

Transactions on
Case-Based Reasoning
Vol. 4, No 1 (2011) 3-17
©2011, ibai-publishing,
ISSN: 1867-366X ,
ISBN: 978-3-942952-09-5,

ibai Publishing

www.ibai-publishing.org

The Problem of Normalization and a Normalized Similarity Measure by Online Data

Anja Attig and Petra Perner

Institute of Computer Vision and Applied Computer Sciences, IBaI
Kohlenstr. 2, 04107 Leipzig, Germany
pperner@ibai-institut.de, www.ibai-institut.de

Abstract. Case-based reasoning, image or data retrieval is based on similarity determination between the actual case and the cases in a database. It is preferable to normalize the similarity values between 0 and 1 in order to be able to compare different similarity values based on a scale. Similarity is thus imparted with a semantic meaning. The main problem arises when the case base is not yet complete and contains only a small number of cases while the other cases are collected incrementally as soon as they arrive in the system. In this case the upper and lower bounds of the feature values cannot be estimated close to the real values. This paper concerns possible methods for predicting the upper and lower bounds of a feature value and the problems that arise when these values are not correctly estimated due to a limited number of samples or a parameter distribution that is not available a-priori. The aim is to develop a method for learning the upper and the lower bounds of a feature value and to develop a methodology for dealing with change in semantic meaning of the similarity.

1 Introduction

Case-based reasoning, image or data retrieval is based on similarity determination between the actual case and the cases in a database. It is preferable to normalize the similarity values between 0 and 1 in order to be able to compare different similarity values based on a scale. A scale between 0 and 1 gives us a symbolic understanding of the meaning of the similarity value. The value of 0 indicates identity of the two cases while the value of 1 indicates the cases are unequal. On the scale of 0 and 1

the value of 0.5 means neutral and values between 0.5 and 0 mean more similarity and values between 0.5 and 1 mean more dissimilarity

Different normalization procedures are known. The most popular one is the normalization to the upper and lower bounds of a feature value.

The main problem arises when the case base is not yet completely filled and contains only a small number of cases while the other cases are collected incrementally as soon as they arrive in the system. In this case the upper and lower bounds of a feature value $[x_{\min,i,k}, x_{\max,i,k}]$ can only be judged based on this limited set of cases at the point in time t_k and must not meet the true values of $x_{\min,i}$ and $x_{\max,i}$ of feature i . Then the scale of similarity might change over time periods t_k which will lead to different decisions for two cases at the points in time t_k and t_{k+1} .

This paper describes the problem of normalization in an incremental case-based learning system.

This paper deals with possible methods for predicting the upper and lower bounds of a feature value and the problems that arise when these values are not correctly estimated due to a limited number of samples or information that is not available a-priori.

The aim is to develop a method for learning $x_{\min,i}$ and $x_{\max,i}$ of a feature i and a methodology for dealing with change in the semantic meaning of *Sim*.

In Section 2, we describe related work while in Section 3 we describe the normalization problem based on an upper and a lower value for the feature. The experimental set-up is described in Section 4 and the results are presented in Section 5. The results are discussed in Section 6 and finally conclusions are set forth in Section 7.

2 Related Work

The determination of similarity between cases is one of the key concepts in CBR [1]. Several studies on similarity have been done for this purposes. Learning of similarity measures has been introduced by Richter [2] as an outcome of the PATDEX project, where the weights and the features are updated based on the feedback of the user. Xiong and Funk studied the building of similarity by using learning strategies such as relevance feedback [3] and memetic algorithm [4]. Strategies for feature weighting to learn the similarity have been introduced by Wettschereck and Aha [5], Little et. al [6], Ahn et. al [7] and Bichindaritz [8][9]. Jarmulak et. al [8] used genetic algorithm to update the feature weights. Learning similarity function by qualitative feedback has been studied by Cheng and Hullermeier [9]. Transforming similarity into a linear space of feature combination has been studied by L. Bobrowski, M. Topczewska [12]. Graph similarity and learning of graph similarity has been studied by Perner [13].

In all these studies plays normalization not a central role. The parameters for normalization and the kind of normalization are just given a-priori.

However, studies in data mining show that normalization has a big impact on the final results [14, 15].

Normalization allows comparing different experiments based on the same application independent from the scale of the features. Normalization is a necessary step in many applications. Well known normalization procedures are the min/max normalization, the z-transformation, the log transformation, and the rank transformation [15, 16].

