Fuzzy Models For Recurring Phenomena In Time Series

Gernot Herbst, Steffen F. Bocklisch

***Abstract*— This article aims at extending fuzzy classification methods to the recognition of local patterns in time series. Firstly, a classifier model for patterns will be introduced, which allows for the prevalent uncertainty in measured data. It is able of classifying subsequences, and, due to its *white box* character, is easily comprehensible as well as modifiable by users. Based upon that, different pattern recognition tasks will be introduced, all from the classification perspective, which shift their emphasis from data mining to diagnosis and prognosis. Due to the classification approach, diagnosis and prognosis may always have both quantitative and qualitative character. Secondly, a fuzzified notion of periodicity will be introduced, which will also enable a more unambiguous recognition of recurring patterns. The overall goal of this work is to provide interpretable and transparent models and tools which may be combined to create a consistently fuzzy recognition system for time series patterns.**

I. Introduction and Basic Models

A. Local phenomena in time series

Under a *phenomenon* in a time series (from the Greek $\varphi\alpha\iota\nu o\mu\varepsilon\nu o\nu$, something that appears) we would like to understand a pattern that becomes visible and active locally, i. e. within certain temporal bounds. Despite the term "bounds", phenomena may exhibit a certain fade in- and fade out-behaviour, thusly precise boundaries may not always be determined. In addition to this temporal uncertainty, their spatial characteristics (amplitude) will be influenced by noise or other sources of imprecision, as well. The temporal locality will rule out many existing approaches to time series processing, as they rely on a global model of the time series or even require strict properties such as stationarity. For these reasons, we will focus on shape-based fuzzy models for the detection of phenomena throughout the article.

B. A multivariate parametric fuzzy set

Equation (1) describes a parametric membership function which will form a foundation of the models presented in this article.[1] It is based on the potential function and has already proven useful in numerous areas of application such as process surveillance [1], time series modelling [2] and classifier networks [3]. An advantage of this membership function is the possibility of representing asymmetric fuzzy sets by individual parameters for the left- and right-hand-side function branches.

G. Herbst and S. F. Bocklisch are with the Faculty of Electrical Engineering, Chemnitz University of Technology, D-09107 Chemnitz, Germany. E-Mail: gernot.herbst@etit.tu-chemnitz.de

[1]This choice seems advantageous for the reasons presented in this section, however, the models and methods will be rather general and not irrevocably be tied to a particular membership function type.

$$\mu(x) = \begin{cases} \dfrac{a}{1 + \left(\dfrac{1}{b_l} - 1\right)\left(\dfrac{r-x}{c_l}\right)^{d_l}} & : \quad x < r \\[2ex] \dfrac{a}{1 + \left(\dfrac{1}{b_r} - 1\right)\left(\dfrac{x-r}{c_r}\right)^{d_r}} & : \quad x \geq r \end{cases} \tag{1}$$

Apart from the normalised case ($a = 1$), a represents the maximum truth value in the course of μ, occurring at $\mu(x = r)$. The six parameters $b_{l/r}$, $c_{l/r}$ and $d_{l/r}$ determine the extent and shape of the class. The basic effect of b ($0 < b < 1$) and c ($c > 0$) can be understood from Fig. 1, whereas d ($d \geq 2$) influences the manner of μ's descent to zero, with increasing d leading to a sharper descent and $d \to \infty$ resulting in a rectangular shape. This is particularly interesting since crisp sets such as interval descriptions can be formulated using (1) as well.

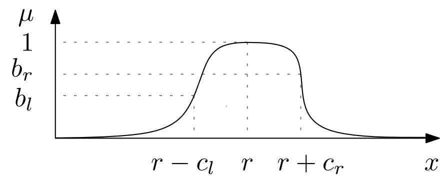

Fig. 1. Normalised parametric membership function (1).

The interpretability of these parameters facilitates the formulation of expectations by experts on the one hand, and enables the quick comprehension of results of a learning step on the other hand, where the parameters are determined by an algorithm [1] processing a given set of data representing one class of objects.[2]

Another key advantage of this function concept is that a multivariate parametric membership function can be derived from a conjunction of one-dimensional sets using a compensatory Hamacher intersection operator [1]. The n-fold operator is given by (2). Contrary to the often-employed min-intersection operator, it takes into account all provided truth values in a compensatory manner. Equation (3) shows the resulting multidimensional membership function in a simplified, since symmetric form, and Fig. 2 depicts two examples of membership functios defined over a two-dimensional feature space.

