Title: Fuzzy-Citation-KNN: a fuzzy nearest neighbor approach for multi-instance classification

Authors: Pedro Villar, Rosana Montes, Ana María Sánchez, Francisco Herrera

Event: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2016), Vancouver (Canada), pp.946-952, july 24-29, 2016.

DOI: 10.1109/FUZZ-IEEE.2016.7737790

Fuzzy-Citation-KNN: a fuzzy nearest neighbor approach for multi-instance classification

Pedro Villar[a], Rosana Montes[a], Ana María Sánchez[a], Francisco Herrera[b]

[a]Dept. of Software Engineering; University of Granada
18071 - Granada, Spain. e-mail: pvillarc,rosana,amlopez@ugr.es

[b]Dept. of Computer Science and A.I.; University of Granada,
18071 - Granada, Spain. e-mail: ocordon,herrera@decsai.ugr.es

Abstract

This contribution deals with multi-instance classification, where the labeled data samples are bags composed on instances instead of labeled instances as in standard classification. Every bag contains a number of traditional instances (described by a number of attributes) and the number of instances is not usually the same in all the bags. So, the whole bag is labeled but the instances that compose the bag are not individually labeled. We propose a fuzzy sets based extension of the well known algorithm called Citation-KNN, a reference method in multi-instance classification. Citation-KNN uses two types of examples in the classification rule: neighbors and citers of the bag to be classified. We analyze two versions of our proposal, one of them using both neighbors and citers, and the other one using only neighbors. Our approach uses the Hausdorff distance and it is based on the FuzzyKNN algorithm.

Several data-sets from KEEL data-set repository are used in the experimental study and we compare our proposals with the original Citation-KNN algorithm.

1 Introduction

In multi-instance classification (MIC)[1], the data examples are unordered collections (bags) of instances. The instances are similar than the examples used in standard classification. So, all the instances have the same structure. On the other hand, the number of instances can vary among bags. The main characteristic of MIC is the absence of the classification label in each instance, the class label is assigned to the bag as a whole. Therefore, the data-sets used in MIC are more complex than the used in standard classification and it is necessary to use specific learning algorithms to classify unknown bags.

There are several algorithms for dealing with MIC in the specialized literature [2] [3]. Many of them are based on algorithms proposed for standard classification. One example is the Citation-KNN algorithm [4], a MIC learning method based on the well known k-nearest neighbor classification algorithm. Citation-KNN uses in its classification rule the "citers" of the bag to be classified, apart from the neighbors of that bag. Citers are the bags that include the bag to be classified in their neighboring.

When applying a k-nearest neighbor approach, a way to measure the distance between two examples is needed. In the context of MIC, an example is a bag that contains multiple instances. Therefore, it is necessary to characterize how the distance between two sets of instances could be measured. We will employ the same metric used in the description of Citation-KNN [4]: the Hausdorff distance.

Fuzzy sets theory have been widely used in many traditional (single-instance) machine learning applications, specially in classification tasks, but so far not in MIC. In a recent contribution [5], a framework for multi-instance classifiers based on fuzzy set theory is presented. From the best of our knowledge, there are no proposals of fuzzy nearest neighbor approaches adapted to MIC. In [6], a fuzzy-based adaptation of the K-Nearest Neighbor algorithm is proposed (FuzzyKNN). This method is considered the major reference in the combination of fuzzy logic and K-nearest neighbor. A recent published review [7] in fuzzy nearest neighbors algorithms, performs a comparative study among many proposals and FuzzyKNN obtains very accurate results.

The main purpose of this contribution is to extend the Citation-KNN algorithm, proposing a fuzzy-based adaptation of that method. We will use in our learning method some ideas used for designing the FuzzyKNN algorithm. Our proposal includes a preliminary phase in which membership values for each class are calculated for every example of the training data-set. For classifying a new example, a membership degree for each class is computed, considering the membership values obtained in the first phase weighted by the distance to the new example. The class with greater membership degree is assigned. We will analyze two variants depending on the examples used for calculating the class membership degree: one of them considers both neighbors and citers of the new example, and the other one only considers the neighbors.