For the min/max normalization we need to know the upper and lower bounds of the feature value. In case of a technical system the values might be judged by looking at the systems characteristic [16, 17] or by calculating the system transfer function. Other applications rely on the perceptual observation for the application [18]. However, in most of today's applications this is not applicable, in particular when the data arrive in temporal sequence.

The other transformations require knowledge of the probability distribution of the application. If a large enough set of data is available, the probability distribution can be estimated from the data. This is the parametric method.

The situation is tricky if we are dealing with online applications where we want to start with a small set of data that increases over time and we need to develop incremental methods for normalization.

In our study we investigated what the problem with normalization is in CBR. A study in that direction has not been done before in CBR although the method of normalization and the method of parameter setting has a big impact on the performance of the CBR system. With our study we like to direct the intention of the researchers to this problem and give a first idea how to deal with it.

3 The Problem of Normalization and Similarity

Suppose we have a data set of n-features. Each feature of the data set obeys a distribution (see Fig.1), this can be any distribution. In case of Fig. 1 we chose the normal distribution.

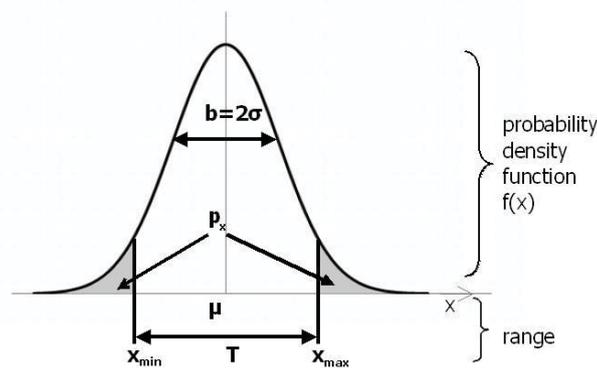


Fig. 1. Distribution of one feature and parameters of the distribution

We can define a tolerance range T_i for each feature i that gives us an upper bound $x_{\max,i}$ and a lower bound $x_{\min,i}$ for the values of the feature i . The tolerance range should be determined in such a way that the number of samples that are outside the tolerance range is low. The error rate $p_{x,i}$ is the number of non-classified case $N_{non,i}$ to all cases N :

$$p_{x,i} = \frac{N_{non,i}}{N} \quad (1)$$

The error rate of the feature i is equal to the marked area under the probability density function in Figure 1:

$$p_{x,i} = P(x_i \notin T_i) \quad (2)$$

$$p_{x,i} = P\left(\left[x_i < T_{\min,i}\right] \cup \left[x_i > T_{\max,i}\right]\right) \quad (3)$$

which is

$$p_{x,i} = P(x_i < T_{\min,i}) + P(x_i > T_{\max,i}) \quad (4)$$

The error rate should decrease the larger T_i becomes (see Figure 2). However, T_i should not be chosen too large. Feature values that do not appear frequently should not be taken into account and should be considered an outlier.

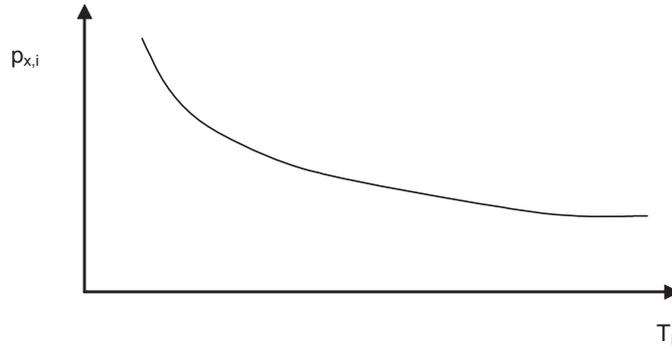


Fig. 2. Error rate $p_{x,i}$ in relation to tolerance range T_i

We can observe the error rate and adjust T_i when the error rate exceeds a predefined threshold as the classifier has evaluated k samples.

Another important parameter that has to be observed is the similarity value.

The similarity between two cases x and y is calculated as follows:

$$Sim(x, y) = \frac{1}{n} \left(\sum_i (x'_i - y'_i)^j \right)^{\frac{1}{j}} \quad (5)$$

with $j = 1$ the city-block-metric, x' and y' the normalized feature of x and y .

Let y be cases that are not classified and x^l the classified cases. The classification rule based on all features for a similarity-based classification can be:

$$Sim = \min_{\forall x^l} Sim(y, x^l) \quad (6)$$

in case of the nearest-neighbor rule and

$$Sim(y, x^l) \leq K \quad (7)$$

in case of the semantic similarity rule with K the threshold for similarity.