[2]The parameters of (1) may be computed from data as follows: Firstly, the central, "average" object value r is being calculated. Then $c_{l/r}$ are chosen such that all learning objects are included in these left- and right-hand-sided spans. Finally, the shape parameters b and d parameters are—if not specified manually—being computed to best fit the dispersion of the data.

$$\cap_{Ham}^{n} = \frac{1}{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\mu_i}} \quad (2)$$

$$\mu(\boldsymbol{x}) = \frac{a}{1 + \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{b_i} - 1\right)\cdot\left|\frac{x_i - r_i}{c_i}\right|^{d_i}} \quad (3)$$

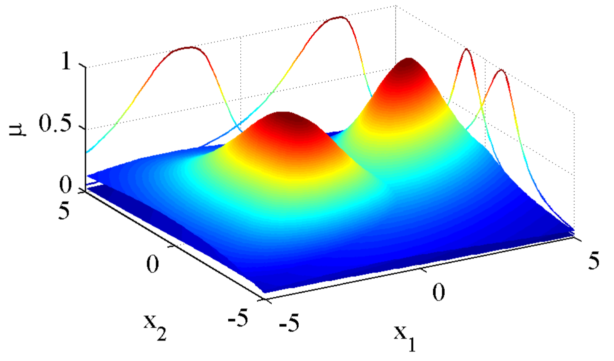

Fig. 2. Two exemplary two-dimensional membership functions.

C. A fuzzy model for time series patterns

Based on the membership function concept described in section I-B, [4] and [2] introduce a model for multivariate time series sequences. It allows to compute a continuous degree of similarity between a candidate sequence (time series pattern) and the model, and therefore treats pattern matching as a fuzzy classification problem. The model may be employed for equidistantly sampled univariate or multivariate time series.

The basic principle is to describe each point i of a pattern of length N by a fuzzy class (membership function) $\mu_i(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, which is the reason for the soft tolerance towards noise and other sources of imprecision that this model offers.

To compare a pattern against the model, i. e. to classify a pattern given by N points $\boldsymbol{x}(1), \ldots, \boldsymbol{x}(N)$, one has to combine the individual classification results of each point of the pattern to an overall degree of similarity $\mu \in [0, 1]$ by means of a fuzzy conjunction operator:

$$\mu = \mu_1(\boldsymbol{x}(1)) \cap \cdots \cap \mu_N(\boldsymbol{x}(N)) \quad (4)$$

Following the fundamental ideas of the membership function type of section I-B, it suggests itself to employ the same conjunction operator given by (2). In this manner, the model for the entire pattern could also be thought of as *one* fuzzy class of dimension $N \cdot n$ (number of points in pattern times dimensionality of the time series). Thusly classification of time series sequences would be completely in line with the multivariate classification paradigm behind section I-B. Contrary to many *black box* models such as neural networks, one key advantage of this model is to support partial classification, i. e. classification of any possible subsequence.

Fig. 3 depicts an example of an univariate pattern (length: 20 seconds, sample time: one second) described by this fuzzy time series model. By means of this figure one can nicely comprehend the interpretation of that series of classes as a corridor with soft boundaries that a candidate pattern would follow depending on its similarity to this particular pattern.

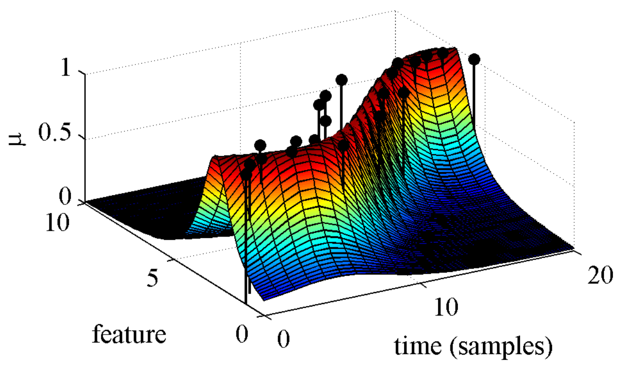

Fig. 3. Fuzzy description of a time series sequence along with a noisy candidate pattern (both generated artificially for illustration).

