SumTime-Turbine: A Knowledge-Based System to Communicate Gas Turbine Time-Series Data

Jin Yu and Ehud Reiter and Jim Hunter and Somayajulu Sripada

Department of Computing Science
University of Aberdeen
Aberdeen, AB24 3UE, UK
{jyu, ereiter, jhunter, ssripada}@csd.abdn.ac.uk

Abstract: *SumTime-Turbine* produces textual summaries of archived time-series data from gas turbines. These summaries should help experts understand large data sets that cannot be visually presented in a single graphical display. *SumTime-Turbine* is based on pattern detection, knowledge-based temporal abstraction (KBTA), and natural language generation (NLG) technology. A prototype version of the system has been implemented and is currently being evaluated.

1 Introduction

In order to get the most out of gas turbines, *TIGER* [2] has been developed by Intelligent Application Ltd to continuously monitor and assess the condition of gas turbines. *TIGER* can collect and archive about 600 data points per second from more than 250 channels [1]. Since these archived data are potentially very valuable for supporting diagnosis, anomaly detection, and prediction, it's worthwhile to develop techniques and implement tools to help domain experts to mine these data. Currently, human examination of time series data is generally done either by direct inspection of the numerical values of the data (for small data sets), by graphical visualisation, or by statistical analyses. The volume of *TIGER* data is so huge that it's not feasible for domain engineers to go through the graphical displays looking for events of interest. A further possibility is the generation of textual summaries. So a knowledge-based system named *SumTime-Turbine* is being implemented to produce text summaries of these archived temporal data.

The value of *SumTime-Turbine* is that it could provide a good abstraction of the data in terms of event patterns and give a concise summary from these abstractions for engineers. So it could help engineers to formulate useful knowledge that is beneficial to fault detection and diagnosis in gas turbines.

The organisation of the remainder of this paper is as follows. Section 2 presents the architecture of *SumTime-Turbine*. Main functions and implementation of the prototype system of *SumTime-Turbine* are explained in Section 3. Evaluation methods on the system are introduced in Section 4. Section 5 compares our system with those of others and Section 6 gives future work in the prototype system.

2 Architecture of SumTime-Turbine

Based on knowledge acquisition sessions with human experts, we have discovered that the system should perform the tasks shown in Figure 1.