The error rate of the classifier is p_c that is defined as the number of misclassified case N_{mis} to the number of all cases N

$$p_c = \frac{N_{mis}}{N} \quad (8)$$

If the similarity value is not between 0 and 1 (see Figure 3), we normalize the similarity measure by the upper and the lower bounds of the feature value.

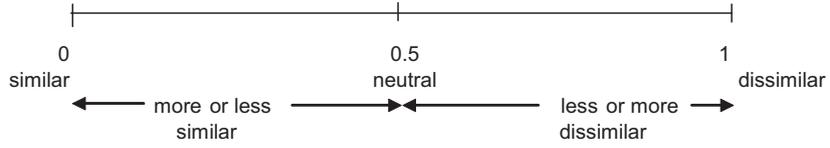


Fig. 3. Similarity values, scale and semantic meaning

This gives us a semantic understanding of similarity. Identity of two cases results in the similarity value of 0 and dissimilarity of two cases results in the similarity value of 1. Between the values 0 and 0.5 the cases are more or less similar; between 0.5 and 1 the cases are more or less dissimilar (see Fig. 3).

The difference in similarity

$$\Delta Sim = Sim_{x,y} - Sim_{x,z} \quad (9)$$

allows us to judge the similarity relation between different cases x , y and z , and should reflect the real distance between the cases and, as such, provide us with a semantic understanding of the similarity.

The difference between two cases x and y can be observed over time. The change of the similarity between times t_k and t_l can be calculated:

$$\Delta Sim(x, y, t_k, t_l) = \left| Sim_{t_k}(x, y) - Sim_{t_l}(x, y) \right| \quad (10)$$

For the points in time t_k and t_l , we can calculate the mean change of similarity and the maximum change of similarity for the cases of a data set. Depending on the outcome, further actions have to be taken.

If the values of $x_{\min,i}$ and $x_{\max,i}$ are chosen too low or too large and do not come close to the real values, then the scale behind Sim may not be correct and misclassification may occur. In the worst case, two cases that are similar may be judged dissimilar since the value of Sim is over 0.5.

If we know the probability density distribution of a feature i ; we can judge the parameters $x_{\min,i}$ and $x_{\max,i}$ according to Dreyer and Sauer [19].

If we do not know a-priori the distribution and if the data are collected incrementally, then we cannot judge these parameters in advance. The normalization and the scale for similarity will become a problem. We will set forth in the following Section the problems that arise when we normalize to $x_{\min,i}$ and $x_{\max,i}$ and how we can deal with the problems in case of an incremental case-based learning system.

4 Experiment

In the following we want to demonstrate how the classifier behaves if we do not know the right values for the upper and lower bounds at different points in time. The experiment is done based on the iris data set. The data set has four features and 150 data. The data are grouped in to three classes.

We observe the development of the error rate for each feature depending on 5 different data sets $N_1 \dots N_5$.

The following tests have been carried out:

1. Development of the error rate $p_{x,i}$

For the data set N_1 we choose randomly five data from each class from the iris data set. For data set N_2 we choose again five data from each class and add this data to data set N_1 . Analogously, we choose for N_3 ten data from each class and for N_4 and N_5 fifteen data from each class. Note that the same data can be used in two or more data sets.

For each data set N_k and for each feature i , we calculate $x_{\min,i,k}$ and $x_{\max,i,k}$.

Then we compute $T_{i,k} = x_{\max,i,k} - x_{\min,i,k}$ for each feature i .

Furthermore, let \tilde{T}_k be the set $\tilde{T}_k = \{T_{1,k}, \dots, T_{i,k}, \dots, T_{n,k}\}$.

For the data sets N_k and $x \in N_{k+1} \setminus N_k$ we check for each feature i if $x_{\min,i,k} \leq x \leq x_{\max,i,k}$ or not.

Then we can calculate $p_{x,i,k}$

$$P_{x,i,k} = \frac{\text{number nonclassified objects}}{\text{number all objects } x \in N_{k+1} \setminus N_k} \quad (11)$$

2. Development of the error rate p_c

We choose the cases in data sets N_1 as cases. As a test data set we choose the cases from the data set $N_5 \setminus N_1$. For the following test we normalized the data with the different \tilde{T}_k 's calculated for the different data sets $N_1 - N_5$. We use our classification system **ProtoClass** [20] for this test and chose $NN=1$ and $NN=3$ for the nearest neighbors.

