving the best median results, which can be explained because the strategies based on the differential evolution approach, the swarm intelligence methods and the GA based on the strong concept of dominance are very reliable and highly efficient strategies which provide very good results in most of the multiobjective optimization problems. Moreover, the amount of fuzzy rules they need does not seem excessively high.

4.3.2. Comparison with state-of-the-art

Despite the fact that accuracy and interpretability are antagonistic goals [5], it must be taken into account that accuracy is still an important factor because there is no point in having a highly interpretable solution that gives bad results. We show Table 1 whereby we present the five best results we have obtained from our empirical study for the Pearson Correlation Coefficient in comparison with

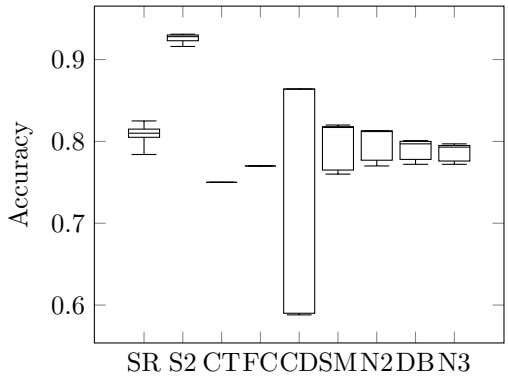

Figure 12: Results of the interpretable approaches for the calculation of **Spearman Rank Correlation** over the Miller & Charles dataset. SR = Symbolic Regression, S2 = Symbolic Regression with optimization of the Operator Precedence, CT = Consensus or Trade-Off, FC = Fuzzy Logic Controller, CD = CellDE, SM = SMPSO, N2 = NSGA-II, DB = DBEA, N3 = NSGA-III

the best existing approaches for solving the MC30. Although the results are dependent on how the model is trained, we can observe that some configurations are capable of obtaining results above those obtained by classical methods. Since we are working with nondeterministic solutions, we refer to the median value.

In Table 2, we show the five best results we have obtained when solving the MC30 dataset benchmark using Spearman Rank Correlation. In fact, we report the state-of-the-art plus the five best results achieved following our approach. As it is possible to see, some of the configurations are capable of obtaining results above those obtained by classic methods. However, there is more diversity than in the previous case. In addition, the amount of fuzzy rules required to obtain these results is not too high.

As a result of our experimental study, it is possible to observe that our approach is able to place different configurations among the best solutions in relation to the MC30 benchmark dataset. However, the solutions are, in general, better for the Pearson correlation coefficient than for the Spearman Rank Correlation. This is because these strategies seem to work better with absolute similarity values than with relative values.

Algorithm	Score	p-value
Google distance [17]	0.470	$8.8 \cdot 10^{-3}$
Huang et al. [33]	0.659	$7.5 \cdot 10^{-5}$
Jiang & Conrath [39]	0.669	$5.3 \cdot 10^{-5}$
Resnik [62]	0.780	$1.9 \cdot 10^{-7}$
Leacock & Chodorow [45]	0.807	$4.0 \cdot 10^{-8}$
Lin [47]	0.810	$3.0 \cdot 10^{-8}$
Faruqui & Dyer [27]	0.817	$2.0 \cdot 10^{-8}$
Mikolov et al. [56]	0.820	$2.2 \cdot 10^{-8}$
CoTO [52]	0.850	$1.0 \cdot 10^{-8}$
FLC [54]	0.855	$1.0 \cdot 10^{-8}$
NSGA-III, 7 rules	0.863	$1.0 \cdot 10^{-8}$
DBEA, 4 rules	0.867	$8.5 \cdot 10^{-9}$
NSGA-II, 8 rules	0.873	$3.5 \cdot 10^{-9}$
CMAES, 14 rules	0.886	$3.2 \cdot 10^{-9}$
GDE3, 12 rules	0.891	$1.1 \cdot 10^{-9}$