In order to illustrate the good performance of the proposed fuzzy citation-KNN algorithm, we will compare the obtained results among the original Citation-KNN method and the two versions of our proposal: with and without considering the citers in the classification rule.

We have selected a collection of binary-class MIC data-sets from KEEL data-set repository [8] for developing our experimental analysis. Furthermore, we will perform a statistical analysis using non-parametric tests [9] [10] [11] to find significant differences among the obtained results.

This paper is organized as follows. First, Section 2 introduces the preliminary concepts used in this paper: multi-instance classification, the distance metric used (Hausdorff distance) and the Citation-KNN algorithm. Next, in Section 3 we will describe our proposal, a fuzzy-based approach of the Citation-KNN algorithm. The next section describes the experimental study. Finally, in Section 5, some conclusions will be pointed out.

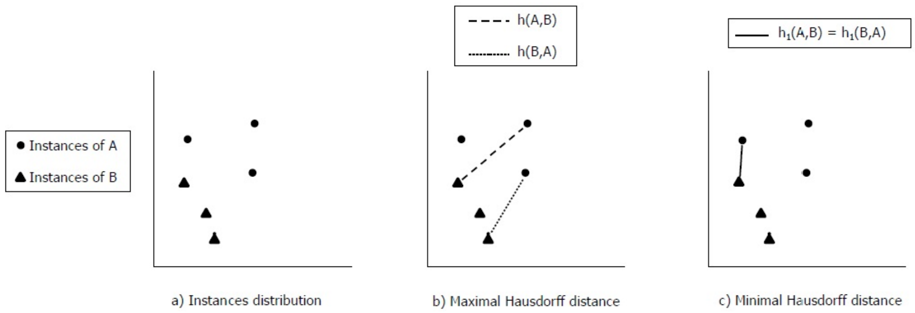

Figure 1: Hausdorff distance in a two-dimensional instance space

2 Preliminaries

This section introduces the main concepts of multi-instance classification, defines the Hausdorff distance and describes the algorithm used for design our proposal: Citation-KNN.

2.1 Multi-instance classification

Multi-instance classification(MIC) was originally described in [1] and it has received much attention since that seminal paper. There are several application areas where MIC have been used. The drug activity was the first problem trying to be solved by MIC [1]. Other examples are image classification [12], content-based image retrieval [13] [14], web index page recomendation [15], robot control [16] and bioinformatics [17].

A multi-instance data-set for a MIC problem has the following structure:

$$T = \{(X_1, l_1), (X_2, l_2), \ldots, (X_n, l_n)\}$$

Where $X_i \in 2^\chi$ are labeled bags, that is, composed of non-labeled instances whose universe of discourse is χ, which corresponds to the feature set describing the instances. The value l_i is the class associated to the bag X_i and it is assumed to be drawn from a finite set L. A multi-instance classifier Γ try to predict the label of a new unlabeled bag, that is:

$$\Gamma : 2^\chi \to L$$

There are several ways to group the MIC algorithms. Usual taxonomies are based on the information level used to classify new bags. In [2], the multi-instance classifiers are divided into three groups. One of them is called the instance space paradigm, where the classification task rely on the instance-level information. The other two groups use the bag-level information for the classification task: the bag-space paradigm and the embedded space paradigm. A brief description of each paradigm is presented next:

- *Instance space paradigm*: In this paradigm, a instance-level classifier is build to discriminate the instances in positive bags from those in negative ones. The final bag-level classifier is obtained by aggregating instance-level scores. The learning process considers the characteristic of individual instances, not the characteristics of the whole bag. Methods of this paradigm are Axis-Paralel Rectangle [1], Diverse Density [18] and miSVM [19].

- *Bag space paradigm*: In this case, the classification of a new bag rely on the information provided for the whole bag, not for the individual instances. Algorithms that follow this paradigm are Citation-KNN [4] and MI-Graph [20].

- *Embedded space paradigm*: this paradigm performs a mapping from the bag space to a single vector space. Next, a traditional single-instance classifier is trained. Examples are SimpleMI [21], YARDS [22], DD-SVM [23], MILES [24], GMIL [25] and BARTMIP [26].