A fuzzy time series pattern may either be formulated by an expert—enabled by the interpretability of the individual parameters—or be learned from a set of patterns describing the same phenomenon, employing the learning algorithm mentioned in section I-B for each point of the pattern. Fig. 4 shows an exemplary set of pattern instances leading to a fuzzified description in Fig. 5 using this procedure.

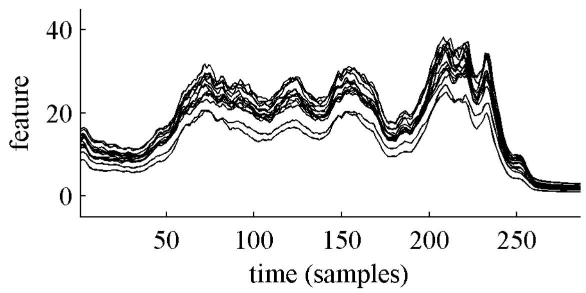

Fig. 4. Several instances of a time series pattern (“Coffee” dataset from the UCR time series database [5]).

II. Pattern Recognition Tasks

A. From offline to online pattern recognition

In the following, three different pattern recognition tasks shall be described with special regard to their respective degree of *online-ness*, all of them viewed from a fuzzy classification perspective. We assume to possess a model for a pattern of a certain length L that allows to classify an equally long subsequence of a univariate or multivariate time series $\boldsymbol{x}(t)$ resulting in a degree of similarity between the model and the candidate pattern.[3]

[3]During this section, no assumptions w. r. t. sampling time etc. will be made, instead, a more general “continuous” case will be sketched. A suitable model for equidistantly sampled time series has been proposed in the previous section.

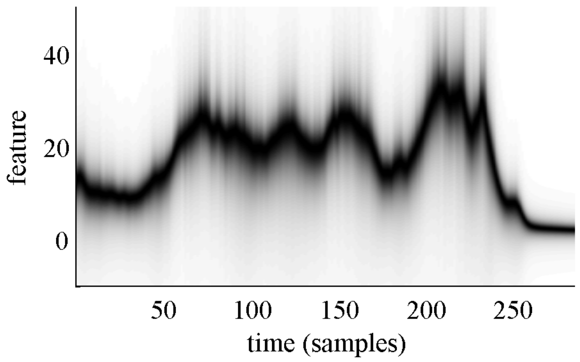

Fig. 5. Fuzzy time series pattern learned from Fig. 4, here in a 2D representation. Dark areas correspond to high degrees of membership.

1) The data mining perspective.: The "most offline" task—depicted by Fig. 6—consists of finding a completed pattern in a time series being available as a whole. The results of this task could be locations in the time series where the pattern is believed to have occurred, complemented by a degree of similarity $\hat{\mu}$ of the respective subsequence of $\boldsymbol{x}(t)$ to the pattern described by our model. This corresponds to transforming the time series $\boldsymbol{x}(t)$ into a series of events (or symbols). Internally, this could be achieved by classifying all possible subsequences of the time series and afterwards deciding on the best match. This kind of pattern recognition could be part of a data mining task.

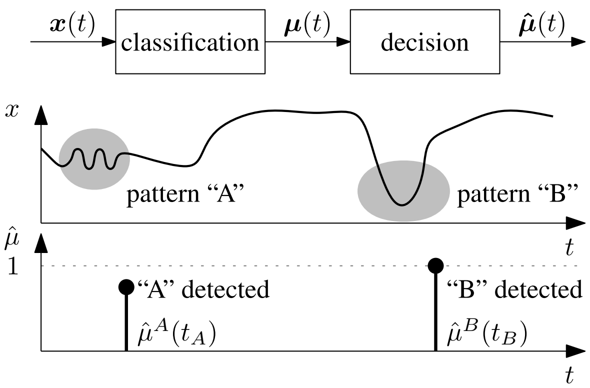

Fig. 6. Offline recognition of completed patterns: Two different patterns were recognised in a time series $x(t)$.