Table 1: Results for the **Pearson Correlation Coefficient** achieved by existing methods over the MC30 dataset

Algorithm	Score	p-value
Jiang & Conrath [39]	0.588	$4.0 \cdot 10^{-4}$
Lin [47]	0.619	$1.6 \cdot 10^{-4}$
Aouicha et al. [8]	0.640	$8.0 \cdot 10^{-5}$
Resnik [61]	0.757	$5.3 \cdot 10^{-7}$
Mikolov et al. [56]	0.770	$2.6 \cdot 10^{-7}$
Leacock & Chodorow [45]	0.789	$8.1 \cdot 10^{-8}$
NSGA-III, 8 rules	0.793	$2.1 \cdot 10^{-14}$
DBEA, 5 rules	0.797	$2.1 \cdot 10^{-14}$
NSGA-II, 8 rules	0.812	$1.1 \cdot 10^{-14}$
SMPSO, 10 rules	0.817	$1.9 \cdot 10^{-8}$
Bojanowski et al. [13]	0.846	$6.3 \cdot 10^{-9}$
Zhao et al. [68]	0.857	$1.4 \cdot 10^{-9}$
CellDE, 6 rules	0.864	$1.1 \cdot 10^{-10}$
FLC [54]	0.891	$8.3 \cdot 10^{-12}$

Table 2: Results for the **Spearman Rank Correlation** achieved by existing methods over the MC30 dataset

Finally, it is also interesting to note that although it is true that the number of fuzzy rules needed to obtain these results is not very low, it is not excessive either. Therefore, a well-trained human operator should not have major problems understanding the model.

4.4. Time analysis

Last but not least, it is necessary to study the time required to obtain the Pareto fronts of the solutions we have seen. In Figure 13, we show the average execution time for each of the multiobjective strategies considered. These times represent, once again, the average time resulting from a total of 10 independent runs of each of the strategies.