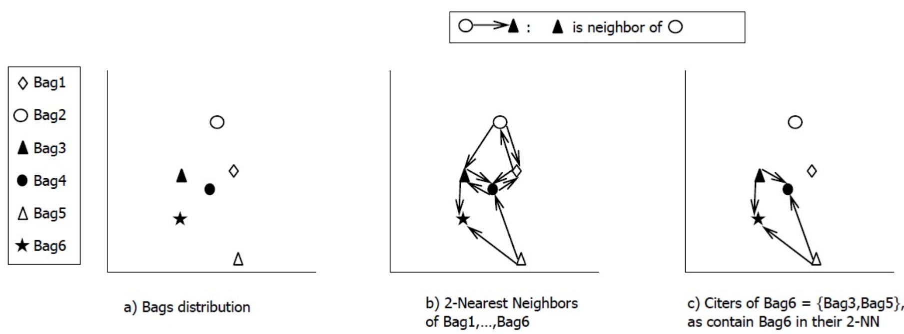

Figure 2: 2-nearest citers of bag 6

2.2 Distance metric

All methods that follow the nearest neighbor approach consider a subgroup of nearest elements in the classification process. Therefore, it is necessary to choose a distance value between two bags in a multi-instance framework. However, a bag is an unordered set of instances and two bags can contain different number of instances. There are some metrics proposed in the MIC literature: the Hausdorff distance (used in Citation-KNN algorithm) [4], the Earth Movers Distance (EMD) [27], the Chamfer distance [28], etc.

Given two set of instances $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_m\}$, the Hausdorff distance ($H(A,B)$) is defined as [4]:

$$H(A,B) = max\{h(A,B), h(B,A)\}$$

where

$$h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

The Hausdorff distance is very sensitive to a single outlying point of A or B. To increase the robustness with respect to noise, a possible modification to the Hausdorff distance is to take the k-th ranked distance rather than the largest ranked one:

$$h_k(A,B) = \underset{a \in A}{kth} \min_{b \in B} \|a - b\|$$

When $k = m$, the distance is the same as $h(A,B)$ defined above and it is called the maximal Hausdorff distance. When $k = 1$, the minimal one of the m distances determines the value of the distance:

$$h_1(A,B) = \min_{a \in A} \min_{b \in B} \|a - b\| = \min_{b \in B} \min_{a \in A} \|b - a\| = h_1(B,A)$$

In this case $H(A,B) = H(B,A) = h_1(A,B) = h_1(B,A)$. We will use this minimal Hausdorff distance due to its better behavior than the original formulation (maximal Hausdorff distance), as it is shown in the experiments developed in [4]. A graphical example of the Hausdorff distance in a two-dimensional instance space is shown in Figure 1.

2.3 Citation-KNN algorithm

This algorithm [4] was designed specifically for MIC and follows the nearest neighbor approach. For classifying a new bag b, Citation-KNN not only considers the neighbors of b (called references in [4]), but also considers the bags that count b as a neighbor (called citers). Therefore, the K-nearest references of b are the K-nearest neighbors of b. For setting up the C-nearest citers, the C-nearest neighbors of all bags are located. Next, every bag that includes b in his C-nearest neighboring is a citer of b. We must note that, given a C value, the number of citers of a bag cannot be determined a priori. For example, given a data-set with five bags $\{b_1, b_2, b_3, b_4, b_5\}$, their nearest neighbors are shown in Table 1. Therefore, if $K = 3$, the K-nearest neighbors of b_3 are $\{b_4, b_6, b_1\}$. On the other hand, if $C = 2$, the C-nearest citers of b_4 is the set $\{b_1, b_3, b_5, b_6\}$, the C-nearest citers of b_2 is the set $\{b_1\}$, the C-nearest citers of b_6 is the set $\{b_3, b_5\}$ and there are no C-nearest citers for b_5. Figure 2 shows graphically the process of calculating the 2-nearest citers of one bag. In the figure, a two dimensional distribution of bags is considered and the euclidean distance between two bags is equivalent to the minimal Hausdorff distance of these two bags, following the relationship shown in Table 1. In the original description of Citation-KNN [4], the value C was empirically set to $K + 2$, reflecting that citers seem to be more important than neighbors.