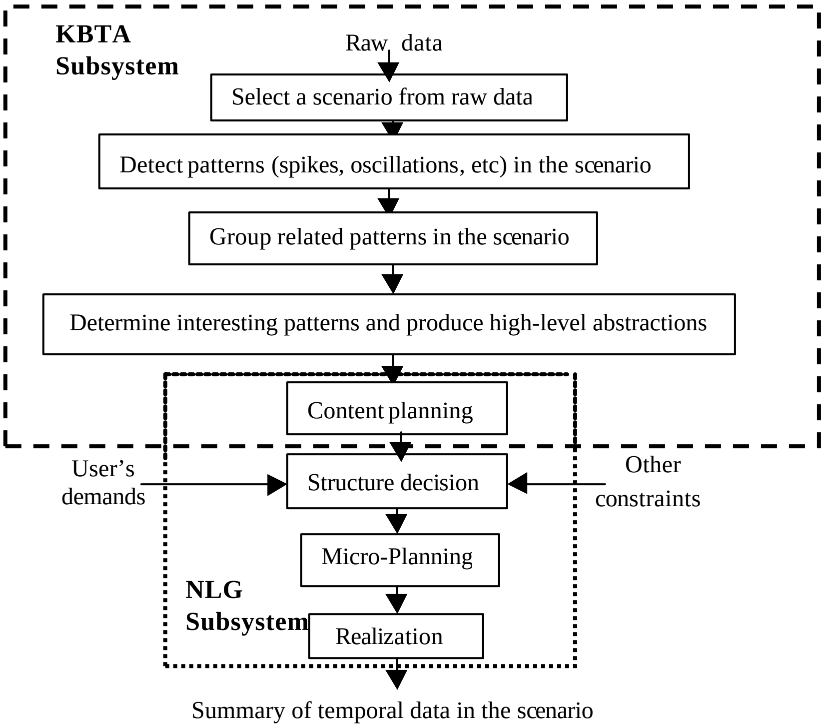

Fig. 1. Architecture of SumTime-Turbine

This KBS system could be implemented in two subsystems: KBTA and NLG, to carry out these tasks. These two subsystems form separate, reusable modules, which are connected through a content planning module (Fig. 1). The advantage of this architecture is that both the subsystems are reusable. At the same time, much information about natural language generation provided by the KBTA subsystem probably can be used for the reliable generation of summary in the NLG subsystem, thus improving the quality of contents, word usage, etc in the output summary.

3 Functions of the Prototype System of SumTime-Turbine

A prototype system of *SumTime-Turbine*, implemented based on the above architecture and techniques, includes eight analysis functions. (1) Primitive Pattern Recognition, (2) Primitive Pattern Description, (3) High Level Abstraction, (4) Summary, (5) Interesting Pattern Recognition, (6) Primitive Pattern Evaluation, (7) Interesting Pattern Evaluation, and (8) Summary Evaluation; plus two evaluation functions: (1) Non-Experts Marking Up Patterns, and (2) Experts Marking Up Patterns.

3.1 Primitive and Interesting Pattern Recognition

In the gas turbine domain, we have noticed that engineers are very interested in turbulence patterns that can be further classified into three primitive patterns: spikes, oscillations, and steps. Each of them has different special meanings for engineers when they investigate the raw data. A systematic method, including a turbulence-locator and a pattern-classifier, has been developed to automatically identify such patterns [8].

We have found that the process of primitive pattern recognition is domain-independent while that of determining which patterns are interesting is domain-dependent. Interesting pattern recognition applies domain knowledge to decide which primitive patterns are interesting. For example, in the gas turbine domain, spikes that occur simultaneously across all channels such as a set of spikes at 23:55:57 (Fig. 2) are interesting for domain engineers while small stand-alone spikes such as a spike at 23:59:02 in channel TNH (Fig. 2) are not interesting. In a sense, primitive patterns should include as many candidates for interesting patterns as possible.

3.2 Primitive Pattern Description

This function abstracts informatiton about patterns including: pattern names, start time of patterns, temporal length of patterns, size of patterns. Such information can be explained in linguistic format. The following is a linguistic description about primitive patterns occurring in channel FSGR in Figure 2.

Channel name: FSGR (Linguistic format)
Very big downward spike at 21:50:08, 21:51:36, 21:53:36, and 22:36:05.
Big erratic spike at 23:56:17.

3.3 High Level Abstraction.

This function is mainly based on KBTA method [4] [5]. Currently it carries out vertical and horizontal aggregation. Vertical aggregation is based on simultaneous check and horizontal aggregation involves joining nearby patterns to form a main set of patterns. The following is the results of high level abstracton on the sample data set.

Vertical aggregation (Linguistic format)
Spikes in all channels at 21:49:49, 21:51:31, 21:53:17, 23:55:57
Mostly spikes with some steps in all channels at 22:35:02
Horizontal aggregation (Linguistic format)
Spikes in all channels at 21:49:49 and 23:55:57
Mostly spikes with some steps in all channels at 22:35:02,