2) The diagnosis perspective.: A "more online" task would be to find completed patterns in a streaming time series, i. e. classifying the latest available subsequence at every point in time. As we assume a fuzzy model for the pattern, this results in a time series $\mu(t)$ containing the degree of similarity of the subsequence $\boldsymbol{x}([t-L]\ldots t)$ and the model, cf. Fig. 7. In a truly fuzzy sense, each pattern would be found at each position in the time series, albeit with a possibly negligible degree of similarity.

This procedure could be seen as a basis of the previous pattern recognition task, now without the decision step—which obviously is not as easily feasible, since a decision on the global best match cannot be inferred from a yet-to-be-completed time series $\mu(t)$. An interesting side note is, however, that the evolving classification result $\mu(t)$—being a time series itself—can again be processed using time series analysis methods in further steps [6], [7]. Finding completed patterns in an "online" manner could be interpreted as a diagnostic task. As in technical or medical diagnosis, a decision often has to be made immediately after a pattern's detection, and thusly under incomplete knowledge.

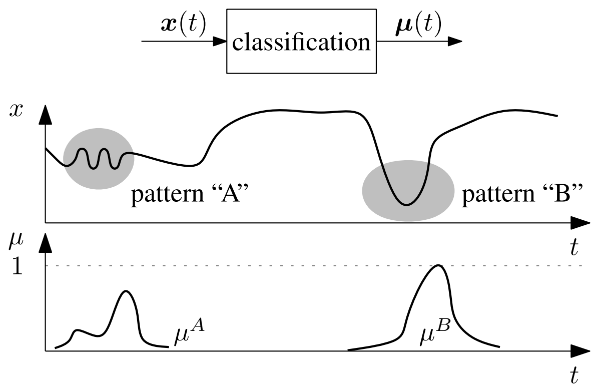

Fig. 7. Online recognition of completed patterns. The two time series of recognition results $\mu^A(t)$ and $\mu^B(t)$ are obtained by sliding the models over the time series $x(t)$ and continuously computing a degree of similarity to the respective subsequence.

3) The prognosis perspective.: The "most online" and generalised task arises from the previous pattern recognition problem when patterns shall be detected while they are evolving. This would shift the viewpoint from a diagnosis to a prognosis perspective. In machine surveillance, for instance, this would allow operators to perform preventive maintenance actions *before* an actual (severe) damage occurs, rather than just being able to diagnose a damage *afterwards*.

In doing so, the complexity of the pattern recognition problem is being increased by another dimension. In addition to the information which pattern is being active to which degree at any given point in time t, one has to determine the time τ elapsed *within* the respective pattern, compare Fig. 8. The remainder of the pattern may then be used to forecast $\boldsymbol{x}(t)$ locally.

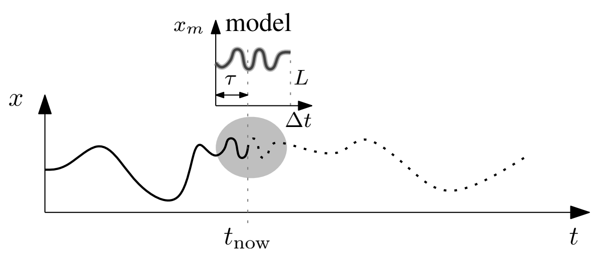

Fig. 8. Matching of evolving patterns. At the current point in time t_{now}, the model containing the desired pattern $x_m(\Delta t)$ is being adjusted in its relative position τ such that the subsequences $x([t_{\text{now}}-\tau]\ldots t_{\text{now}})$ and $x_m(0\ldots\tau)$ match optimally.

From the fuzzy logic perspective, obtaining τ would require a crisp decision, which—in a fuzzy recognition system—should be left to the user. With truly fuzzy models, *each* pattern would be found at *every* point in time with *all* possible relative positions τ. In this article, we therefore propose to describe this (rather complex) recognition result for a certain pattern at some point in time t by a fuzzy set $\mu(t, \tau)$, $0 \leq \tau \leq L$, which contains the truth values representing the similarities of the subsequences $\boldsymbol{x}([t-\tau] \ldots t)$ (corresponding to partially elapsed patterns) to the respective parts of the model, cf. Fig. 9.