Therefore, Citation-KNN needs to give a distance metric and two parameters. The minimal Hausdorff distance is proposed in the definition paper. One

Table 1: Nearest neighbors of six bags

	$K=1$	$K=2$	$K=3$	$K=4$	$K=5$
b_1	b_4	b_2	b_3	b_6	b_5
b_2	b_1	b_3	b_4	b_6	b_5
b_3	b_4	b_6	b_1	b_2	b_5
b_4	b_1	b_3	b_6	b_2	b_5
b_5	b_6	b_4	b_1	b_3	b_2
b_6	b_4	b_3	b_5	b_1	b_2

$$\mu_c(b) = \frac{\sum_{i=1}^{K} \mu_c(x_i)(1/\|b-x_i\|^{2/(m-1)}) + \sum_{j=1}^{n_c(b)} \mu_c(x_j)(1/\|b-x_j\|^{2/(m-1)})}{\sum_{i=1}^{K}(1/\|b-x_i\|^{2/(m-1)}) + \sum_{j=1}^{n_c(b)} (1/\|b-x_j\|^{2/(m-1)})} \quad (1)$$

of the parameters is the number of references (or neighbors), the parameter K, and the other one is the parameter C, that determines the set of citers. Considering binary-class MIC problems, four values are calculated for deriving the class label of a new unseen bag b. Being $Ne(b)$ the set of K-nearest neighbors of b_k and $Ci(b)$ the set of C-nearest citers of b, the values are:

- K_p: Number of positive bags in $Ne(b)$
- K_n: Number of negative bags in $Ne(b)$
- C_p: Number of positive bags in $Ci(b)$
- C_n: Number of negative bags in $Ci(b)$

Once these values have been calculated, the classification rule is:

if $(K_p + C_p > K_n + C_n)$ **then**
    $class = positive$
else
    $class = negative$
end if
$return(class)$

Obviously, $K_p + K_n = K$, but the total number of citers $(C_p + C_n)$ is not known a priori. Therefore, whether K is odd or even and wether C is odd or even, the sum $K_p + C_p + K_n + C_n$ can be an even number. So, a tie between the number of positive bags and negative bags is possible. In the original Citation-KNN algorithm [4], the tie is always solved assigning the negative class to the bag (as in the classification rule shown before). This decision was caused by the two data-sets used for testing the algorithm in the paper (musk1 and musk2). In the musk1 data-set, the authors found some contradictory cases of negative bags with majority of positive bags in their neighbors. We think that decision can be justified if there is some bonus information about a concrete data-set as in the previous case. When using Citation-KNN as a general purpose MIC learning algorithm, we think that the possible ties must be resolved by means of a more generic rule, as we will explain in the experimental section.

3 Fuzzy-based approach of Citation-KNN algorithm

In this section we present our proposal, a new fuzzy-based algorithm for MIC that follows the structure of Citation-KNN. The method is composed of two stages:

- A preliminary phase where class membership are derived, obtaining a value in $[0, 1]$ for each instance and class of the training data-set. In [6] three possibilities for computing these membership values was proposed, being the 'crisp' option one of them. The best performing method needs, for each instance x_i of the training data-set, to compute its k_{init} nearest neighbors. Then, the membership values are assigned according the following function:

$$\mu_c(x_i) = \begin{cases} 0.51 + (v_c/k_{init}) \times 0.49 & \text{if } c = \omega \\ (v_c/k_{init}) \times 0.49 & \text{otherwise} \end{cases}$$

  where v_c is the number of neighbors belonging to class c and ω is the original class label of x_i. The parameter k_{init} is usually set to a value between $\{3, \ldots, 10\}$ [7]. The sum of all the membership values will always be 1. The effect of eq.3 is that bags close to the center of the class distribution keep their original crisp membership values (1.0 to the original class and 0.0 to the rest of classes). However, bags close to the boundaries among classes divide part of their membership values among the nearest classes. We must note that the coefficients ensure that the largest membership value will be assigned to the ω class, regardless of the neighboring bags.