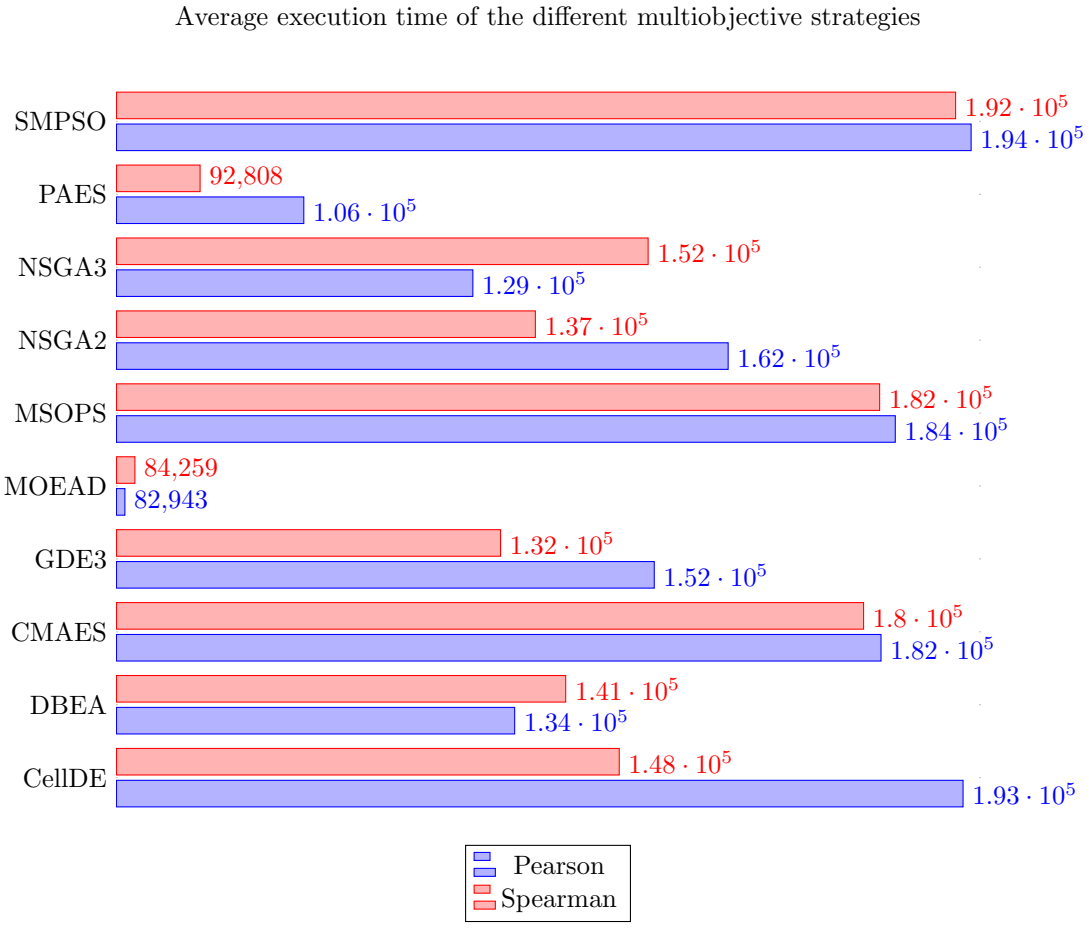

Figure 13: Average execution time of the multiobjective strategies considered when solving the MC30 benchmark dataset

As can be seen, the MOEA/D technique is the fastest of all. However, this technique fails to place its results among the state-of-the-art. Techniques such as GDE3 or NSGA-II, which show much better results, require longer execution times to obtain the respective Pareto fronts. It is up to the human operator to decide whether to settle for results obtained more quickly, or to wait a little longer and obtain a better median accuracy and interpretability.

4.5. Discussion

We have seen that semantic similarity controllers are systems being able to analyze semantic similarity values and produce a meaningful score through a complex yet human-understandable aggregation strategy. The high interpretability levels that can be achieved using semantic similarity controllers can represent the aggregation strategy through rules that have an easy translation into the human language. Moreover, this kind of controller can approximate any nonlinear function since it has the behavior of a universal approximator [15] so it must be able to model any kind of semantic correspondence relationship.

However, when designing automatic controllers, there is a risk of obtaining models with many rules that are very difficult to interpret. Although the accuracy that could be achieved might be good, if a human operator cannot understand the mode of operation, it would not be helpful in many cases. The advantage of multiobjective techniques is that they allow an automatic design in which the decision about the final configuration falls on the human operator. In some cases, the human operator may prefer greater accuracy, while it might be preferred to have better interpretability in other cases. And this is precisely where our proposal contributes.

From our empirical study, it can be observed that there is variety in the results obtained. For example, GDE3 achieved the best results when studying the Pearson correlation, and CellDE was the best when studying the Spearman Rank. However, we do not want it to go unnoticed that NSGA-II is one of the most reliable algorithms in this context since it has always been among the best methods in all the studied cases. This is because the methods based on the notion of dominance are very reliable

and can implement highly efficient strategies, which provide excellent results in most multiobjective learning scenarios. Concerning execution times, there is also some variability. But because the training phase only has to be performed once, since the learned model will be put into operation, the different times obtained are not significant enough to make a big difference.

In summary, we have seen how our approach can facilitate decision-making to human operators in semantic similarity measurement. Even in the case that ANN can achieve high levels of accuracy. Our approach is better at interpretability of the resulting model, reduce the number of training resources, and facilitate the transfer learning processes. However, it is necessary to analyze various strategies to determine the optimal one for the chosen scenario. If it not possible to count on the required time or means, it should always be possible to rely on a reasonably reliable method such as NSGA-II. In addition, and as a general rule, these lessons learned could be of particular interest in a wide range of application domains that are currently dominated by black-box solutions.