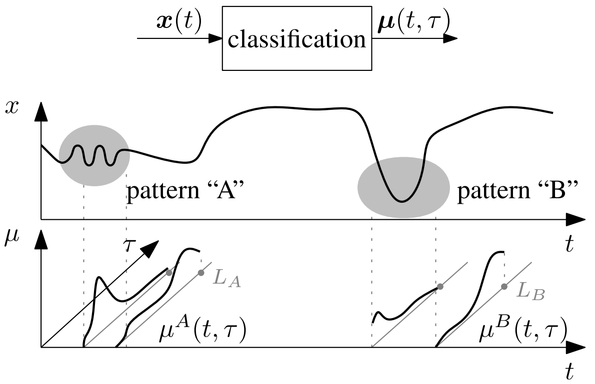

Fig. 9. Online recognition of patterns. At each point in time t, the maximum truth values of μ^A and μ^B w. r. t. τ point to the relative positions τ associated to the best match of the partially elapsed patterns "A" and "B".

It appears worthy to mention that describing the recogition result by a fuzzy set—although it might seem overly complex at first—is perfectly in line with the fuzziness of the recognition task. When searching for completed patterns, a truth value μ for each point in time t could be a sufficient fuzzy recognition result. Searching for incompleted patterns additionally requires information about τ, which, by means of the fuzzy set $\mu(t, \tau)$, is being given in fully fuzzy form. For each t, $\mu(t, \tau)$ can be interpreted as a fuzzy measure[4] for the relative position τ of the partially elapsed pattern.

One example of the flexibility offered by $\mu(t, \tau)$ is the possibility to modify or emphasise certain parts of this set to focus the detection of the respective pattern to a certain stage of evolvement τ, for example if the detection of a pattern would be especially important in its first few seconds. This could, for instance, be achieved by conjunction with a set representing a fuzzy window of interest for τ.

B. Relation of the pattern recognition tasks

The three different pattern recognition tasks described in the previous section are interconnected in a top-down manner. The "more online" tasks are more generalised and may form the basis for "more offline" tasks. The generalisation not least becomes visible in the complexity of their results. The most complex and most general result is a fuzzy set evolving over time $\mu(t, \tau)$, proposed in section II-A.3, which forms the result of online recognition of evolving patterns.

The online recognition of completed patterns can therefore be seen as a specialised case of the latter task, and, in fact, its recognition result (the time series of truth values $\mu(t)$) could be derived from $\mu(t, \tau)$, as it already contains the degree of the completed pattern (length L):[5]

$$\mu(t) = \mu(t, \tau = L) \tag{5}$$

To obtain a decision on the position of a detected pattern for the data mining task described firstly, the time series of recogntion results $\mu(t)$ must be completely available to enable to globally optimal decision on the best match.

$$(\hat{\mu}, t^*) = f[\mu(t)] \tag{6}$$

A very simple approaches of obtaining such a decision would be the search for the maximum truth value in $\mu(t)$:[6]

$$\hat{\mu} = \max \mu(t), \quad t^* = \arg\max_t \mu(t) \tag{7}$$

Fig. 10 visualises the relation between the three types of pattern recognition tasks described in section II-A at the level of truth values: Trying to find a pattern at each possible position t with all possible relative positions τ leads to a series of fuzzy sets $\mu(t, \tau)$. From this, the "semi-online" recognition result (online detection of completed patterns) $\mu(t)$ can be derived according to (5). Using a simple decision procedure as described by (7), the time series $\mu(t)$ could be transformed into a series of events (symbols) as shown in the picture.

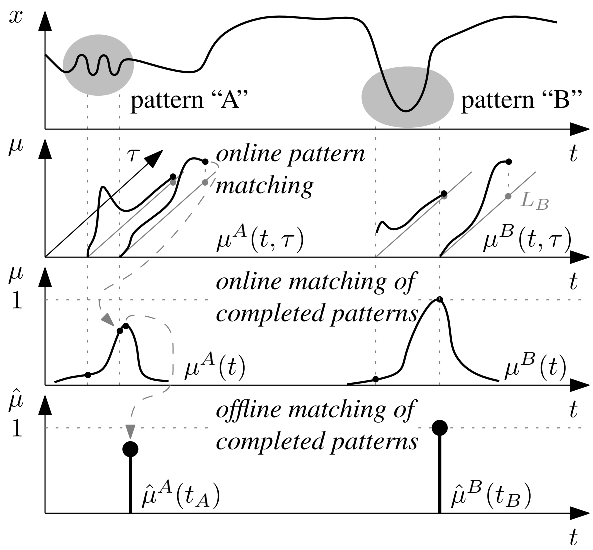

Fig. 10. Relation of pattern matching tasks.