- Classification rule. For a new unseen bag b, the K-nearest neighbors and the C-nearest citers are computed as well as described for the Citation-KNN algorithm. Then, a membership degree of b in each class is calculated, where each neighbor of b and each citer of b votes for each class using their membership values previously computed. These votes are weighted over the inverse of the distance to b. Finally, all votes are added according the equation used (equation 1). In that equation, $n_c(b)$ is the number of citers of bag b. It can be seen that the neighbors of b and the citers of b contributes equally to the calculus. The parameter m determines how heavily the distance is weighted when calculating each neighbor or citer contribution to the membership value. As m increases, the relative distances from b have less effect. As m approaches to one, the closer neighbors contribute much more than those farther away.

Finally, the class with the greatest combined vote is assigned to b. In the unlikely case of a tie between combined votes, the class of the nearest neighbor to b is the final prediction. An advantage of using this fuzzy scheme is the level of assurance in the classification, provided for the class membership values [6].

Therefore, considering a binary-class MIC problem, the classification rule is:

```
if (μ_p(b) > μ_n(b)) then
    class = positive
else
    if (μ_p(b) < μ_n(b)) then
        class = negative
    else
        class = 1NN(b)
    end if
end if
return(class)
```

Where $\mu_p(b)$ is the membership value obtained for the positive class, $\mu_n(b)$ is the membership value of the negative class and $1NN(b)$ is the class of the nearest neighbor of b. We denote this proposal as FuzzyCitation-KNN, since it is a fuzzy-based extension of the Citation-KNN algorithm.

We are also interested in analyze the behavior of our proposal in a more classical nearest neighbor way, in that the classification rule ignores the citers and only uses the neighbors. In this case, the classification rule remains equal than FuzzyCitation-KNN but the calculus of the membership degree of b in each class is modified. So, the equation of figure 3 is simplified as it only considers the contribution of the neighbors:

$$\mu_c(b) = \frac{\displaystyle\sum_{i=1}^{K} \mu_c(x_i)(1/\|b - x_i\|^{2/(m-1)})}{\displaystyle\sum_{i=1}^{K}(1/\|b - x_i\|^{2/(m-1)})}$$

Therefore, this last approach can be considered a MIC adaptation of the FuzzyKNN method [6] and we have denoted this algorithm as FuzzyKNN-MIC.

4 Experimental Study

In this section, we will first provide details of the binary class multi-instance problems chosen for the experimentation (subsection 4.1). Next, we will introduce the methods selected for comparison and a brief discussion over the parameters setting (subsection 4.2). Then, we will describe the statistical tests applied to compare the obtained results along the experimental study (subsection 4.3). Finally, we show the results obtained for all the methods and the statistical analysis (subsection 4.4).

4.1 Data-sets

We have used ten binary class multi-instance data-sets from KEEL data-set repository [8][1]. Table 2 summarizes the characteristics of these data-sets: number of bags, number of positive bags, number of negative bags and number of attributes. There are different imbalance ratios, from totally balanced data-sets to data-sets with a certain degree of imbalanced, like the three mutagenesis data-sets.

Table 2: Summary description of data-sets.

Data-set	**#Bags.**	**#Pos.**	**#Neg**	**#Atts.**
eastwest	20	10	10	26
elephant	200	100	100	232
fox	200	100	100	232
musk1	92	45	47	168
musk2	101	62	39	168
mutagenesis-atoms	188	63	125	12
mutagenesis-bonds	188	63	125	12
mutagenesis-chains	188	63	125	26
tiger	200	100	100	232
westeast	20	10	10	26

4.2 Algorithms of comparison and parameters

We will compare the performance of our two proposals (FuzzyCitation-KNN and FuzzyKNN-MIC) and the method in which they are based, the original Citation-KNN method [4]. We have modified the original rule of Citation-KNN for the case of a tie, as the reason for the original rule was determined for a single data-set. If the sum of neighbors and citers belonging to the positive class is equal to the sum of neighbors and citers of the negative class, the final prediction is the label class of the nearest neighbor.

We will use the leave-one-out scheme in the experiments. Therefore, the training data-set used for calculating the initial membership values in the two fuzzy methods (FuzzyCitation-KNN and FuzzyKNN-MIC) is the whole data-set except the bag to be classified.