3. Observation of the Similarity Values

Example

For the data set N_1 (with 15 cases) we calculate the normalized similarity among each other for the 5 tolerance range sets $\tilde{T}_1 - \tilde{T}_5$.

Then we calculate for each two cases x and y the following changes of similarity:

$$\Delta Sim(x, y, t_1, t_2) \quad (12)$$

$$\Delta Sim(x, y, t_2, t_3) \quad (13)$$

$$\Delta Sim(x, y, t_3, t_4) \quad (14)$$

$$\Delta Sim(x, y, t_4, t_5) \quad (15)$$

Subsequently, we calculate the mean change of similarity and the maximum change of similarity from (12)-(15).

5 Results

Figure 4 shows the nonclassified data in % depending on the size of the data sets and the data set $N_1 \dots N_5$. The error rate is as high as it is when we do have small sample sets. It drops significantly when we have a significantly large data set of 30 samples but the curve is in general not asymptotic. When new upper and lower values of a feature i are encountered, then the error rate goes up and T_i has to be changed. We did not take into account the difference ΔT_i in order to update T_i . If we were to do that we would be able to deal with the outlier problem.

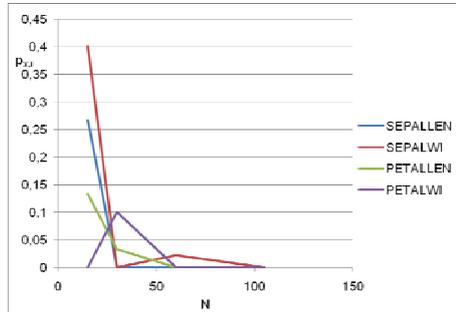


Fig. 4. Error rate $p_{x,i}$ versus sample number

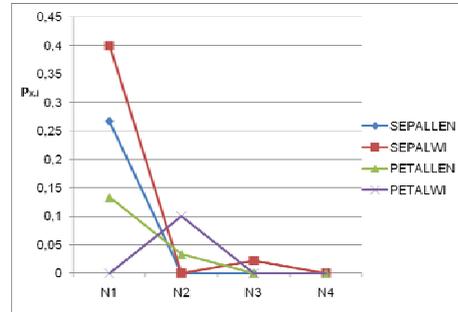


Fig. 5. Error rate $p_{x,i}$ versus sample sets $N_1...N_4$

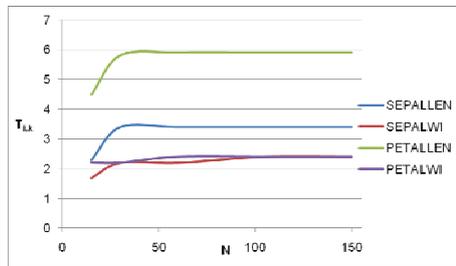


Fig. 6. Tolerance range $T_{i,k}$ versus sample number

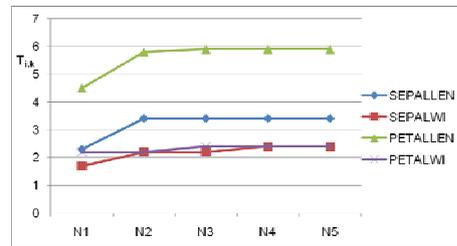


Fig. 7. Tolerance range $T_{i,k}$ versus sample sets $N_1...N_5$

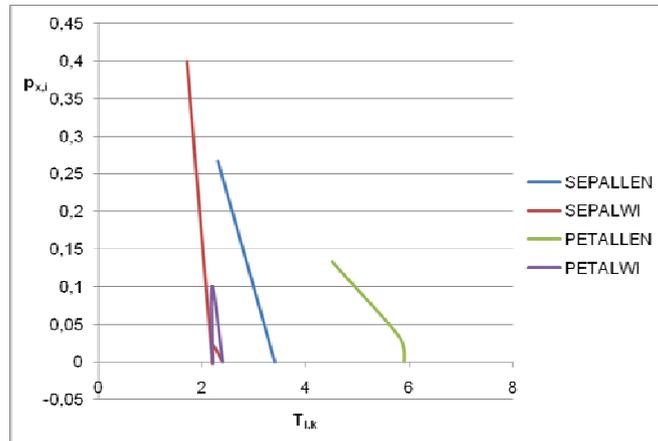


Fig. 8. Error rate $p_{x,i}$ versus $T_{i,k}$ for all four different features

Figure 6 shows that $T_{i,k}$ increase as more data are arriving and that, after some time, it goes into saturation. Figure 8 shows the nonclassified data in % depending on $T_{i,k}$. Small changes of $T_{i,k}$ causes large changes of the error rate.

