

# Chapter 1

## Conclusions

The work presented here solves the problem of finding the similarity degree between two fuzzy predicates in the fuzzy logic domain. As mentioned in chapter ??, the motivation is fixing the imprecise nature of the technology that current web engines utilize. Once again informally, we introduce a framework where, when one queries a fuzzy concept such as *fast red car* the framework is able to make use of a crisp info such as the maximum speed of a particular car, and in the end returns a set of fuzzy answers. But on top of that, now the framework is able to retrieve answers regarding to similar queries like the ones for *fast orange car* or *average-speed red car*, with as expected lower credibility values. In this regard, a sound mathematical formalism has been introduced (see Chap. ??) which enabled the representation of predicates as graph trees. By that we are able to utilize an efficient search methodology as in the cases of graph problems, for detecting similar pairings of subconcepts. The last step is constructing an operational evaluation function that computes the resulting degree of similarity. (See Sec. ??)

The most recent approach in the field of interest is the *Structure Based Method(SBM)* [?]. *SBM* makes use of the structural property by constructing the predicate tree via taking the head of a fuzzy rule as the root of the tree and by branching the tree regarding the body of the rule. Here one point of concern is that, the construction of the tree is recursive but not the evaluation of the similarity degree between the predicates. As a result of this the trees need to be structurally equivalent for the algorithm to be able to execute. This hole in the algorithm is patched by introducing identity predicates when one tree is lacking internal nodes or leaf nodes compared to the other one. But as Lu also confirms [?], this causes a "distortion" effect on the final evaluation. Especially in cases such as where the branching factors differ by a big margin or when one tree is highly unbalanced, this effect would be clearly apparent.

The main idea behind this work is introducing a methodology that avoids the shortcomings of the preceding approach. These points of problems and the way we tackle with them are inspected and demonstrated in detail in their corresponding sections.

- In order to evaluate the degree of similarity between two predicates, *SBM* requires their corresponding predicate trees to have the same graphical structure. Since in most scenarios, this indeed is not going to be the case, the missing branches of the trees are expanded with *identity predicates*. Moreover for the similarity evaluation function to operate, a default similarity proximity value is defined between an arbitrary predicate and the identity predicate. Assume the cases where there is a relatively big branching factor difference between the two predicate trees of interest. In such an example as discussed earlier, the smaller tree is going to be filled in the appropriate places with identity predicates. Thus most of the similarity values of predicate pairings are going to come from the default similarity rule which is defined for the identity predicate and so the final computation will resemble that value inevitably. As mentioned in section ??, in *SBM* as the missing number of nodes in one tree increases, the similarity degree calculated by the algorithm converges to the default values that is defined between an arbitrary predicate and the identity predicate.
- ◊ The justification of our algorithm avoiding this mishap is pretty straightforward. Since it does not introduce any such external knowledge, and just use the original information of the initial knowledge base, no such shortcomings are encountered in any scenarios. (See Sec. ??)
- After every level of expansion of the predicate trees, *SBM* checks every predicate pair between the two trees with respect to their resulting similarity degree. Before the next level of expansion is pursued, a filtering is done on the tree by selecting the best pair combination on the level. The problem with this approach is that the information on a prior level is incomplete, and thus wrong steps can be taken when filtering that will cause the loss of crucial information for the main focus, *i.e.* comparing the similarity values of the main predicates. (See Sec. ??)
- ◊ Once again, as the predicate trees are maintained in their original forms, the algorithm is able to work effectively in a manner that resembles *Depth First Search*, on the complete structure emitting the need of filtering some parts of the data. (See Sec. ??)
- The way that *SBM* utilizes the credibility values of the fuzzy rules in the similarity evaluation function for the predicates, proves to be problematic when the branching factor of the predicate tree is small. As the unit distance of the credibility values is directly summed with the values of similarity pairs in the numerator of the equation, when the cardinality  $n$  is small, the values of credibilities simply overshadow the values of the similarity pairs. (See Sec. ??) For instance, in the case where there is a single subconcept pairing, the effect of the credibility values over the final result would be exactly fifty percent which is highly excessive.