[4]Although not being a fuzzy number in the sense of Dubois and Prade [8], especially owing to the very likely multimodality, it matches the spirit of a fuzzy number as a fuzzified representation of a real-valued number quite well.

[5]The definition in (5) does not necessarily have to hold, this depends on the manner $\mu(t, \tau)$ is being computed and if $\mu(t)$ and $\mu(t, \tau = L)$ have the same semantic meaning, *viz.* being a truth value describing the similarity of a completed pattern to the latest available subsequence (length L) of the time series $\boldsymbol{x}(t)$.

[6]With this simple approach, however, only one instance of a pattern may be found in the time series $\boldsymbol{x}(t)$.

III. RECURRING PHENOMENA

Under a recurring phenomenon in a time series we would like to understand a phenomenon or pattern which repeatedly occurs in a time series $\boldsymbol{x}(t)$ according to a certain *rhythm*. Contrary to strict periodicity, rhythmic behaviour—in the manner we would like to understand it in this article—does not necessarily have to be associated with a fixed frequency. A rhythm might be a frequency that varies, "breathes" in certain (fuzzy) boundaries, such as many natural rhythms do, especially in biological systems.

A. Indicating features.

In the following, we would like to denote a phenomenon in $\boldsymbol{x}(t)$ as *recurring according to a certain rhythm* if one or more features $\boldsymbol{y}(t)$ may be derived from $\boldsymbol{x}(t)$ or t which exhibit similar, characteristic values every time the phenomenon occurs. The respective $\boldsymbol{y}$ then shall be called *indicating features*. These features do not inevitably have to be temporal features, although the latter are the most obvious ones, such as the temporal distance to the previous instance of this phenomenon. Other examples can be found in calendar-based timestamps, where each consecutive timestamp is attached several attributes of different temporal granularities [9], [10] such as day, week, month. For instance, a phenomenon might occur around the 20th of each month. Although a precise frequency cannot be given for this repetitive behaviour, it clearly obeys a certain rhythm. We could attach the feature y ("day of month") to this phenomenon, which would lead to similar values (e. g. $y = day \approx 20$) for each instance.

B. Fuzzy expectations for indicating features

The goal of this article is to contribute to the online recognition of evolving patterns, with special emphasis on recurring patterns. Using fuzzy models and methods may help to detect patterns in *real-world* time series, i. e. time series exposed to noise and imprecision. In section II-A, however, we saw that in a truly fuzzy sense, *every* pattern can be found at *every* position t of a time series $\boldsymbol{x}(t)$ at *all* stages of evolvement τ at the same time.

It therefore appears necessary to make recognition results less ambiguous—if possible, in a fuzzy manner, i. e. without crisp decisions.[7] These should always be delayed to the very last step, so that a decision can be made based upon complete, non-pruned information that also includes the imprecision of the data and the recognition procedure.

In the case of a recurring phenomenon, it would be possible to employ its characteristic feature y (e. g. day of month, see above) as an indicator of its occurrence. More drastically, we could "turn off" the recognition of this particular phenomenon if the current value of the respective feature does not match the phenomenon's typical behaviour.

In this article, we propose to model the typical, *expected* behaviour w. r. t. the value of y associated to a phenomenon by means of a fuzzy set $\mu_e(\boldsymbol{y})$ defined over the space of its indicating features. Fig. 11 shows such an expectation for the examplary phenomenon mentioned above. Assessing the compliance of a phenomenon's characteristic features therefore becomes a second classification task. This approach resembles the idea of expectation functions presented in [7].

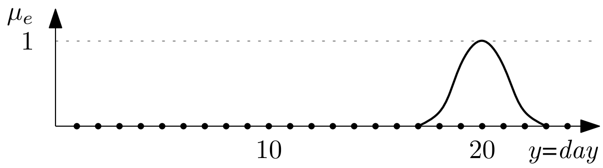

Fig. 11. Expectation μ_e for an exemplary rhythmic phenomenon with the characteristic feature "day of month".