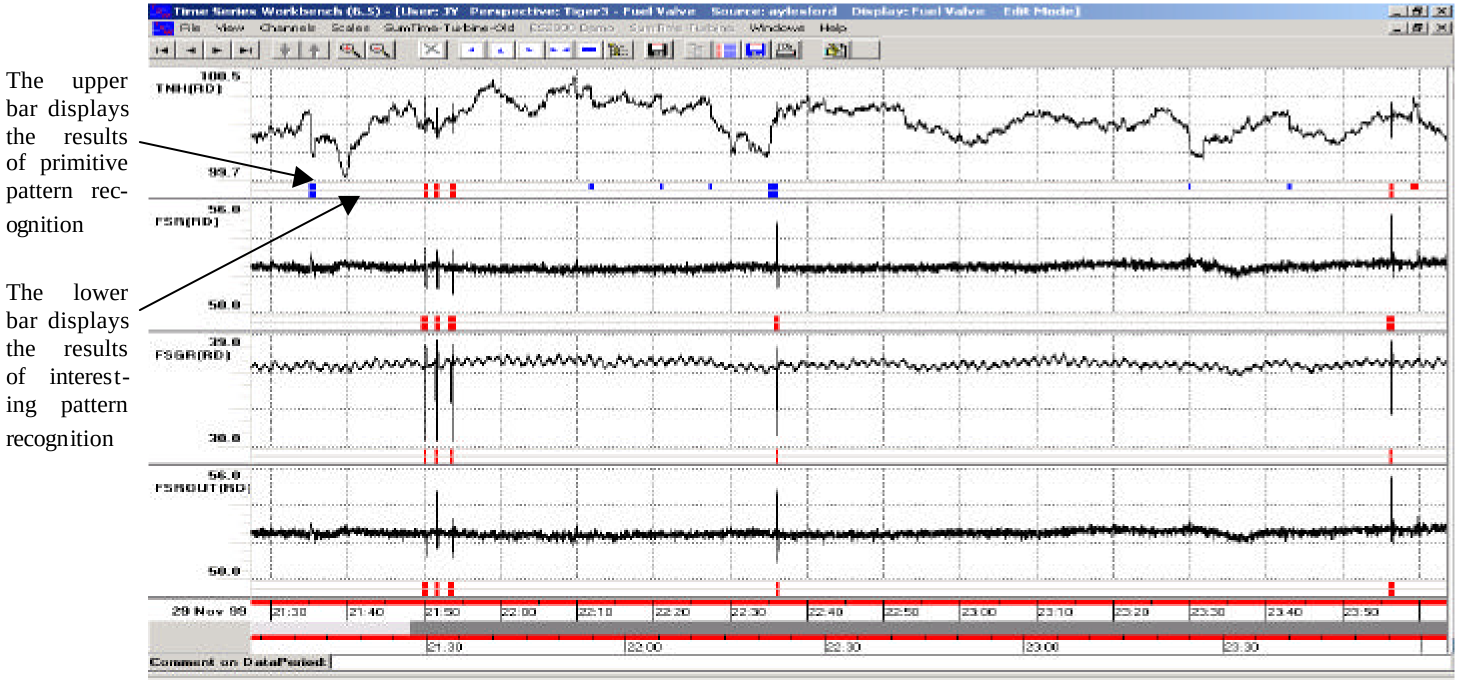

Fig. 2. Results of primitive and interesting pattern recognition produced by the sy stem

From the above results, we can see that 5 sets of simultaneous patterns are discovered in the sample data set after vertical aggregation. Horizontal aggregation reduces these to 3 main sets of simultaneous spikes because three sets of spikes at 21:49:49, 21:51:31, and 23:55:57, which are near each other, form one main set of spikes.

3.4 Summary

In this function, a summary with two parts is produced from different high-level abstractions. The first part gives the background knowledge about the current scenario and the other part describes interesting patterns in the scenario. The following is the summary produced by the prototype system on the sample temporal data set. We plan to use NLG technique [3] to improve the linguistic quality of the summary.

This scenario is about Fuel Valve subsystem which is being monitored by channels: TNH, FSR, FSGR, FSROUT, when the gas turbine is running in normal load state from 21:03:41.00 28 Nov 99 to 00:03:41 29 Nov 99.

During the time period, 3 main sets of spikes simultaneously occur in these channels. For example: Spikes in all channels at 21:49:49 and 23:55:57. Mostly spikes with some steps in all channels at 22:35:02.