- ◊ In our approach unit distance of the credibility values of the fuzzy rules is not thrown in between the similarity proximity values of the pairings. They are maintained separately, until the similarity values are processed thoroughly in the evaluation function. And only in the end utilized via the operator that is defined by the fuzzy rules themselves. (See Sec. ??)
- Yet another shortcoming of *SBM* is that the method is not able to utilize the information which comes from shared similarity concepts in similarity predicate pairings. The method tries to fix the pairings in an analogous manner of a cartesian product. In other words every subconcept is only mapped to another subconcept from the other predicate tree, thus only one of the fuzzy similarity relations where a particular subconcept appears is taken into account. (See Sec. ??)
- ◊ In very brief terms, our methodology is not bounded with such limitations as the preceding one. A particular subconcept may appear in as many fuzzy similarity rules as it may, and all of these information is realized and processed as the method is able to search and collect all of the similarity relations distinctly without one affecting the other. (See Sec. ??)
- *SBM* has no means for representing similarity relations as fuzzy rules. In other terms, all of the similarity relations that the approach contains are solely facts.
- ◊ In ??, we introduce sound syntax and semantics for representing similarity relations with credibility values. This extension enables us to investigate another topic of research, namely evaluating confidence values for computations of similarity degrees between two fuzzy predicates. (see Chap. ??)

It is only fair to also mention the shortcomings of our approach. Remember that when evaluating the final degree of similarity between two predicates, the predefined similarity relations on subconcepts are taken into account in the evaluation algorithm. Whereas in real world scenarios, one may encounter knowledge bases which do not contain complete information. In order to take into account such cases, we introduce three approximation methods concerning the confidence that we attain to the result of our similarity evaluation methodology over fuzzy predicates.

- The first approach is a naive one called the *Vertex Approach*, where the percentage of the subconcepts which appear in a similarity relation is computed with respect to the total number of subconcepts in the knowledge base. (See Sec. ??) The calculations of *Vertex Approach* prove to be highly plausible. However the problem of the method is that since it's only concerned with the cardinality of the

subconcepts participating in similarity relations, or vertices in graph terms, in the scenario where more relations between the same set of subconcepts are introduced, it is not able to utilize the newly added information.

- The intuition behind the second approach, namely *Edge Approach*, is eliminating the shortcoming of the preceding one. This time the percentage of the similarity pairings between the subconcepts is computed with respect to the total number of potential pairings in the knowledge base. Speaking in graph terms, the latter equates counting the number of added edges between the two predicate trees. Yet in this approach since the main parameter (i.e. similarity pairings) is normalized by a relatively large-scale parameter (i.e. potential pairings), the evaluated credibility values are dramatically lower than the expected ones.
- The source of motivation of the last approach is gathering beneficial characteristics of the first two approaches, whereas avoiding the shortcomings. In the light of this idea we propose the *Hybrid Approach* which merges two approaches together by attaining them some weights. For a given knowledge base and a similarity pairing as a result of the similarity evaluation algorithm, the methodology is able to compute a credibility value which is plausible and at the same time can realize all of the information presented by the knowledge base.

On top of these we introduce firstly a naive intuitionistic approach which is then followed by an atomized process of determining the weights in the credibility evaluation equation, in a similar manner how the problem is realized in [?]. In chapter ?? we realize this method by training the system with the original knowledge base, and then using the evaluated weights in order to compute the credibility values of the similarity relations. We have been highly successful with the application in the sense that we have computed values which are extremely close to their corresponding real-world numbers. Moreover, we have evaluated conservative results, in other terms values which do not exceed the original degrees of similarity. As discussed in chapter ??, this notion holds great value for maintaining *Multi-Adjoint* properties. These two features depict that our methodology holds great promise for the real-life applications.

The current direction of the research is looking into other scientific fields from where we may find a new inspiration for a better methodology. One potential candidate that stands out is the set of sound and efficient algorithms used in artificial neural networks.