For all the algorithms, we have used the minimal Hausdorff distance as metric for determining the nearest neighbors. The range values for the K parameter

[1] `http://www.keel.es/dataset.php`

Table 5: Prediction accuracy for different values of K

Algorithm	$K=0$	$K=1$	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$	$K=7$
Citation-KNN	68.41	68.41	68.41	69.04	68.73	**69.11**	68.87	69.03
FuzzyKNN-MIC	–	68.41	68.71	68.81	69.02	68.97	68.56	**69.76**
FuzzyCitation-KNN	67.92	69.79	69.41	70.92	71.51	70.74	71.51	**71.60**

is the set $\{1, 2, 3, 4, 5, 6, 7\}$. It is not usual to choose even numbers for the K parameter in the K-nearest neighbor based algorithms when dealing with binary-class problems. The reason is to avoid the possible ties between positive and negative neighbors. In our proposals (both fuzzy methods), the use of membership values of the training data-set weighted by the distance, reduces hugely the possibility of a tie in the two final class membership values. In the Citation-KNN algorithm, as mentioned in Section 2.3, there is the same probability of finding a tie if K is an odd or an even number. Therefore, there are no reasons for removing even values of K in our experiments.

Considering the two algorithms that use the citers of a bag (Citation-KNN and FuzzyCitation-KNN), the value C is set to $K+2$ in all cases, as it is recommended in [4]. For these two algorithms, we have included in the experiments the possibility of $K = 0$ and so, $C = 2$. In this case, only the citers of the bag to be classified contribute for the calculus of the membership values. The option of $K = 0$ was also included in the experiments developed in [4].

Our two fuzzy proposals also require values for two parameters used in the equation shown in figure 3. The parameter m, that regulates the distance weighting and the k_{init} parameter, that determines the number of neighbors used in the calculus of the membership values of the first phase of both algorithms. The parameter m is set to 2, the same value used in the FuzzyKNN definition paper [6]. The range of recommended values for parameter k_{init} in [7] is the set $\{3, \dots, 10\}$. Preliminary proofs shown that both proposed methods present better results with lower values of k_{init}, as well as the limited influence of this parameter in the algorithms behavior. We have run our two proposals (with all the possible values for parameter K), considering three values for k_{init} ($\{3, 4, 5\}$). The mean results in accuracy of these experiments are shown in Table 3. It can be observed that the differences are tight and the best mean accuracy were obtained with $k_{init} = 3$ in both methods. Therefore, we suggest $k_{init} = 3$. Only the obtained results with this value are shown in the tables of results of the next subsection. Table 4 summarizes the parameters chosen for each method.

Table 3: Mean of prediction accuracy with $K = \{1, \dots, 7\}$ for different values of k_{init}

Algorithm	$k_{init} = 3$	$k_{init} = 4$	$k_{init} = 5$
FuzzyKNN-MIC	**68.91**	68.60	68.24
FuzzyCitation-KNN	**70.89**	70.83	70.11

4.3 Statistical tests for performance comparison

In order to carry out the comparison of the classifiers appropriately, non-parametric tests should be considered, according to the recommendations made in [9][10]. In this contribution, we consider the Friedman Aligned test for both computing the ranking of the algorithms according to its performance and the p-value that determines significant differences among the results. Then, we proceed with a Holm non-parametric statistical procedure for $1 \cdot n$ comparisons. Therefore, the adjusted p-values (APV) associated with each comparison are obtained, which represent the lowest level of significance of a hypothesis that results in a rejection. Any interested reader can find additional information on the thematic website `http://sci2s.ugr.es/sicidm/`, where software for the application of the statistical tests is provided.

