International Journal of Approximate Reasoning 78 (2016) 192–209

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Mining information from time series in the form of sentences of natural language

Vilém Novák

University of Ostrava, Institute for Research and Applications of Fuzzy Modeling, NSC IT4Innovations, 30. dubna 22, 701 03 Ostrava 1, Czech Republic

ARTICLE INFO

Article history:
Received 30 September 2015
Received in revised form 5 July 2016
Accepted 12 July 2016
Available online 18 July 2016

Keywords:
Fuzzy natural logic
Evaluative linguistic expressions
Intermediate quantifiers
Fuzzy transform
Fuzzy type theory
Tectogrammatical tree

ABSTRACT

The goal of this paper is to provide a more detailed explanation of the principles how special formulas that characterize properties of trend of time series can be formed and how they are interpreted. Then we show how these formulas can be used in a tectogrammatical tree that construes special sentences of natural language, using which information on behavior of time series is provided. We also outline the principles of mining this information. The last part is devoted to application of the theory of intermediate quantifiers to mining summarized information on time series also in sentences of natural language.

1. Introduction

Mining information from time series is an interesting problem that is extensively elaborated. A nice overview of various tasks raised in connection with time series and some of the possible methods for solving them is presented by Fu in [7] where one can find also a lot of references to special methods. One of the tasks is generation of automatic comments in natural language. This problem is, in fact, wider because time series is only one possible sort of data. This was initiated almost 60 years ago by Luhn [17] (see also [23] and the citations therein) and since then, many authors contributed to it.

Summarization of knowledge about time series is a narrower problem. In the literature, one can find methods for finding interesting patters in time series (cf. [41]). In [15] Kobayashi and Okumura describe method for generation of comments in natural language that can serve as recommendation for stock broker about what happened with stock price trends during the day. Kacprzyk and Wilbik in [11] present a method for finding sequences of monotonous behavior and on the basis of them characterize similarity of time series.

A special task is summarization of knowledge about time series using quantifiers of natural language. This task was solved using techniques of fuzzy set theory especially by Kacprzyk, Wilbik, Zadrożny [12,13] and Castillo-Ortega, Marín, Sánchez (see, e.g., [2,3]). The authors suggested various heuristic methods for mining information on the basis of which proper natural language expressions can be generated. In this paper, we suggest alternative methods for this task. First, we apply techniques based on the theory of *fuzzy transform* that makes it possible to analyze time series. Then, we apply the theory of *fuzzy natural logic* (FNL) that provides a formal model of semantics of special expressions of natural language as

E-mail address: Vilem.Novak@osu.cz.

http://dx.doi.org/10.1016/j.ijar.2016.07.006
0888-613X/

well as schemes for reasoning with them. We first form special formulas of FNL and then obtain expressions of natural language by interpretation of them.

Ramos-Soto et al. in [39] analyze two variants how linguistic descriptions of data can be produced: using a standard natural language generation approach (cf. [40]) or using template-based NLG. Both approaches have their pros and cons. The former is often taken as superior over the latter because of its ability to generate richer structures that are closer to the way how people speak. However, as pointed out by van Deemter et al. in [43] this distinction is now still more blurred because both kinds of systems are developing and even the NLG systems have to be simplified because of extreme complexity of natural language. Therefore, they are not that far from templates.

In this paper, we will suggest methods for generation of natural language expressions characterizing time series. Our solution is essentially template-based. Its merit consists in the fact that we stem from the formal model of semantics developed within FNL.

We address three tasks: (a) characterization of trend in natural language, (b) finding intervals of definite character of trend, and (c) summarization of the characteristics of time series using intermediate quantifiers. A special attention is payed especially to task (c), because tasks (a) and (b) were in detail elaborated in the previous papers [30,32,33].