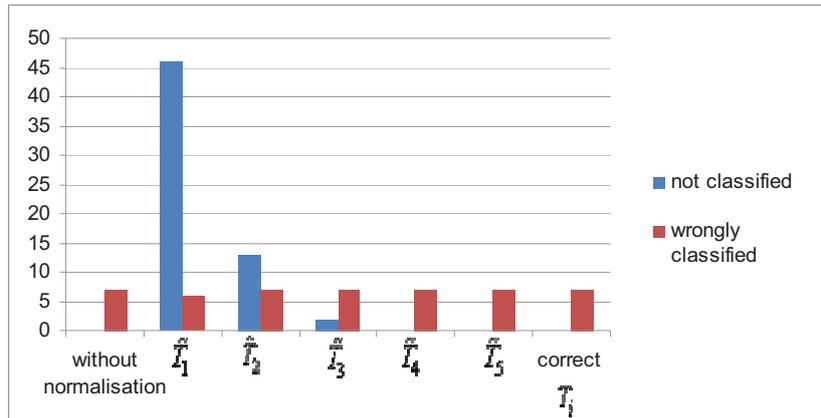


Fig. 9. Error rate p_c versus tolerance range $\tilde{T}_1 \dots \tilde{T}_5$ for $NN = 1$

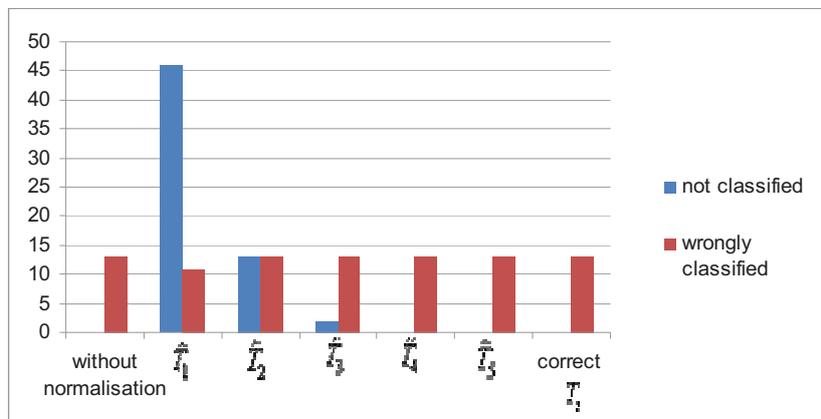


Fig. 10. Error rate p_c versus tolerance range $\tilde{T}_1 \dots \tilde{T}_5$ for $NN = 3$

The nonclassified data decreases the more $T_{i,k}$ converges toward the correct T_i (see Figure 9 for $NN=1$ and Figure 10 for $NN=3$). In this example the wrongly classified data are always the same. This might be different in other experiments but at least it shows that the wrong classification is not the main problem. The problem is that some samples cannot be classified and case-based maintenance [21] has to be started. The classification is done based on the nearest neighbor principle. The semantic scale does not play a significant role in this situation. If we look for the changes in the similarity value, it is apparent that the similarity value changes

significantly. Figure 11 shows for some samples the pair-wise similarity value for different \tilde{T}_k . The samples become more similar as $T_{i,k}$ becomes larger and as $T_{i,k}$ converges more toward the real T_i .

The mean in similarity range is from 0.07 to 0.0 (see Figure 12). The max similarity value range is from 0.2 to 0.0. This means that the semantic of the similarity changes drastically. Samples that are more dissimilar become more similar after $T_{i,k}$ converges toward the real T_i s.