5. Conclusions

In a time where big data and data analysis are important players in many application scenarios, the need for people to trust the data-driven systems they use for their daily operations is crucial. However, in recent times, the field of semantic similarity is involved in a race to improve accuracy over and over again. This issue has caused systems to pay little attention to their interpretability.

To overcome this problem, we have presented a novel approach for establishing a proper trade-off between accuracy and interpretability when setting up novel semantic similarity controllers. The rationale behind our proposal is based on the strategic aggregation of simple semantic similarity measures using a multiobjective learning approach. Such a multiobjective approach is necessary since we are trying to model proper interactions between orthogonal goals. Our proposal overcomes the traditional problems associated with the neural solutions that are often characterized by the growing presence of models being accurate but incomprehensible, the excessive consumption of resources for

their training, and their inability to extrapolate the knowledge learned.

We have studied the behavior of multiobjective approaches considered to be state-of-the-art. Our results show how it is possible to find a front of solutions where the human operator can define the tolerance thresholds. Our results are promising because we have been able to achieve fairly good accuracy values with semantic similarity controllers with high degrees of interpretability. Furthermore, the model does not require large amounts of data for training, and its ease of understanding facilitates its application in analogous domains. Therefore, our results could be of particular relevance in environments where a few hundredths of additional accuracy does not compensate for the lack of interpretability of the models needed.

As future work, it would be desirable to compare various families of solutions for designing the semantic similarity controllers. For example, it could be of great interest to compare the Active Learning-based solutions [14] or the pair-wise methods [1] with the family of evolutionary algorithms to compare their performance in implementing systems for automatic assessment of semantic similarity.

Competing interest

Authors have no competing interest to declare.

Acknowledgments

We would like to thank in advance the anonymous reviewers for their help towards improving this work. This research work has been partially supported by the Austrian Ministry for Transport, Innovation and Technology, the Federal Ministry of Science, Research and Economy, and the Province of Upper Austria in the frame of the COMET center SCCH. It was also supported by 4IE+ project (0499_4IE_PLUS_4_E) funded by the Interreg V-A Spain-Portugal (POCTEP) 2014-2020 program and by RTI 2018-094591-B-I00 (MCIU/AEI/FEDER, UE) project.

References

[1] Aggarwal, M. (2019). Learning of a decision-maker's preference zone with an evolutionary approach. *IEEE Trans. Neural Networks Learn. Syst.*, *30*, 670–682. URL: https://doi.org/10.1109/TNNLS.2018.2847412. doi:10.1109/TNNLS.2018.2847412.

[2] Alcalá, R., Gacto, M. J., & Herrera, F. (2011). A fast and scalable multiobjective genetic fuzzy system for linguistic fuzzy modeling in high-dimensional regression problems. *IEEE Trans. Fuzzy Syst.*, *19*, 666–681. URL: https://doi.org/10.1109/TFUZZ.2011.2131657. doi:10.1109/TFUZZ.2011.2131657.

[3] Alcalá-Fdez, J., Alcalá, R., Gacto, M. J., & Herrera, F. (2009). Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms. *Fuzzy Sets and Systems*, *160*, 905–921. doi:10.1016/j.fss.2008.05.012.

[4] Alonso, J. M., Castiello, C., & Mencar, C. (2015). Interpretability of fuzzy systems: Current research trends and prospects. In *Springer handbook of computational intelligence* (pp. 219–237). Springer.

[5] Alonso, J. M., & Magdalena, L. (2011). HILK++: an interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers. *Soft Comput.*, *15*, 1959–1980. doi:10.1007/s00500-010-0628-5.

[6] Angelov, P. P., & Buswell, R. A. (2003). Automatic generation of fuzzy rule-based models from data by genetic algorithms. *Inf. Sci.*, *150*, 17–31. doi:10.1016/S0020-0255(02)00367-5.