C. Using fuzzy expectations

In the following, we sketch two options to employ expectations represented by μ_e for online pattern recognition:

1) Post-detection assessment: In this case, pattern recognition is being performed as outlined in section II-A, delivering results in the form of truth values. Before the results are reported to the user, the occurrence of the respective phenomenon is being evaluated by comparing its indicating feature values to the *a priori* knowledge described by μ_e. The results of the latter classification task now have to be combined with the pattern matching results, e. g. by a fuzzy conjunction operator, delivering the final recognition result $\mu_r(t)$, cf. Fig. 12. This procedure may be interpreted as a fuzzy weighting or "decision" on the importance or worthiness of the recognition result at this point in time.

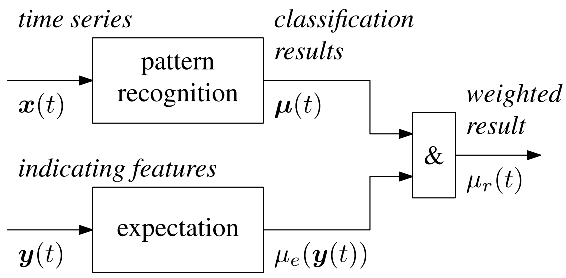

Fig. 12. Post-detection comparison of expectation and actual occurrence by conjunction of truth values.

2) Pre-detection activation: While the previous approach helps to disambiguate recognition results by combining them with *a priori* expectations, the latter may also be used to influence the actual pattern recognition step as shown in Fig. 13. Although concrete methods remain subject of further research, the pattern recognition algorithm might, for instance, try to omit computationally expensive operations if the pattern is scarcely being expected at this point in time, instead computing the similarity in a less detailed manner.[8]

[7] We cannot reduce the number of possible results in a fuzzy sense, as this involves hard decisions. However, since we are working with fuzzy truth values, it is possible to emphasise or diminish recognition results and thusly ease the user's subsequent decision step.

[8] If necessary, one could even deactivate the recognition of a pattern if it is not at all being expected right now, i. e. $\mu_e(\boldsymbol{y}(t)) \ll 1$. This, of course, would diminish the fuzzy character of the recognition procedure.

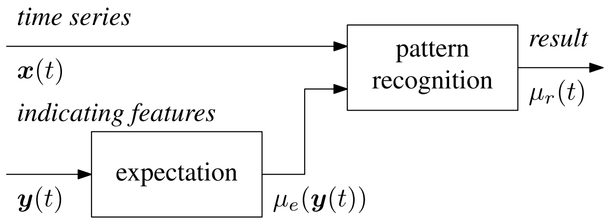

Fig. 13. Pre-detection assessment of indicating features to trigger, (de)activate or influence the actual pattern recognition step.

D. Expectations for offline and online recognition

When searching offline or for completed rhythmic patterns, the associated expectation $\mu_e(\boldsymbol{y})$ only has to represent the respective indicating feature value at the time the pattern is completed. In contrast, when searching for evolving patterns, the pattern recognition system has be to alert ($\mu_e > 0$) already when the pattern is about to begin. As the indicating feature value may change during the progression of the pattern, μ_e would accordingly have to cover a larger region of the indicating feature space. Here, the flexibility offered by a parametric fuzzy set as described in section I-B pays off: It is easly possible to adjust or learn the parameters such that the set describes a fuzzy interval of the feature space.

E. Learning expectations (example)

By employing indicating features and expectations, it furthermore becomes possible to distinguish similar patterns that are semantically different. In fact, it even becomes possible to learn different classes of rhythms by analysing the indicating feature space, e. g. by means of clustering methods, as in the following example: We process a time series of the daily number of *http* requests processed by a web server, cf. Fig. 14. For demonstration purposes, a trivial pattern ("low server load") will be searched. As a year-based rhythm is being suspected, each instance found will be attached the indicating feature "day of year".