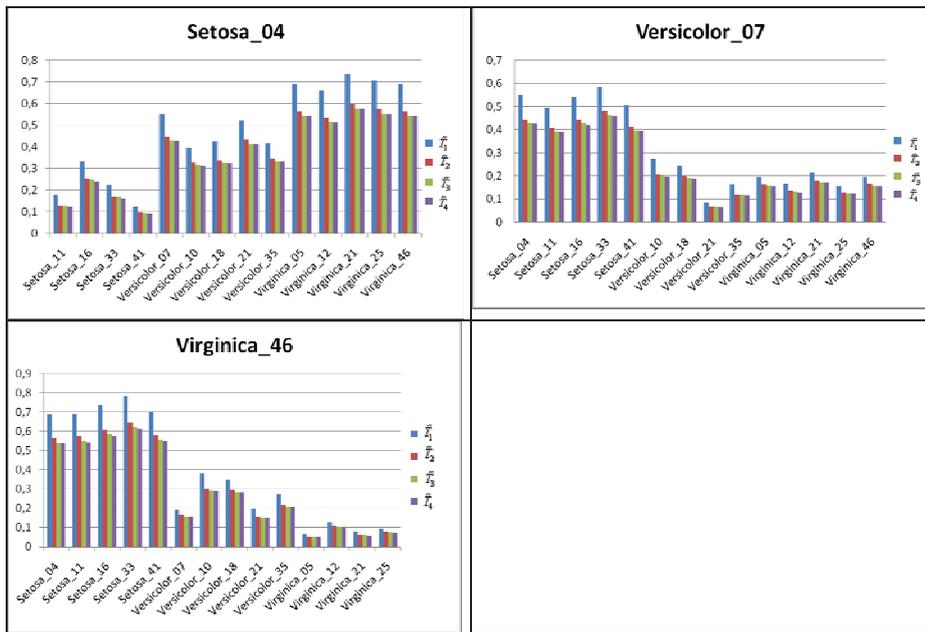


Fig. 11. Pair-wise similarity for three cases depending on several cases

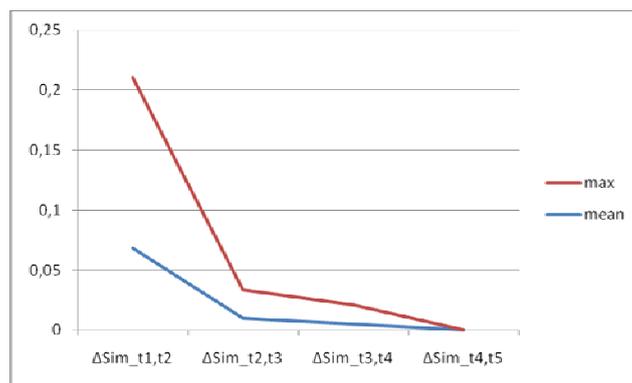


Fig. 12. Mean and maximal similarity difference for four different points in time

6 Discussion

Normalization of the similarity measure by the lower and upper bounds of the features i allow us to compare different samples on a scale between 0 and 1 . The similarity is thus imparted with a semantic meaning and allows a human to better grasp the similarity.

This semantic is only true if the values used for normalization are true.

If we normalize to a lower and an upper bound of the features i , then $x_{\min,i}$ and $x_{\max,i}$ must be true values. If we only use the nearest neighbor rule then classification is done based on the nearest distance. The semantic scale plays no role. We are then only faced with the problem that samples cannot be classified because of wrong lower and upper bounds. Depending on how large an error rate we allow for each feature, we have to change the lower and upper bounds after some time.

A case-based reasoning system has to monitor the lower and upper bounds for the feature value of each feature and start case-based maintenance when the error rate $p_{x,i}$ exceeds a threshold. The system should keep the frequency histogram for each feature in memory and update this histogram incrementally until a probability density distribution can be estimated from the histogram.

If we rely on the semantics of the similarity helping a human to judge the similarity between two samples, we are faced with the problem of wrong classification and with the problem of wrong semantic. Two cases that might have been more dissimilar might become more similar after some time. If these two cases are judged based on the semantic rule, they might be wrongly classified in the end.

This means that, as long as T_i has not reached the true values, the semantic similarity rule should not be used for classification or the semantic similarity should not be shown for comparison purposes. It is better to rely on the nearest neighbor principle.

7 Conclusion

Normalization allows us to compare different cases on a scale from 0 to 1 and provides us with a semantic understanding of similarity. The value of 0 means identity and the value of 1 means dissimilar. In between, the value of 0.5 means neutral while the values on a scale from 0.5 to 0 mean more similar and the values from 1 to 0.5 mean more dissimilar.

The problem of normalization has not been studied in CBR yet under the aspect of different normalization methods and the methods for parameter setting. The situation is tricky if we are dealing with online applications where we want to start with a small set of data that increases over time and we need to develop incremental methods for normalization.

In our study we investigated what the problem with normalization is in CBR. A study in that direction has not been done before in CBR although the method of

normalization and the method of parameter setting has a big impact on the performance of the CBR system. With our study we like to direct the intention of the researchers to this problem and give a first idea how to deal with it.

The values of the parameter for normalization must be the true values. If we normalize to an upper and a lower bound of the features i , then $x_{\min,i}$ and $x_{\max,i}$ should be the true values; if not, problems arise. One of these problems is misclassification of cases based on one or more features. The feature value may lie outside the min or max value. Then the case cannot be classified because of this problem and the system must adjust these values based on the data processed in the respective time period. This is a process of case-based maintenance.

