



12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS
2016, 29-30 August 2016, Vienna, Austria

Linguistic resumes in software engineering: the case of trend summarization in mobile crash reporting systems

Konstantin Y. Degtiarev^{a,*}, Nikita V. Remnev^{a,b}

^a National Research University Higher School of Economics (HSE), Faculty of Computer Science, School of Software Engineering, 3 Kochnovsky
pass., 125319, Moscow, Russia

^b Ru-Beacon (Empatika company's project)

Abstract

The construction of time series linguistic summaries is a topic that draws attention of researchers for many years. The full-fledged software implementation (the pilot web-application) that supports the complete process of linguistic summarization of time series construction is presented in the paper. The program can be used in professional groups for discussions and rapid data analysis. Virtual mobile crash reporting system (MCRS) supplies the test input data used as an example.

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of the Organizing Committee of ICAFS 2016

Keywords: Time series; linguistic summary; software engineering; software development; summarizer; validity of linguistic summary; fuzzy set; linguistically quantified proposition; mobile crash reporting system; analysis of trends; graphical reports; web-application

1. Introduction

In the age of information technology's impetuous progress that influences daily human activity, comprehension, analysis and processing of intensive data flows assume ever greater importance. The power of such flows compels people to resort to the help of means to extract factual entities and construct summary based on domain-specific information brought to their notice via *organa sensuum*. The important fact of obtaining the information should be supplemented with a sole aspect of what we derive from it. Such findings may facilitate perceptibly different types

* Corresponding author. Tel.: +7-495-772-9590.

E-mail address: kdegdiarev@hse.ru

of routine tasks related to grasping the meaning and key points of problems, predicting the future, decision-making, and so forth. Summarization of data is ingeniously associated with the ability of humans “to *communicate* observations of the world in a useful and comprehensible manner” that is convenient for use by both individuals and companies^{13,14}.

According to Merriam-Webster dictionary, the word ‘**summary**’ is defined as “using few words to give the most important information about something”. The construction and use of such summaries accompany any kind of human activity, and thriving fields of IT and software engineering are not an exception. Multifarious human activity would not be possible without efforts aimed at the development of software to run on hardware platforms of mobile devices, desktop computers and servers. In the field of software engineering, data summarization plays significant and useful role in respect of construction of *units of communication*¹³ used in discussion of project’s details with heterogeneous groups of people embracing various types of stakeholders, developers, testers of software products.

It is natural that the data (information) in use are always connected with a time factor that has a significant impact on the conclusions derived on the basis of such information. Timeline has a great importance in information handling, since it sets up a base to explain association with events that may influence these changes. Definitely, available data can be visualized by graphs giving proper account to time stamps, if any. Graph’s granularity as a main drawback of such representation may conceal from view important, but quite short periods of changes that make the findings not complete. This fact lays the basis for the whole research area to deal with the construction of time series linguistic resume, bearing in mind that natural language is mainly “a system for describing perceptions, which are intrinsically imprecise”, therefore most of phrases and sentences formulated in natural language are fuzzy¹⁵.

Software development process is complex multiphase sequence of actions (with repetitive back-offs to previous steps, if needed) appreciably linked to human contacts and diverse descriptions in natural language. Problem analysis, preparation of specifications, planning of development steps and definition of software architecture can be designated as revealing examples. The output of development process is the software product, the quality of which can be good, endorsed by users, but generally not perfect. Development teams have to gather information about the state of the program after its deployment, problems that occur during the runtime, issues that cause failure (crash) – e.g. the number of crashes, backtraces of the process’s threads, usage of CPU resources, etc. Nowadays, the collecting and storage of crash reports are fully automated. The analysis of reports pursues the long-term object to obtain accurate interpretation and promptly fix the origin causing the crash. It seems reasonable to conjecture that the application of algorithms for constructing linguistic resume of time series may facilitate routine tasks of developers and influence positively the time management of software maintenance.

Crash reporting systems consist of two main modules, which ensure operation of the whole system – the first one is a built-in mobile application module to detect critical errors and to send error reports (codes, etc.) to server system. The latter being the second module of the system analyzes reports and presents processed information as graph(s). Systems per se are not engaged in further discussion – they are simply treated as ‘black boxes’ that receive errors as inputs and present information relating to errors in the graphical form. Analysis of such graphs can be considered as a key constituent of *exploratory data analysis* (EDA) aimed at visual revealing of patterns.

Graphical reports are rather attractive, but they suffer from evident shortcomings mentioned in passing above. The information concerning errors usually consolidates several features that cannot be displayed on the same graph discernibly. Users having little or no experience with software development can be in a predicament to interpret data, to understand data format as well as the meaning of constituent features of the error report shown to him (her).

The construction of short linguistic summary on the ground of obtained data in respect to crash experienced in practice seems to be clear and extremely convenient way out for the user of such system. The rest of the paper is organized as follows: at first (Section 2), selected core research studies on time series linguistic summarization (TSLS) and related topics are briefly discussed. Following basic publications by renowned scientists L. Zadeh, R. Yager, J. Kacprzyk and S. Zadrożny, Sections 3 and 4 cover some general ideas as well as distinctive features relating to the process of TSLS construction; the importance of summaries in various situations associated with the field of software engineering is also elucidated here. Section 5 is devoted to the description of the scope of pilot web-based application aimed at construction of linguistic summaries. Conclusion and final remarks are drawn in Section 6.

2. Linguistic Summarization. Related Work

One of the first papers entirely dedicated to the description of linguistic summary essence belongs to R. Yager¹. The paper presents the way to summarize data set in the form of linguistic values that can be quantified as fuzzy

Download English Version:

<https://daneshyari.com/en/article/4961614>

Download Persian Version:

<https://daneshyari.com/article/4961614>

[Daneshyari.com](https://daneshyari.com)