Time Series Classification
with Linguistic Summaries

Katarzyna Kaczmarek and Olgierd Hryniewicz

Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447 Warsaw, Poland

Abstract

Soft computing techniques may provide various forms of human-consistent summaries about large time series databases, e.g., linguistic summaries, frequent patterns, fuzzy IF-THEN rules. Within this research, we focus on linguistic summaries constructed as linguistically quantified propositions, that may be exemplified by *'Among all increasing trends, most are short'*. We pose the question whether such imprecise results of summarization may successfully support the classification of time series data. Within the proposed approach, we classify a vector of linguistic summaries instead of classifying crisp time series. The approach is illustrated with experiments on artificial and benchmark real-life time series datasets. It turns out to be very promising for the classification of autoregressive time series by the probabilistic models.

Keywords: time series analysis, classification, linguistic summaries

1. INTRODUCTION

'On most nights the resident had a high level of restlessness' cf. Wilbik and Keller [1] or *'Decreasing windspeed segment occurs after highly increasing air pressure segment'* cf. Höppner et al. [2] are only some examples of data mining results. The data mining and knowledge discovery methods aim at discovering implicit, easy for interpretation and potentially useful knowledge to describe raw data, also time series datasets. The literature is extensive, and such results have been applied mostly for the descriptive or analytical purposes, e.g., [3, 4, 5, 6, 7], but also for classification [2] or decision making and prediction [8, 9]. Nonetheless, the potential for applications of the imprecise data mining results in the latter contexts seems to be much broader. We pose the question whether linguistic summaries [10, 11, 3] may successfully support the classification of time series.

Within this contribution, time series are represented as a vector of linguistic summaries, and we verify whether such representation is useful for the classification problem. Proper representation is a very important aspect within the data mining research [12]. In [9], the authors introduce solution based on the Perfilieva's F-transform and fuzzy relations. Rakthanmanon et al. [13] show that under Dynamic Time Warping (DTW) one can search large datasets much quickly than the current state-of-the-art Euclidean distance search algorithms. However, the main advantage of the proposed linguistic representation of time series is its human-consistency.

This paper is a continuation of our previous works which deal with the problem of supporting time series analysis and forecasting with granular computing [14, 15, 16]. The proposed approach is modification of part of the algorithm described in [16] which has been dedicated to short time series. The proposed method is also in line with the granular computing perspective as introduced in [14]. However, now the focus lays on the classification problem, not forecasting. Furthermore, the information granules considered in this paper are limited to linguistic summaries, and the calculations of the quality measures are performed automatically.

The main objective of this paper is to introduce method for time series classification using the summarization results. We represent time series by linguistic summaries due to the Yager's protoforms [17] modeled with the use of fuzzy logic. Then, the vector with respective degrees of truth is used to classify time series. The proposed *Classification with Linguistic Summaries* (CLS) method assumes employing techniques from the fuzzy set theory and the data mining field (segmentation, summarization, supervised learning).

The comparative analysis of the classification performance is presented on the simulational case study and the benchmark dataset from the UCR Repository [18]. In terms of classification accuracy, the experiments show that for the autoregressive datasets, the proposed approach significantly outperforms traditional methods, and for the real-life data, the accuracy is average among other methods. One of the main advantages of the proposed method is its human-consistency. We conclude that the linguistic summaries bring additional value and may support the classification of time series.

The structure of this paper is as follows. Next Chapter explains the discovery of linguistic summaries from the time series datasets. In Chapter 3, the proposed method for classification with linguistic summaries is presented. The numerical results of the experiments are gathered in Chapter 4. This paper concludes with general remarks and further research opportunities in Chapter 5.

2. DISCOVERY OF LINGUISTIC SUMMARIES

The data mining tools enable generation of various semantic and lexico-grammar structures summarizing large datasets. To formally describe the quantified sentences, we adapt the classic calculus of linguistically quantified propositions and the concept of the protoform in the sense of Yager [17] developed by Kacprzyk et al. [19, 3, 20].