[7] Antonelli, M., Ducange, P., Lazzerini, B., & Marcelloni, F. (2009). Multi-objective evolutionary learning of granularity, membership function parameters and rules of mamdani fuzzy systems. *Evolutionary Intelligence*, *2*, 21–37. doi:10.1007/s12065-009-0022-3.

[8] Aouicha, M. B., Taieb, M. A. H., & Hamadou, A. B. (2016). LWCR: multi-layered wikipedia

representation for computing word relatedness. *Neurocomputing*, *216*, 816–843. URL: https://doi.org/10.1016/j.neucom.2016.08.045. doi:10.1016/j.neucom.2016.08.045.

[9] Barba-González, C., García-Nieto, J., Nebro, A. J., Cordero, J. A., Durillo, J. J., Delgado, I. N., & Montes, J. F. A. (2018). jmetalsp: A framework for dynamic multi-objective big data optimization. *Appl. Soft Comput.*, *69*, 737–748. URL: https://doi.org/10.1016/j.asoc.2017.05.004. doi:10.1016/j.asoc.2017.05.004.

[10] Beume, N., Naujoks, B., & Emmerich, M. T. M. (2007). SMS-EMOA: multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.*, *181*, 1653–1669. URL: https://doi.org/10.1016/j.ejor.2006.08.008. doi:10.1016/j.ejor.2006.08.008.

[11] Bleuler, S., Brack, M., Thiele, L., & Zitzler, E. (2001). Multiobjective genetic programming: Reducing bloat using spea2. In *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546)* (pp. 536–543).

[12] Bodenhofer, U., & Bauer, P. (2003). A formal model of interpretability of linguistic variables. In *Interpretability issues in fuzzy modeling* (pp. 524–545). Springer.

[13] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *TACL*, *5*, 135–146.

[14] Campigotto, P., Passerini, A., & Battiti, R. (2014). Active learning of pareto fronts. *IEEE Trans. Neural Networks Learn. Syst.*, *25*, 506–519. URL: https://doi.org/10.1109/TNNLS.2013.2275918. doi:10.1109/TNNLS.2013.2275918.

[15] Castro, J. L., & Delgado, M. (1996). Fuzzy systems with defuzzification are universal approximators. *IEEE Trans. Systems, Man, and Cybernetics, Part B*, *26*, 149–152. doi:10.1109/3477.484447.

[16] Chaves-Gonzalez, J. M., & Martinez-Gil, J. (). Evolutionary algorithm based on different semantic

similarity functions for synonym recognition in the biomedical domain. *Knowl. Based Syst.*, *37*, 62–69. doi:10.1016/j.knosys.2012.07.005.

[17] Cilibrasi, R., & Vitányi, P. M. B. (2007). The google similarity distance. *IEEE Trans. Knowl. Data Eng.*, *19*, 370–383. URL: https://doi.org/10.1109/TKDE.2007.48. doi:10.1109/TKDE.2007.48.

[18] Cingolani, P., & Alcalá-Fdez, J. (2013). jfuzzylogic: a java library to design fuzzy logic controllers according to the standard for fuzzy control programming. *Int. J. Comput. Intell. Syst.*, *6*, 61–75. doi:10.1080/18756891.2013.818190.

[19] Cordón, O. (2011). A historical review of evolutionary learning methods for mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *Int. J. Approx. Reason.*, *52*, 894–913. URL: https://doi.org/10.1016/j.ijar.2011.03.004. doi:10.1016/j.ijar.2011.03.004.

[20] Corne, D. W., Jerram, N. R., Knowles, J. D., & Oates, M. J. (2001). Pesa-ii: Region-based selection in evolutionary multiobjective optimization. In *Proceedings of the 3rd annual conference on genetic and evolutionary computation* (pp. 283–290).

[21] Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, *6*, 182–197. URL: https://doi.org/10.1109/4235.996017. doi:10.1109/4235.996017.