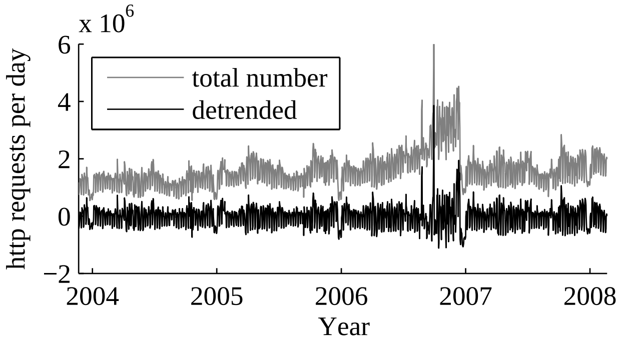

Fig. 14. *http* requests processed by Chemnitz University's web server. To enable comparability between years, the time series was detrended.

Following an offline pattern recognition step, all instances were examined in the indicating feature space. Visually or using clustering techniques, four classes can be found. These classes, subsequently transformed into fuzzy classes according to section I-B (as shown in Fig. 15), may now serve as expectations to distinguish semantically different pattern instances. The first class, for example, corresponds to the vacation period after the winter term, whereas the second class of instances is caused by single holidays during the summer term.

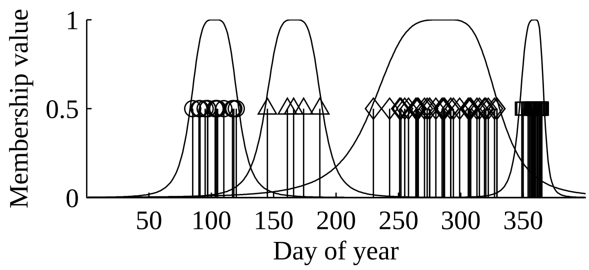

Fig. 15. Four expectation classes found for the rhythmic phenomenon "low server load" with the indicating feature "day of year". The singletons, representing detected instances and their feature values, have been set to $\mu = 0.5$ only for visualisation purposes.

IV. Conclusions

This article aimed at providing tools for pattern recognition in time series that allow fuzzified handling in an integrative manner. Different facets of pattern recognition were sketched as fuzzy classification problems. The two main contributions are the introduction of the online classification task for evolving time series patterns, and the concept of fuzzified expectations w. r. t. indicating features. These may describe rhythmic, but not necessarily periodic behaviour in a generalised manner, and may be used to disambiguate recognition results. As shown in an example, they may help to differentiate semantically different patterns, as well. Further work will include to jointly apply these methods to energy time series, which typically feature rhythmic patterns.

References

[1] S. F. Bocklisch, *Prozeßanalyse mit unscharfen Verfahren.* Berlin: Technik, 1987.
[2] S. F. Bocklisch and M. Päßler, *Fuzzy Control: Theory and Practice.* Springer Verlag, 2000, ch. Fuzzy time series analysis, pp. 331–345.
[3] S. F. Bocklisch and A.-J. Hempel, "A conception for networks of fuzzy pattern classifiers," in *Proceedings 13th East West Fuzzy Colloquium*, Zittau, 2006.
[4] M. Päßler, "Mehrdimensionale Zeitreihenmodellierung und -prognose mittels Fuzzy Pattern Modellen," Diploma thesis, Chemnitz University of Technology, 1998.
[5] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, "The UCR time series classification/clustering homepage," http://www.cs.ucr.edu/~eamonn/time_series_data/, 2006.
[6] F. Steimann, "The interpretation of time-varying data with DIAMON-1," *Artificial Intelligence in Medicine*, vol. 8, no. 4, pp. 343–357, 1996.
[7] G. Herbst and S. F. Bocklisch, "A hierarchical fuzzy approach to temporal pattern recognition," in *6th EUROSIM Congress on Modelling and Simulation*, B. Zupančič, R. Karba, and S. Blažič, Eds., 2007.
[8] D. Dubois and H. Prade, "Fuzzy numbers: An overview," in *Analysis of Fuzzy Information, Vol.I: Mathematics and Logic*, J. C. Bezdek, Ed. CRC Press, Boca Raton, FL, 1987, pp. 3–39.
[9] C. Bettini, C. E. Dyreson, W. S. Evans, and R. T. Snodgrass, "A glossary of time granularity concepts," *Lecture Notes in Computer Science*, vol. 1399, pp. 406–413, 1998.
[10] W.-J. Lee, J.-Y. Jiang, and S.-J. Lee, "Mining fuzzy periodic association rules," *Data and Knowledge Engineering*, vol. 65, no. 3, pp. 442–462, 2008.