2.1. Basic definitions

Let us now recall that a discrete time series is a sequence $y = \{y_i\}_{i=1}^m$ of observations measured at successive $\{1, ..., m\}$ moments and at uniform time intervals. Linguistic summaries describe general facts about evolution of time series with quasi natural language. The formal definitions are as follows.

Linguistic summary [17]

Let $O = \{o_1, o_2..., o_b\}$ denote a finite set of objects (e.g., trends of sequence data) in a considered domain. The properties of objects are measured by observables and are called attributes. Let $A = \{a_1, a_2..., a_r\}$ denote a finite set of attributes (e.g., dynamics of change, duration), and $S = \{s_1, s_2..., s_l\}$ is a finite set of imprecise labels for attributes (e.g., *quickly increasing, short*). The protoform-based linguistic summary

$$LS : Q \ o \ are \ P \tag{1}$$

consists of a quantity in agreement Q (quantifier like e.g., *most, among all*), summarizer P (attribute together with an imprecise label) about objects $o \in O$ and a measure T of validity or truth of the summary.

A **short protoform** takes the form: *Among all objects, Q are P*, and may be exemplified by the following sentence: *Among all trends, most are short.*

An **extended protoform** takes the following form: *Among all R objects, Q are P*, where R is a qualifier. The extended protoform may be exemplified by the following sentence:*Among all increasing trends, most are short.*

There have been proposed different measures to evaluate the quality of linguistic summaries, see [11]. One of the first, and still most common in practice, is the degree of truth (validity).

Degree of truth [21]

$$T(LS) = \mu_Q \left(\frac{\sum_{i=1}^n (\mu_R(y_i) \wedge \mu_P(y_i))}{\sum_{i=1}^n \mu_R(y_i)} \right) \tag{2}$$

where $\mu_R, \mu_P, \mu_Q : \mathbb{R} \to [0, 1]$ are the membership functions of fuzzy sets representing the qualifier R, summarizer P and quantifier Q, respectively.

The fuzzy set theory and linguistic variables [22] are adapted to handle the data imprecision. The interpretation for imprecise labels about trends and quantifiers for the set of linguistic summaries are modeled as the fuzzy trapezoidal numbers.

2.2. Segmentation

The segmentation methods transform the signal or the series of data points into a series of labeled segments. The review of the segmentation methods supported with the empirical study and the critical discussion is presented in Keogh et al. [23].

One of the simplest approach to the segmentation is the piecewise linear approximation, which assumes that the time series is approximated by the line segments. Within this research, the Sklansky and Gonzales' broken-line algorithm [24] based on the idea of a sliding window (on-line) is adapted. Successive segments are constructed forming a broken-line.

The adapted algorithm creates the segments (trends) based on the past observations, and then verifies if the new point fits the segment. Let (x_i, y_i) denote a pair (i, y_i) from time series $\{y_i\}_{i=1}^m$. The new point (x_i, y_i) is added to the existing segment if the distance d between the new line and the previous points is smaller than the user-defined threshold value $\epsilon \in [0, 1]$, and d is defined as follows:

$$\frac{|(y_l - y_f)x_i - (x_l - x_f)y_i + y_f x_l - y_l x_f|}{\sqrt{(y_l - y_f)^2 + (x_l - x_f)^2}} \tag{3}$$

where (x_f, y_f) and (x_l, y_l) are, respectively, the first and the last point of the current segment.

Having segmented the whole time series, labels from S are assigned for each segment. The sliding window width needs to be defined a priori. For further details and the alternative modifications of the segment construction within the online algorithms, see Wilbik [25].

2.3. Summarization

Discovering linguistic summaries is approached by the full search algorithm (for all attributes and their labels) modified by user-defined thresholds values to limit the search space due to the Trend Analysis System [20]. In short, the algorithm starts from a loop generating simple summaries for all attributes' values from S. The degrees of truth and focus are used for setting up thresholds. The degree of focus is calculated as follows.