Particularly, some large patterns occurred. For example: in channel TNH, a medium drop step at 21:35:04; in channel FSR, very big downward spike at 21:50:08, 21:51:36, 21:53:36; very big spikes with a medium rise step at 22:35:02 across all channels.

4 Evaluation of the Prototype System of SumTime-Turbine

The prototype system is being evaluated in two stages. The first stage is to evaluate primitive and interesting pattern recognition methods. The second stage is to evaluate the content of the output summary.

We have evaluated the pattern modules in randomly selected data sets from the *Tiger* data archive as described in [8]. The preliminary evaluation results are promising and we are extending the evaluation to more data sets

We have not yet evaluated output summary. Two possible methods have been proposed. One is let experts score summaries produced by *SumTime-Turbine*. Another is let experts write summaries about the same scenario and then compares the computer generated and human writers' summaries.

5 Related Work

There are a number of systems, which are related with summarised time-series data, of which are *RESUME*, *GoalGetter*, and *SumTime-Mousam*.

RESUME [4] uses KBTA method to create temporal abstractions from medical data. The KBTA framework provides the most comprehensive starting point for generating summaries of temporal data. However, it doesn't produce textual summaries.

GoalGetter [7] is a data-to-speech system, which generates Dutch spoken summaries of football matches. Its input data is teletext, while the input data of *SumTime-Turbine* is complex high-frequency multi-channel time-series data.

SumTime-Mousam [6] generates textual weather forecasts for the offshore oil rig applications by summarising time series data produced by numerical weather prediction (NWP) models. A major difference between the two systems is that *SumTime-Mousam* works with much sparse data (one point every three hours instead of one point second in *SumTime-Turbine*).

6 Future Work

Currently a working prototype system of *SumTime-Turbine* has been implemented. More functions are being developed and will be added into the system. For example, in KBTA subsystem, temporal pattern matching technique will be used to detect patterns suggested by domain experts. In NLG subsystem, micro planning and realisation will be improved in order to enhance the linguistic quality of text.

Acknowledgements

We are grateful to our collaborators at IA, especially Dr. Rob Milne and Dr. Jon Aylett, for their contributions to knowledge acquisition, and system implementation. This project is supported by the UK EPSRC under grant GR/M76881.

References

1. R. Milne, L.T. Massuyes (1997), Model Based Aspects of the TIGER Gas Turbine Condition Monitoring System, LAAS Report, 97416, CAAS-CNRS, Toulouse France
2. R. Milne, L.T. Massuyes, T. Escobet (2000), TIGER with Model Based Diagnosis: Initial Deploymen, LAAS Report, 00553, CAAS-CNRS, Toulouse France
3. E. Reiter and R. Dale (2000), Building Natural Language Generation Systems, Cambridge University Press
4. Y. Shahar, M. A. Musen (1996), Knowledge-Based Temporal Abstraction in Clinical Domains, Artificial Intelligence in Medicine 1996 8(3): 267-298
5. Y. Shahar, M. Molina (1998) “Knowledge-Based Spatial temporal Linear Abstraction”. Pattern Analysis and Applications 1(2): 91-104, 1998
6. S. G. Sripada, E. Reiter, J. Hunter, J. Yu (2002), Segmenting Time Series for Weather Forecasting. In Ann L. Macintosh, Richard Ellis, and Frans Coenen (Eds) Applications And Innovations in Intelligent Systems X, Springer, London, pages 193-206
7. M. Theune, E. Klabbers (2001), From Data to Speech: A General Approach, Natural Language Engineering 7(1): 47-86
8. J. Yu, J. Hunter, E. Reiter, and S. G. Sripada (2002), Recognising Visual Patterns to Communicate Time-Series Data in the gas turbine domain, In Ann L. Macintosh, Richard Ellis, and Frans Coenen (Eds) Applications And Innovations in Intelligent Systems X (Cambridge, U.K.), Springer, London, pages 105-118