The structure of the paper is as follows. In the next section we briefly introduce main mathematical concepts, namely the fuzzy type theory (FTT), two theories belonging to FNL, the fuzzy transform and also precisely specify the concept of time series. The main contribution of this paper is in Section 3. We first introduce special formulas of FTT representing properties of time series, overview some already published methods and focus on mining linguistic information on the course of time series. The subsequent subsection is focused on mining summarized information using the concept of intermediate quantifier. In conclusion, we summarize the results of this paper and also outline future research that, among others, will be focused on utilization of results in the theory of generalized syllogistic reasoning.

2. Preliminaries

The theoretical frame for the methods developed in this paper is formed by *fuzzy natural logic* (FNL) and *fuzzy (F-)transform*. Recall that the former is a formal logical theory that consists of (a) a formal theory of evaluative linguistic expressions (see [27]), (b) a formal theory of fuzzy IF–THEN rules and approximate reasoning (see [26,31]) and (c) a formal theory of intermediate and generalized fuzzy quantifiers (see [6,20,22,28]). FNL is a mathematical theory developed using formalism of the *fuzzy type theory* (FTT) (see [25,29]). Its formal language is extension of the lambda calculus.

By a fuzzy set we understand in this paper a function $A : U \longrightarrow E$ where E is a set of truth values. By $\mathscr{F}(U)$, we denote the (crisp) *set* of all fuzzy sets on the universe U. Hence, the formula $A \in \mathscr{F}(U)$ is either true or false. If A is a fuzzy set on U then we often write $A \underset{\sim}{\subset} U$. The symbol $A(u)\big/u$ denotes a fuzzy singleton where $u \in U$ is an element and $A(u)$ is its membership degree.

2.1. Fuzzy type theory

The chosen version of FTT is the Łukasiewicz one (Ł-FTT). Some of its main concepts are briefly overviewed below.

Truth values are supposed to form the standard Łukasiewicz MV$_\Delta$-algebra

$$\mathscr{L} = \langle [0,1], \vee, \wedge, \otimes, \rightarrow, 0, 1, \Delta\rangle \tag{1}$$

where Δ is a unary operation such that $\Delta(a) = 1$, if $a = 1$ and $\Delta(a) = 0$ otherwise.

Each formula in FTT is assigned a type. This can be understood as an index encoding the kind of objects that are represented by the given formula. The primitive types are o representing truth values and ϵ representing primary objects. We will also consider the type τ which will represent both real numbers as well as time moments. More complex types are formed by concatenation of simpler ones.

The *language* J of Ł-FTT consists of variables $x_\alpha, \ldots$, special constants $c_\alpha, \ldots$ ($\alpha \in Types$), the symbol λ, and brackets. The connectives (which are special constants) are *fuzzy equality/equivalence* $\equiv$, *conjunction* $\wedge$, *implication* $\Rightarrow$, *negation* $\neg$, *Łukasiewicz conjunction* $\boldsymbol{\&}$, *disjunction* $\vee$, and *delta* $\boldsymbol{\Delta}$. As usual, we will write fuzzy equality between formulas of type α as $(A_\alpha \equiv B_\alpha)$, instead of $(\equiv B_\alpha)A_\alpha$. Note that this is a formula of type o (truth value).

If A is a formula or a variable of type α, then we write A_α.[1] Hence, if $\alpha \neq \beta$ then A_α and A_β are different formulas. Sometimes we will also write $A \in Form_\alpha$ to stress that A is a formula of type α. If the type of a formula is clear from the context, we will omit it to simplify reading of the formulas. Recall that $\lambda x_\alpha\, A_\beta$ is a formula of type $\beta\alpha$. If $A_{\beta\alpha}$ is a formula of type $\beta\alpha$ and B_α is a formula then $A_{\beta\alpha}B_\alpha$ is a formula of type β. We will omit brackets wherever possible and use them only to clarify reading of more complex formulas.

Semantics of FTT is defined on the basis of a *general frame*

$$\mathscr{M} = \langle (M_\alpha, =_\alpha)_{\alpha \in Types}, \mathscr{E}_\Delta\rangle \tag{2}$$

[1] In FTT we do not distinguish between terms and formulas and call all them just formulas. Various authors call them alternatively λ-terms.