Degree of focus [26]

$$d_{foc}(LS) = \frac{1}{n} \sum_{i=1}^n \mu_R(y_i) \tag{4}$$

where $\mu_R : \mathbb{R} \to [0, 1]$ is the membership function of fuzzy set representing the qualifier R.

The thresholds thr_T and thr_{foc} for the minimal accepted degrees of truth and focus of the linguistic summary need to be defined a priori. If the degree of truth of the simple summary satisfies the threshold thr_T, then the extended summaries are approached. Next, depending on the degree of focus, the linguistic summary is accepted and the search is continued or the tree branches are cut off.

3. CLASSIFICATION WITH LINGUISTIC SUMMARIES

The time series classification task is to map the time series to one of the predefined classes, two or more.

In [27], Berthold and Hand provide a comprehensive review of the supervised learning methods and the classification accuracy measures for the time series data. There could be distinguished three main approaches to the time series classification task:

- **feature-based**: Time series are represented using a set of derived properties or features (e.g., mean, variance), and thereby the temporal problem is transformed to a static one and may be resolved with one of the methods for the general problem of data classification. For example, in [28], the authors propose an approach to extract thousands of interpretable features from the time series, whereas in [29], Nanopoulos et al. use only the statistical features for time series classification.
- **sequence-distance-based**: There needs to be defined the distance function (e.g., the simple Euclidean distance) which measures the similarity between a pair of sequences, and then determines the quality of the classification.
- **model-based**: The classification of the sequences is based on a predefined model like the Gaussian mixture model, Hidden Markov Model, etc.

The proposed *Classification with Linguistic Summaries* (CLS) method realizes the feature-based approach to time series classification. First, time series is transformed into a feature vector consisting of linguistic summaries, and then, the conventional classification method is applied to classify it.

Among many approaches to the general problem of data classification, the Support Vector Machine (SVM) introduced by Vapnik [30] and Cortes and Vapnik [31] are regarded as a computationally efficient way of learning separating hyperplanes in a high dimensional feature space [27]. Another efficient method regarded as difficult to beat in terms of classification accuracy in the study by [32] is one-nearest-neighbor with Euclidean distance.

Algorithm 1 depicts a high-level description of the proposed *Classification with Linguistic Summaries* (CLS) method dedicated to time series. The output of the algorithm is the classifier CL learned on the labeled training data $[Y^s, C^s]$.

The input for the algorithm are labeled time series, precisely, s discrete time series Y^s and the corresponding s-vector with C^s labels (classes) describing the time series. The input for the algorithm are also imprecise labels S to be processed during the time series abstraction and the discovering of linguistic summaries. For the clarity reasons, it is presented that the considered series (Y_m^s and Tr_k^s) have the same lengths (m and k, respectively). Nonetheless, the same length does not influence the overall procedure and the algorithm is valid also for series of different lengths.

Algorithm 1 Time Series Classification with Linguistic Summaries (CLS)

Input: Y^s, C^s, S
Output: CL

1: $S = $ define_imprecise_labels
2: $min_a = $ define_minimal_accuracy
3: $i = 1$
4: $Tr^s, Li^s, V^s = \emptyset$
5: **for** each $Y_i \in Y^s$ **do**
6:      $Tr_i = $ segmentation(Y_i)
7:      $LI_i = $ summarization(Tr_i)
8:      $V_i = $ quality_evaluation(LI_i)
9: $CL = $ data_classification(V^s, C^s)
10: **if** validation_accuracy(V^s, C^s) $\geq min_a$ **then**
11:      return CL
12: **else**
13:      results not validated; return to beginning

The proposed algorithm starts from creating fuzzy sets for the imprecise labels (Line #1) and from establishing the threshold for the minimal accuracy of the classifier (Line #2) . Then, each time series from Y^s is segmented, summarized and its linguistic summaries are evaluated with the degree of truth (Line #5-8).