If we want the similarity to have semantic meaning, then the meaning depends on the true values. If the values for normalization are too low, then the cases will become more similar if the values converge toward the true values.

This means that cases that have been found to be dissimilar and to belong to the same class must be re-classified after some time.

This is not feasible in an online-system; since the true parameters for normalization are not known. Therefore, it is better to classify based on the nearest neighbor rule instead of the semantic similarity rule. Nonetheless, even if the scale is not the right one, it can help to judge the similarity between different cases. However, it might not have the right semantic meaning. The question is: Can we predict the right semantic meaning out of different similarity measures at different times. This question is left to be answered by further work.

References

1. Richter, M.M.: Introduction to Case-Based Reasoning. In: Lenz, M., Bartsch-Spörl, B., Burkhardt, H.D., Wess, S. (Eds.) *Case-based Reasoning Technology: from Foundations to Applications*, InAI, vol. 1400, pp. 1-16, Springer, Heidelberg (1998)
2. Richter, M.: Classification and Learning of Similarity Measures. In: Opitz, Lausen, Klar (Eds.) *Proc Jahrestagung der Deutschen Gesellschaft für Klassifikation (Studies in Classification, Data Analysis, Knowledge Organisation)* Springer Verlag (1992)
3. Xiong, N., Funk, P.: Building similarity metrics reflecting utility in case-based reasoning. *Journal of Intelligent & Fuzzy Systems* 17, 407-416 (2006).
4. Xiong, N., Funk, P.: Combined feature selection and similarity modeling in case-based reasoning using hierarchical memetic algorithm. In: *Proceedings of the IEEE World Congress on Computational Intelligence*, pp. 1537-1542 (2010)
5. Wettschereck, D., Aha, D.: Weighting features. In: Veloso, M. M., Aamodt, A (Eds.): *ICCBR 1995, Incs*, vol. 1010, pp. 347-358, Springer Heidelberg (1995)
6. Little, S., Colantonio, S., Salvetti, O., Perner, P.: Evaluation of Feature Subset Selection, Feature Weighting, and Prototype Selection for Biomedical Applications, *Software Engineering & Applications* 3 (1), pp. 39-49 (2010)
7. Ahn, H., Kim, K., I. Han: Global optimization of feature weights and the number of neighbors that combine in a case-based reasoning system. *Expert Systems* 23, 290-301 (2006)
8. Bichindaritz, I.: Methods in Case-Based Classification in Bioinformatics: Lessons Learned. In: Perner, P. (Edt.), *ICDM 2011, LNCS*, vol. 6870, pp. 300-313, Springer, Berlin (2011)

9. Bichindaritz I.: Comparison of Reuse Strategies for Case-Based Classification in Bioinformatics. In: Wiratunga, N., Ram, A. (Eds.) ICCBR 2011, Incs, vol. 6880, pp. 393-407, Springer, Berlin (2011)
10. Jarmulak, J., Craw, S., Rowe, R.: Genetic algorithms to optimize CBR retrieval. In: Blanzieri, E., Portinale, L (Eds.) EWCBR 2000, Incs, vol. 1898, pp. 136-147, Springer, Heidelberg (2000)
11. Cheng, W., Hüllermeier, E.: Learning similarity functions from qualitative feedback In: Althoff, K.D., Bergmann, R., Minor, M., Hanft, A. (Eds.): ECCBR 2008. Incs, vol. 5239 pp. 120-134, Springer, Heidelberg (2008)
12. Bobrowski, L., Topczewska, M.: Improving the K-NN Classification with the Euclidean Distance Through Linear Data Transformations. In: Perner, P. (Ed.): ICDM 2004, Incs, vol. 3275, pp. 23-32, Springer, Heidelberg (2004)
13. Perner P.: Case-Based Reasoning for Image Analysis and Interpretation, In: Chen, C., Wang, P.S.P.: Handbook on Pattern Recognition and Computer Vision, 3rd Edition, pp. 95-114, World Scientific Publisher (2005)
14. Fayyad, U.M., Piatetsky-Shapiro G., Smyth P, Utuhursamy, R.: Advance in Knowledge Discovery and Data Mining, AAAI Press (1996)
15. Perner, J.: Characterizing Cell Types through Differentially Expressed Gene Clusters Using a Model-based Approach, ibai-publishing, Fockendorf, ISBN 978-3-942952-01-9 (2011)
16. Stolcke A., Kajarekar S., Ferrer L.: Nonparametric feature normalization for SVM-based speaker verification. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2008, pp. 1577 – 1580 (2008)
17. Molau S., Pitz M., Ney, H.: Histogram based normalization in the acoustic feature space, 2. IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 21-24 (2001)
18. Pisoni, D.B.: Long-term memory in speech perception: Some new findings on talker variability, speaking rate and perceptual learning. *Speech Communication* 13 (1-2) October, 109-125 (1993)
19. Dreyer, H., Sauer, W.: *Prozeßanalyse*, Verlag Technik, Berlin (1982)
20. Perner, P.: Prototype-Based Classification. *Applied Intelligence* 28(3), 238-246 (2008)
21. Perner, P.: Case-base maintenance by conceptual clustering of graphs. *Engineering Applications of Artificial Intelligence*. 19 (4), 381-295 (2006)