[22] Deb, K., & Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Trans. Evol. Comput.*, *18*, 577–601. URL: https://doi.org/10.1109/TEVC.2013.2281535. doi:10.1109/TEVC.2013.2281535.

[23] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci.*, *41*, 391–407.

[24] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, .

[25] Durillo, J. J., Nebro, A. J., Luna, F., & Alba, E. (2008). Solving three-objective optimization problems using a new hybrid cellular genetic algorithm. In G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, & N. Beume (Eds.), *Parallel Problem Solving from Nature - PPSN X, 10th International Conference Dortmund, Germany, September 13-17, 2008, Proceedings* (pp. 661–670). Springer volume 5199 of *Lecture Notes in Computer Science*. URL: https://doi.org/10.1007/978-3-540-87700-4_66. doi:10.1007/978-3-540-87700-4_66.

[26] Emmerich, M. T. M., & Deutz, A. H. (2018). A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Nat. Comput.*, *17*, 585–609. URL: https://doi.org/10.1007/s11047-018-9685-y. doi:10.1007/s11047-018-9685-y.

[27] Faruqui, M., & Dyer, C. (2014). 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* (pp. 462–471).

[28] Fazzolari, M., Alcalá, R., Nojima, Y., Ishibuchi, H., & Herrera, F. (2013). A review of the application of multiobjective evolutionary fuzzy systems: Current status and further directions. *IEEE Trans. Fuzzy Syst.*, *21*, 45–65. URL: https://doi.org/10.1109/TFUZZ.2012.2201338. doi:10.1109/TFUZZ.2012.2201338.

[29] Gacto, M. J., Alcalá, R., & Herrera, F. (2010). Integration of an index to preserve the semantic interpretability in the multiobjective evolutionary rule selection and tuning of linguistic fuzzy systems. *IEEE Trans. Fuzzy Syst.*, *18*, 515–531. URL: https://doi.org/10.1109/TFUZZ.2010.2041008. doi:10.1109/TFUZZ.2010.2041008.

[30] Gacto, M. J., Alcalá, R., & Herrera, F. (2011). Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences*, *181*, 4340–4360.

[31] Han, L., Finin, T., McNamee, P., Joshi, A., & Yesha, Y. (2013). Improving word similarity by augmenting PMI with estimates of word polysemy. *IEEE Trans. Knowl. Data Eng.*, *25*, 1307–1322. doi:10.1109/TKDE.2012.30.

[32] Han, L., Kashyap, A. L., Finin, T., Mayfield, J., & Weese, J. (2013). Umbc_ebiquity-core: Semantic textual similarity systems. In M. T. Diab, T. Baldwin, & M. Baroni (Eds.), *Proceedings of the Second Joint Conference on Lexical and Computational Semantics, *SEM 2013, June 13-14, 2013, Atlanta, Georgia, USA* (pp. 44–52). Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/S13-1005/.

[33] Huang, E. H., Socher, R., Manning, C. D., & Ng, A. Y. (2012). 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* (pp. 873–882).

[34] Hughes, E. J. (2003). Multiple single objective pareto sampling. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.* (pp. 2678–2684). IEEE volume 4.

[35] Igel, C., Hansen, N., & Roth, S. (2007). Covariance matrix adaptation for multi-objective optimization. *Evol. Comput.*, *15*, 1–28. URL: https://doi.org/10.1162/evco.2007.15.1.1. doi:10.1162/evco.2007.15.1.1.

[36] Ishibuchi, H., Nakashima, Y., & Nojima, Y. (2011). Performance evaluation of evolutionary multiobjective optimization algorithms for multiobjective fuzzy genetics-based machine learning. *Soft Comput.*, *15*, 2415–2434. URL: https://doi.org/10.1007/s00500-010-0669-9. doi:10.1007/s00500-010-0669-9.