Table 4: Parameter setting

<table>
<tr><th>Algorithm</th><th>K</th><th>C</th><th>m</th><th>k_{init}</th></tr>
<tr><td>Citation-KNN</td><td rowspan="2">$\{0, \ldots, 7\}$</td><td rowspan="2">K+2</td><td>–</td><td>–</td></tr>
<tr><td>FuzzyCitation-KNN</td><td rowspan="2">2</td><td rowspan="2">3</td></tr>
<tr><td>FuzzyKNN-MIC</td><td>$\{1, \ldots, 7\}$</td><td>–</td></tr>
</table>

4.4 Experimental Analysis

Table 5 shows the mean of the prediction accuracy (in %) obtained for the three algorithms for each value of parameter K. The best value for each algorithm is stressed in boldface. As it can be observed, FuzzyCitation-KNN obtains the best accuracy for all possible values of K (except for the case $K = 0$ respect to Citation-KNN). Besides, there is no high variations depending on the value of K in the three algorithms. Thus, the fuzzy-based adaptation of Citation-KNN results in a more accurate way to deal with MIC.

Next, we compare the individual results for each data-set considering the best value of K for each algorithm in Table 6. FuzzyCitation-KNN obtains better accuracy than the other algorithms in the majority of data-sets.

In order to validate these results, we show the results of the statistical analysis performed (described in subsection 4.3). The p-value computed by Friedman Aligned test is 0.02383, implying that there are significant differences among the algorithms compared in this analysis. Therefore, we carry out a post-hoc test (Holm) in order to determine whether the control method (FuzzyCitation-KNN) outperforms the other algorithms. Table 7 shows the results obtained by applying post hoc methods over the results of Friedman Aligned procedure. The obtained adjusted p-values (APV) show significative differences between the best method (FuzzyCitation-KNN) and the other two methods used for comparison.

Table 6: Data-sets accuracy considering the K-value that obtains better accuracy average

Data-set	**FKNN-MIC**	**Cit-KNN**	**FCit-KNN**
	K=7	K=5	K=7
eastwest	50.0	55.0	50.0
elephant	80.5	79.0	83.5
fox	61.0	61.0	61.5
musk1	76.1	78.3	82.6
musk2	72.3	74.3	79.2
mutagenesis-atoms	73.9	75.0	74.5
mutagenesis-bonds	77.1	76.6	76.1
mutagenesis-chains	78.2	74.5	78.2
tiger	78.5	77.5	80.5
westeast	50.0	40.0	50.0
Mean	69.76	69.11	**71.60**

Table 7: Average results with standard deviation average per data-set, Ranks (Friedman Aligned test) and APVs (Holm test). Control method is pointed out with asterisks.

Algorithm	Acc. average	Ranking	APV
Citation-KNN	69.11	19.35 (3)	0.031643
FuzzyKNN-MIC	69.76	17.3 (2)	0.058451
FuzzyCitationKNN	**71.60**	9.85 (1)	*******

5 Conclusions

This contribution proposes a new fuzzy approach for multi-instance classification based on the Citation-KNN algorithm, a reference method in the multi-instance community. Our proposal performs a fuzzy adaptation similar to the proposed in the FuzzyKNN method, a successful fuzzy adaptation of the general K-NN classifier. We have analyzed two options of our proposal, one of them considering neighbors and citers in the classification rule (named FuzzyCitation-KNN) and the other one considering only the neighbors in the classification rule (denoted FuzzyKNN-MIC). FuzzyCitation-KNN outperforms significantly the other two methods and it presents a regular behavior when varying the value of parameter K.

In future works, we will extend this study incorporating new multi-instance data-sets and testing the performance of the two algorithms proposed with other distance metrics.

Acknowledgment

This work was partially supported by the Spanish Ministry of Science and Technology under project TIN2014-57251-P and the Andalusian regional project P11-TIC-7765.

References

[1] T. Dietterich, R. Lathrop, and T. Lozano-Perez, *Solving the multiple instance problem with axis-parallel rectangles*, Artificial intelligence 89(1): 31-71, 1997.

[2] J. Amores, *Multiple instance classification: Review, taxonomy and comparative study*, Artificial Intelligence 201: 81-105, 2013.

[3] E. Alpaydin, V. Cheplygina, M. Loog, and D. M. Tax, *Single-vs. multiple-instance classification*, Pattern Recognition 48(9): 2831-2838, 2015.

[4] J. Wang and J. Zucker, *Solving multiple-instance problem: A lazy learning approach*, Proceedings of the 17th international conference on machine learning: 1119-1125, 2000.