The resulting vector V^s consists of the degrees of truth about the linguistic summaries and it is the feature vector for the static data classification task. The features V^s and its labels C^s are classified with conventional data classification algorithm (Line #9).

To solve the classification problem on the feature vector with linguistic summaries, we consider the SVM algorithm. For details on the kernel functions, the optimal and soft margin classifiers and other important aspects of the Support Vector Networks, see e.g., [31, 27]. Alternatively, the k-Nearest Neighbors algorithm (or kNN for short) introduced by [33], which is also a well-known method for classification, is used for the comparative purposes.

Finally, we validate the approximated accuracy of the learned classifier (Line #10-13) using the 2-fold cross validation which is one of the simplest variation of k-fold cross-validation and tough, useful in practice. Each data point is used for both training and validation on each fold. If the approximated accuracy of the classifier exceeds the desired threshold, the algorithm ends. Otherwise, the procedure is repeated.

4. NUMERICAL RESULTS

The experimental results are presented for both, the artificial and the real-life time series datasets. They illustrate the classification performance of the proposed *Classification with Linguistic Summaries* (CLS) approach.

The accuracy of the proposed method is compared with selected benchmark methods (SVM, kNN). The performance of the classifiers is reported

Table 1: Classes (labels) for time series from the autoregressive datasets.

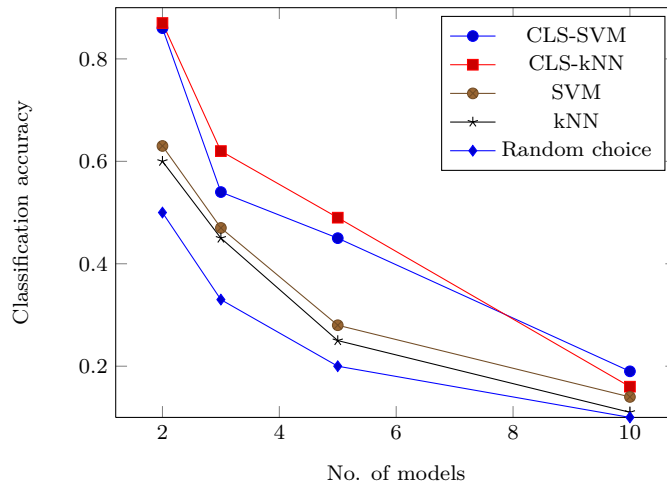| Dataset	Classes (Labels)
2-class	***positive, negative*** autocorrelation
3-class	*positive, neutral, negative* autocorrelation
5-class	*strong positive, moderate positive, neutral, moderate negative, strong negative* autocorrelation
10-class	*very strong pos., strong pos., moderate pos., small pos., neutral, small neg., moderate neg., strong neg., very strong neg.* autocorrelation

Figure 1: Classification accuracy for various datasets: scenarios of 2, 3, 5 and 10 models (classes) due to the proposed CLS-SVM and CLS-kNN methods, the traditional SVM and kNN.

Table 2: Average classification accuracy (std dev) due to 2-fold cross validation.

	Classification method			
	CLS-SVM	**CLS-kNN**	**SVM**	**kNN**
2-class	**0.86** (0.02)	**0.87** (0.03)	0.63 (0.07)	**0.60** (0.00)
3-class	0.54 (0.00)	0.62 (0.02)	0.47 (0.01)	0.45 (0.03)
5-class	0.45 (0.03)	0.49 (0.01)	0.28 (0.02)	0.25 (0.03)
10-class	**0.19** (0.01)	0.16 (0.02)	**0.14** (0.04)	0.11 (0.01)

using the 2-fold cross validation and *precision* calculated as the number of correct positive predictions to the number of positive examples.