[37] Ishibuchi, H., & Nojima, Y. (2007). Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning. *Int. J. Approx. Reason.*, *44*, 4–31. URL: https://doi.org/10.1016/j.ijar.2006.01.004. doi:10.1016/j.ijar.2006.01.004.

[38] Ishibuchi, H., & Nojima, Y. (2015). Multiobjective genetic fuzzy systems. In J. Kacprzyk, & W. Pedrycz (Eds.), *Springer Handbook of Computational Intelligence* Springer Handbooks (pp. 1479–1498). Springer. URL: https://doi.org/10.1007/978-3-662-43505-2_77. doi:10.1007/978-3-662-43505-2_77.

[39] Jiang, J. J., & Conrath, D. W. (1997). 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* (pp. 19–33).

[40] Jiang, S., & Yang, S. (2016). Convergence versus diversity in multiobjective optimization. In J. Handl, E. Hart, P. R. Lewis, M. López-Ibáñez, G. Ochoa, & B. Paechter (Eds.), *Parallel Problem Solving from Nature - PPSN XIV - 14th International Conference, Edinburgh, UK, September 17-21, 2016, Proceedings* (pp. 984–993). Springer volume 9921 of *Lecture Notes in Computer Science*. URL: https://doi.org/10.1007/978-3-319-45823-6_92. doi:10.1007/978-3-319-45823-6_92.

[41] Knowles, J. D., & Corne, D. (2000). Approximating the nondominated front using the pareto archived evolution strategy. *Evol. Comput.*, *8*, 149–172. URL: https://doi.org/10.1162/106365600568167. doi:10.1162/106365600568167.

[42] Kukkonen, S., & Lampinen, J. (2005). Gde3: The third evolution step of generalized differential evolution. In *2005 IEEE congress on evolutionary computation* (pp. 443–450). IEEE volume 1.

[43] Lastra-Díaz, J. J., García-Serrano, A., Batet, M., Fernández, M., & Chirigati, F. (2017). HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset. *Inf. Syst.*, *66*, 97–118. URL: https://doi.org/10.1016/j.is.2017.02.002. doi:10.1016/j.is.2017.02.002.

[44] Lastra-Díaz, J. J., Goikoetxea, J., Taieb, M. A. H., García-Serrano, A., Aouicha, M. B., & Agirre, E. (2019). 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. URL: https://doi.org/10.1016/j.engappai.2019.07.010. doi:10.1016/j.engappai.2019.07.010.

[45] Leacock, C., & Chodorow, M. (1998). Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, *49*, 265–283.

[46] Li, Y., Bandar, Z., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Trans. Knowl. Data Eng.*, *15*, 871–882. URL: https://doi.org/10.1109/TKDE.2003.1209005. doi:10.1109/TKDE.2003.1209005.

[47] Lin, D. (1998). 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* (pp. 296–304).

[48] Magdalena, L. (2020). Fuzzy systems interpretability: What, why and how. In *Fuzzy Approaches for Soft Computing and Approximate Reasoning: Theories and Applications* (pp. 111–122). Springer.

[49] Mamdani, E. H., & Assilian, S. (1999). An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Hum.-Comput. Stud.*, *51*, 135–147. doi:10.1006/ijhc.1973.0303.

[50] Martinez-Gil, J. (2014). An overview of textual semantic similarity measures based on web intelligence. *Artif. Intell. Rev.*, *42*, 935–943. URL: https://doi.org/10.1007/s10462-012-9349-8. doi:10.1007/s10462-012-9349-8.

[51] Martinez-Gil, J. (2016). Accurate semantic similarity measurement of biomedical nomenclature by means of fuzzy logic. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *24*, 291–306. doi:10.1142/S0218488516500148.

[52] Martinez-Gil, J. (2016). Coto: A novel approach for fuzzy aggregation of semantic similarity measures. *Cognitive Systems Research*, *40*, 8–17. doi:10.1016/j.cogsys.2016.01.001.