Appendix

SEPALLEN	Iris	N_1	N_2	N_3	N_4	N_5
$ N_k $	150	15	30	60	105	150
$x_{\min,i,k}$	4,3	4,6	4,3	4,3	4,3	4,3
$x_{\max,i,k}$	7,9	6,9	7,7	7,7	7,7	7,7
$T_{i,k}$	3,6	2,3	3,4	3,4	3,4	3,4
$\mu_{i,k}$	5,8433	5,8467 (Median: 5,8000)	5,7333	5,7617	5,8429	5,8287
$\sigma_{i,k}$	0,8281	0,72198563	0,8401	0,8491	0,8866	0,8647
Number of nonclassified data $x \in N_{k+1} / N_k$	-	4 Setosa_14, Setosa_42, Setosa_43, Virginica_24	0	0	0	-
$p_{x,i,k}$	-	4/15=0,27	0	0	0	-

SEPALWI	Iris	N_1	N_2	N_3	N_4	N_5
$ N_k $	150	15	30	60	105	150
$x_{\min,i,k}$	2	2,7	2,2	2,2	2	2
$x_{\max,i,k}$	4,4	4,4	4,4	4,4	4,4	4,4
$T_{i,k}$	2,4	1,7	2,2	2,2	2,4	2,4
$\mu_{i,k}$	3,0573	3,2600 (Median: 3,2000)	3,0233	3,0383	3,0410	3,0407
$\sigma_{i,k}$	0,4359	0,50448827	0,4967	0,4357	0,4415	0,4313
Number of nonclassified data $x \in N_{k+1} / N_k$	-	6 Setosa_42 Versicolor_04 Versicolor_41 Virginica_07 Virginica_19 Virginica_20	0	1 Versicolor_11	0	-
$p_{x,i,k}$	-	6/15=0,4	0	1/45=0,022	0	-

SEPALWI	Iris	N_1	N_2	N_3	N_4	N_5
$ N_k $	150	15	30	60	105	150
$x_{\min,i,k}$	1	1,3	1,1	1	1	1
$x_{\max,i,k}$	6,9	5,8	6,9	6,9	6,9	6,9
$T_{i,k}$	5,9	4,5	5,8	5,9	5,9	5,9
$\mu_{i,k}$	3,7580	3,8000 (Median: 4,5000)	3,7400	3,7400	3,7733	3,7473
$\sigma_{i,k}$	1,7653	1,83056769	1,7884	1,7676	1,7968	1,7716
Number of nonclassified data $x \in N_{k+1} / N_k$	-	2 Setosa_14 Virginica_19	1 Setosa_23	0	0	-
$p_{x,i,k}$	-	=2/15=0,133	=1/30=0,033	0	0	-

PETALWI	Iris	N_1	N_2	N_3	N_4	N_5
$ N_k $	150	15	30	60	105	150
$x_{\min,i,k}$	0,1	0,1	0,1	0,1	0,1	0,1
$x_{\max,i,k}$	2,5	2,3	2,3	2,5	2,5	2,5
$T_{i,k}$	2,4	2,2	2,2	2,4	2,4	2,4
$\mu_{i,k}$	1,1993	1,2867 (Median: 1,5000)	1,2233	1,2317	1,2152	1,2140
$\sigma_{i,k}$	0,7622	0,85880787	0,7981	0,7884	0,7922	0,7830
Number of nonclassified data $x \in N_{k+1} / N_k$	-	0	3 Virginica_37 Virginica_41 Virginica_45	0	0	-
$p_{x,i,k}$	-	0	3/30=0,1	0	0	-