[5] S. Vluymans, D. Sánchez, Y. Saeys, C. Cornelis and F. Herrera, *Fuzzy Multi-Instance Classifiers*, to appear in IEEE Transactions on Fuzzy Systems, 2016. DOI:10.1109/TFUZZ.2016.2516582

[6] J. Keller, M. Gray, and J. Givens, *A fuzzy k-nearest neighbor algorithm*, IEEE Transactions on Systems, Man and Cybernetics 4: 580-585, 1985.

[7] J. Derrac, S. García and F. Herrera, *Fuzzy nearest neighbor algorithms: Taxonomy, experimental analysis and porspects*, Information Sciences 260: 98–119, 2014.

[8] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, *KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework*, Journal of Multi-Valued Logic and Soft Computing, 17(2-3): 255–287, 2011.

[9] J. Demšar, *Statistical comparisons of classifiers over multiple data sets*, Journal of Machine Learning Research 7: 1–30, 2006.

[10] S. García and F. Herrera, *An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons*, Journal of Machine Learning Research 9: 2677–2694, 2008.

[11] S. García, A. Fernández, J. Luengo, and F. Herrera, *A study of statistical techniques and performance measures for genetics–based machine learning: Accuracy and interpretability*, Soft Computing 13(10): 959–977, 2009.

[12] O. Maron and A. L. Ratan, *Multiple-instance learning for natural scene classification*, Proceedings of the 15th International Conference on Machine Learning: 341-349, 1998.

[13] C. Yang and T. Lozano-Perez, *Image database retrieval with multiple-instance learning techniques*, Proceedings of the 16th International Conference on Data Engineering: 233-243, 2000.

[14] Z. Zhou, M.-L. Zhang, and K.-J. Chen, *A novel bag generator for image database retrieval with multi-instance learning techniques*, Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence: 565-569, 2003.

[15] Z. Zhou, K. Jiang, and M. Li, *Multi-instance learning based web mining*, Applied Intelligence 22(2): 135-147, 2005.

[16] S.A. Goldman and S.D. Scott, *Multiple-instance learning of real-valued geometric patterns*, Annals of Mathematics and Artificial Intelligence 39(3): 259-290, 2003.

[17] Y. Zhang, Y. Chen, and X. Ji, *Motif Discovery as a Multiple-Instance Problem*, Proceedings of the 18th International Conference on Tools with Artificial Intelligence: 805-809, 2006.

[18] O. Maron and T. Lozano-Perez, *A framework for multiple-instance learning*, Advances in neural information processing systems 10: 570-576, 1998.

[19] S. Andrews, I. Tsochantaridis, and T. Hofmann, *Support vector machines for multiple-instance learning*, Advances in neural information processing systems: 561-568, 2002.

[20] Z. Zhou, Y. Sun, and Y. Li, *Multi-instance learning by treating instances as non-iid samples*, Proceedings of the 26th annual international conference on machine learning. ACM: 12491256, 2009.

[21] L. Dong, *A comparison of multi-instance learning algorithms*, Ph.D. dissertation, The University of Waikato, 2006.

[22] J. Foulds, *Learning instance weights in multi-instance learning*, Ph.D. dissertation, The University of Waikato, 2008.

[23] Y. Chen and J. Wang, *Image categorization by learning and reasoning with regions*, The Journal of Machine Learning Research 5: 913-939, 2004.

[24] Y. Chen, J. Bi, and J. Wang, *MILES: Multiple-instance learning via embedded instance selection*, IEEE Transactions on Pattern Analysis and Machine Intelligence 28(12): 1931-1947, 2006.

[25] S. Scott, J. Zhang, and J. Brown, *On generalized multiple-instance learning*, International Journal of Computational Intelligence and Applications 5(1): 2-35, 2005.

[26] M. Zhang and Z. Zhou, *Multi-instance clustering with applications to multi-instance prediction*, Applied Intelligence 31(1): 47-68, 2009.

[27] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid, *Local features and kernels for classification of texture and object categories:A comprehensive study*, International Journal of Computer Vision 73(2): 213–238, 2007.

[28] S. Belongie, J. Malik and J. Puzicha, *Shape matching and object recognition using shape contexts*, IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4): 509–522, 2002.