[53] Martinez-Gil, J. (2019). Semantic similarity aggregators for very short textual expressions: a case study on landmarks and points of interest. *J. Intell. Inf. Syst.*, *53*, 361–380. doi:10.1007/s10844-019-00561-0.

[54] Martinez-Gil, J., & Chaves-González, J. M. (2019). Automatic design of semantic similarity controllers based on fuzzy logics. *Expert Syst. Appl.*, *131*, 45–59. URL: https://doi.org/10.1016/j.eswa.2019.04.046. doi:10.1016/j.eswa.2019.04.046.

[55] Martinez-Gil, J., & Chaves-González, J. M. (2020). A novel method based on symbolic regression for interpretable semantic similarity measurement. *Expert Syst. Appl.*, *160*, 113663. URL: https://doi.org/10.1016/j.eswa.2020.113663. doi:10.1016/j.eswa.2020.113663.

[56] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). 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.* (pp. 3111–3119).

[57] Miller, G., & Charles, W. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, *6*, 1–28.

[58] Nebro, A. J., Durillo, J. J., García-Nieto, J., Coello, C. A. C., Luna, F., & Alba, E. (2009). SMPSO: A new pso-based metaheuristic for multi-objective optimization. In *2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, MCDM 2009, Nashville, TN, USA, March 30 - April 2, 2009* (pp. 66–73). IEEE. URL: https://doi.org/10.1109/MCDM.2009.4938830. doi:10.1109/MCDM.2009.4938830.

[59] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of NAACL-HLT* (pp. 2227–2237).

[60] Phan, D. H., & Suzuki, J. (2013). R2-IBEA: R2 indicator based evolutionary algorithm for

multiobjective optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2013, Cancun, Mexico, June 20-23, 2013* (pp. 1836–1845). IEEE. URL: https://doi.org/10.1109/CEC.2013.6557783. doi:10.1109/CEC.2013.6557783.

[61] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, IJCAI 95, Montréal Québec, Canada, August 20-25 1995, 2 Volumes* (pp. 448–453). URL: http://ijcai.org/Proceedings/95-1/Papers/059.pdf.

[62] Resnik, P. (1999). 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. doi:10.1613/jair.514.

[63] Rychalska, B., Pakulska, K., Chodorowska, K., Walczak, W., & Andruszkiewicz, P. (2016). Samsung poland NLP team at semeval-2016 task 1: Necessity for diversity; combining recursive autoencoders, wordnet and ensemble methods to measure semantic similarity. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016* (pp. 602–608). URL: https://www.aclweb.org/anthology/S16-1091/.

[64] Sánchez, L., Otero, J., & Couso, I. (2009). Obtaining linguistic fuzzy rule-based regression models from imprecise data with multiobjective genetic algorithms. *Soft Comput.*, *13*, 467–479. URL: https://doi.org/10.1007/s00500-008-0362-4. doi:10.1007/s00500-008-0362-4.

[65] Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man Cybern.*, *15*, 116–132. URL: https://doi.org/10.1109/TSMC.1985.6313399. doi:10.1109/TSMC.1985.6313399.

[66] Xiong, N. (2011). Learning fuzzy rules for similarity assessment in case-based reasoning. *Expert Syst. Appl.*, *38*, 10780–10786. doi:10.1016/j.eswa.2011.01.151.

[67] Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.*, *11*, 712–731. URL: https://doi.org/10.1109/TEVC.2007.892759. doi:10.1109/TEVC.2007.892759.

[68] Zhao, Z., Liu, T., Li, S., Li, B., & Du, X. (2017). Ngram2vec: Learning improved word representations from ngram co-occurrence statistics. In M. Palmer, R. Hwa, & S. Riedel (Eds.), *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017* (pp. 244–253). Association for Computational Linguistics. URL: https://doi.org/10.18653/v1/d17-1023. doi:10.18653/v1/d17-1023.