The following attributes and labels defining trends are considered to build up the feature vector with linguistic summaries:

- duration, e.g., *short, medium, long*;

- dynamics, e.g., *increasing, constant, decreasing*;

- variability, e.g., *low, moderate, high.*

The quantifier $Q = \{most\}$ is applied and the degree of truth is used to evaluate the quality of summaries.

4.1. Artificial Datasets

Time series considered in experiments are labeled depending on the probabilistic model and its parameters that produced the time series. This approach is in line with the Box-Jenkins methodology for time series analysis and forecasting [34].

We examine the classification accuracy for 4 different datasets that refer to the scenarios of accordingly 2, 3, 5 and 10 classes, depending on the value of the autoregressive coefficients as presented in Table 1. As to give an example, time series from the dataset called **2-class** are labeled as either *positive* or *negative* according to the value of the autocorrelation coefficients of the probabilistic model that produced the time series.

In this analysis we compare 2 variants of the proposed CLS approach (using either the SVM or the kNN algorithm) and 2 benchmark algorithms: SVM [31] and kNN [33]. Figure 1 shows the classification accuracy. It is observed that the proposed CLS method outperforms competitive methods for all datasets. Table 2 presents further details concerning the average accuracy and its standard deviation.

As demonstrated by results in Table 2, for the **2-class** dataset the classification accuracy is highest for the proposed CLS-kNN method and amounts to 0.87, whereas for the standard kNN it amounts only

Table 3: Exemplary linguistic summaries about the time series from the *Coffee* training dataset.

Linguistic Summary Description	Class 1	Class 2
Among all medium and moderate trends, most are increasing	**0.10**	**0.63**
Among all medium and constant trends, most are moderate	0.76	0.55
Among all medium trends, most are moderate	0.77	0.54
Among all short and high trends, most are increasing	0.38	0.70
Among all medium trends, most are low	0.37	0.38
Among all trends, most are short	**1.00**	**1.00**

to 0.60. When considering the **10-class** dataset, the highest classification accuracy of 0.19 is achieved by the CLS-SVM method, whereas for the standard SVM it amounts to 0.14.

We conclude that for the considered autoregressive datasets, the results for the referenced SVM and kNN methods are consequently significantly lower than the accuracy of the proposed CLS-SVM and CLS-kNN methods. The considered scenarios CLS-SVM and CLS-kNN provide similar results, therefore in the remaining experiments presented in this paper, we limit the consideration to the SVM scenario.

To conclude, the results on the autoregressive datasets are very promising and reflect intuitions that the linguistic summaries are an interesting tool for classifying time series.

4.2. Illustrative example

To illustrate the performance of the approach in the real-life context, classification details of an exemplary time series from the UCR Repository [18] are described. We consider dataset called *Coffee* by Tony Bagnall. It is a 2-class problem, the length of each time series is 286 and there is a total of 56 different time series.

Table 3 demonstrates an exemplary set of evaluated linguistic summaries for time series of both classes from the considered dataset and the average (within training samples from this class) degree of truth.

As observed in Table 3, some types of linguistic summaries are a good discriminant for the classification analysis. For example, the average degree of truth of the linguistic summary *'Among all medium and moderate trends, most are increasing'* of times series from the 1st and 2nd class are 0.10 and 0.63, respectively. At the same time, there exist summaries, e.g., *'Among all trends, most are short'* that do not bring value to the classification in this context because the degree of truth is 1 for both classes.

Table 4 shows accuracy of the proposed method and some competitive common classification approaches: the traditional SVN and kNN, the Decision Tree algorithm C4_5 and the Bayes' nets (NB).

It is observed in Table 4 that the proposed method CLS achieves quite average accuracy of 71.4% being close to the traditional kNN method, that achieves 75.0%. Furthermore, the proposed

method outperforms some benchmarks (Decision Tree and the Bayes' nets). Nonetheless, the best precision of 96.4% on the considered dataset is achieved by the SVM algorithm. For further details on the benchmark methods, see [18].

Table 4: Classification accuracy due to precision on the *Coffee* testing dataset for the proposed CLS and some benchmark methods.

Method	Classification accuracy
SVM	96.4 %
kNN	75.0 %
CLS	**71.4** %
NB	67.9 %
C4	57.1 %

We conclude that the accuracy of the proposed method for the considered real-life dataset is still quite average among other methods, nonetheless the processed features (linguistic summaries) are easy to interpret by user, which may be an important advantage in various practical contexts.

It should be noted, that the obtained results are of preliminary character, and in future should be confirmed by next experiments performed on an extensive set of benchmark datasets to prove as an complete empirical validation. Further forms of linguistic summaries may also be investigated to improve the description of time series.

5. CONCLUSION AND FUTURE WORK

We have presented the *Classification with Linguistic Summaries* (CLS) method that processes imprecise summaries created with the use of linguistically quantified propositions about the evolution of time series. Discrete time series are segmented and summarized. Then, the linguistic summaries are classified with the conventional data classification methods (the Support Vector Machines). The classification of time series with the use of summarization results is an innovative approach.

The experiments show that the proposed approach brings very promising results for the classification of the autoregressive datasets. For the considered real-life data, the accuracy is average among other methods. However, important advantage of the proposed method is its human-consistency.

The main challenges of the proposed solution are the proper selection of the attributes with imprecise labels and the considered types of linguistic summaries. Future research assumes next experimental studies to verify in which contexts linguistic summaries are outstanding to represent the real-life time series data.

References

[1] A. Wilbik and J. Keller. A distance metric for a space of linguistic summaries. *Fuzzy Sets and Systems*, 208:79–94, 2012.

[2] F. Höppner, S. Peter, and M. Berthold. Enriching multivariate temporal patterns with context information to support classification. *Computational Intelligence in Intelligent Data Analysis. Studies in Computational Intelligence*, 445:195–206, 2013.

[3] J. Kacprzyk and A. Wilbik. Using fuzzy linguistic summaries for the comparison of time series: an application to the analysis of investment fund quotations. In *IFSA/EUSFLAT Conf. 2009*, pages 1321 – 1326, 2009.

[4] L. Naoum, G. Raschia, and N. Mouaddib. Quering fuzzy summaries of databases: unary operators and their properties. In *Proceedings of the Joint 4th Conference of the European Society for Fuzzy Logic and Technology and the 11th Rencontres Francophones sur la Logique Floue et ses Applications, Barcelona, Spain, September*, pages 1194–1199, 2005.

[5] P. Conde-Clemente, J. Alonso, and G. Trivino. Interpretable fuzzy system allowing to be framed in a profile photo through linguistic expressions. In *Proc. of 8th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)*, pages 463–468, 2013.

[6] S. Kempe, J. Hipp, C. Lanquillon, and R. Kruse. Mining frequent temporal patterns in interval sequences. *Fuzziness and Knowledge-Based Systems in International Journal of Uncertainty*, 16 (5):645–661, 2008.

[7] F. Mörchen, I. Batal, D. Fradkin, J. Harrison, and M. Hauskrecht. Mining recent temporal patterns for event detection in multivariate time series data. *KDD*, pages 280–288, 2012.

[8] M. Burda, Martin Štěpnička, and Lenka Štěpničková. Fuzzy rule-based ensamble for time series prediction: Progresses with associations mining. In *Strengthening Links Between Data Analysis and Soft Computing*, volume 315, pages 261–271. Springer, 2014.

[9] N. Yarushkina, I. Perfilieva, T. Afanasieva, A. Igonin, A. Romanov, and V. Shishkina. Time series processing and forecasting using soft computing tools. In *RSFDGrC'11 Proceedings of the 13th international conference on Rough sets, fuzzy sets, data mining and granular computing*, pages 155–162. Springer-Verlag Berlin, Heidelberg, 2011.

[10] J. Kacprzyk and S. Zadrożny. Protoforms of linguistic data summaries: towards more general natural-language-based data mining tools. *Soft Computing Systems, IOS Press, Amsterdam*, pages 417–425, 2002.

[11] J. Kacprzyk and S. Zadrożny. Linguistic database summaries and their protoforms: towards natural language based knowledge discovery tools. *Information Sciences*, 173:281–304, 2005.

[12] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[13] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh. Addressing big data time series: Mining trillions of time series subsequences under dynamic time warping. *Journal ACM Transactions on Knowledge Discovery from Data (TKDD) - Special Issue on ACM SIGKDD*, 7(3), 2013.

[14] O. Hryniewicz and K. Kaczmarek. Bayesian analysis of time series using granular computing approach. *Applied Soft Computing*, 2015.

[15] K. Kaczmarek and O. Hryniewicz. Linguistic knowledge about temporal data in bayesian linear regression model to support forecasting of time series. In *Proc. of Federated Conference on Computer Science and Information Systems*, pages 655 – 658, 2013.

[16] O. Hryniewicz and K. Kaczmarek. Forecasting short time series with the bayesian autoregression and the soft computing prior information. In *Strengthening Links Between Data Analysis and Soft Computing*, volume 315, pages 79–86. Springer, 2014.

[17] R. Yager. A new approach to the summarization of data. *Information Science*, 28 (1):69–86, 1982.

[18] E. Keogh, Q. Zhu, B. Hu, Y. Hao, X. Xi, L. Wei, and C. Ratanamahatana. The ucr time series classification/clustering page: www.cs.ucr.edu/~eamonn/timeseriesdata/. 2011.

[19] J. Kacprzyk. Linguistic summarization of time series using a fuzzy quantifier driven aggre-

gation. *Fuzzy Sets Syst*, 159 (12):1485–1499, 2008.

[20] J. Kacprzyk, A. Wilbik, A. Partyka, and A. Ziółkowski. *Trend Analysis System*. Systems Research Institute, Polish Academy of Sciences, Warsaw, 2011.

[21] L. A. Zadeh. A computational approach to fuzzy quantifiers in natural languages. *Computers and Maths with Applications*, pages 149–184, 1983.

[22] L. A. Zadeh. The concept of linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3):199–249, 1975.

[23] E. Keogh, S. Chu, D. Hart, and M. Pazzani. *Segmenting time series: A survey and novel*. World Scientific, 2004.

[24] J. Sklansky and V. Gonzalez. Fast polygonal approximation of digitized curves. *Pattern Recognition*, 12(5):327–331, 1980.

[25] A. Wilbik. *Linguistic summaries of time series using fuzzy sets and their application for performance analysis of mutual funds*. PhD thesis, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland, 2010.

[26] J. Kacprzyk and A. Wilbik. Towards an efficient generation of linguistic summaries of time series using a degree of focus. In *Proc. of the 28th North American Fuzzy Information Processing Society Annual Conference*, pages 1–6, 2009.

[27] M. Berthold and D. Hand. *Intelligent data analysis. An Introduction*. Springer, 2007.

[28] B. Fulcher and N. Jones. Highly comparative feature-based time-series classification. *IEEE Transactions on Knowledge and Data Engineering*, pages 3026–3037, 2014.

[29] A. Nanopoulos, R. Alcock, and Y. Manolopoulos. Feature-based classification of time-series data. *International Journal of Computer Research*, pages 49–61, 2001.

[30] V. Vapnik, B. Boser, and I. Guyon. A training algorithm for optimal margin classifiers. In *In D. Haussler, editor, 5th Annual AGM Workshop on COLT, Pittsburgh. ACM Press*, pages 144–152, 1992.

[31] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[32] E. Keogh and S. Kasetty. On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Min. Knowl. Discov.*, 7(4):349–371, 2003.

[33] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, pages 21–27, 1967.

[34] G. Box, G. Jenkins, and G. Reinsel. *Time Series Analysis, 4th Edition*. Wiley, 2008.