Self-organizing map for conceptual modelling

Algirdas Laukaitis, Olegas Vasilecas

Abstract: *Textual description of the concept plays an important role in the conceptual model comprehensibility. In this paper, the self-organizing map to test the model comprehensibility is suggested. An explanatory text of the concept is transformed into a numerical vector. Several vector spaces are build using the hyponymy of the concepts from the WordNet dictionary. The received conceptual model vector space is tested for self-organisation properties with self-organizing maps. An experiment with the conceptual model self-organizing map and IBM toolbox for natural language understanding shows that it is useful to use them in the systems that support natural language modality. IBM financial services data model (FSDM) was used for the present research.*

Key words: *Natural language understanding, data conceptual modelling, self-organizing maps.*

INTRODUCTION

Natural language is a crucial communication tool in information systems (IS) development. But because of the ambiguity and vagueness of the natural language organisations uses a lot of time and costs just to communicate and find more centric approaches in the systems development. Several studies show that software engineers spend half of their time communicating in order to get information [8]. Even if, at the end of the development, the information systems are described fully in some more formal language (e.g. UML, Java etc.) almost all components of such system have natural language description. Nevertheless, while a huge corpus of documentation (documentation, business requirements, test results etc.) is created in the life cycle of the information system, document index remains the main tool to access the corpus ant there is need for technologies that enables more integration between the design process and natural language processing techniques. Recently there were numerous attempts to build the models for concepts extraction from text corpora [7]. In this paper, self-organizing maps (SOMs) [13] are proposed as a tool to support natural language interfaces in the information system design process and as the test for the conceptual model comprehensibility. Some philosophical arguments for the use of the self-organizing maps in the context of information systems development can be found in the paper of Timo Honkela [9].

The self-organizing map is a set of interconnected neurons. Each neuron represents after the learning some class of the data set. For the conceptual model self-organizing map each neuron can be interpreted as the set of similar concepts.

We can say that if the concepts with similar semantic meaning fall close to each other than conceptual model has a good comprehensibility and if similar concepts are scattered around the map than we can say that explanatory text of the concepts are purely written. Classification error can be used to measure model comprehensibility. In this paper, average quantization error [14] was computed for the conceptual model.

The key property of self-organization depends on vector space that describes the data. First, we investigate the vector space based on concepts description words frequency histogram. Next we transform the vector space by using words weighting technique with text parsing toolbox GATE [3] and with WordNet [17] ontology base to chance vector space.

The practical application for such self-organizing maps can be found in natural language understanding domain such as natural language interfaces with database management systems [1], [2]. Such maps can be used as the robust technique to identify conceptual meaning in particular domain and as a tool for coordination information system development with more conceptual model centric approach.

The contribution of this paper is as follows. First, we describe conceptual model that we used in our experiments and conceptual model vector spaces. Next, self-organized map based on vector space of conceptual model is presented. Finally, experiment with the conceptual model self-organizing map and IBM natural language understanding toolbox [12] is presented. Three students queried the system for concepts identification by asking the system about the facts from database based on the conceptual model. IBM natural language toolbox has been used as the benchmark for concepts identification accuracy.

CONCEPTUAL MODELLING AND NATURAL LANGUAGE UNDERSTANDING

Conceptual centric modelling can be effective tool for driving ambiguity and vagueness out of IS applications. But Conceptual data centric enterprise wide models are rarely build and few organizations even tried to surround their IS and business activities with such models. The problem with conceptual data centric enterprise wide models is that they are difficult to understand. Their abstract and generic concepts are unfamiliar to both business users and information systems professionals, and remote from their local organizational contexts [5]. In that context natural language processing and understanding techniques can be used to solve mentioned problems.

The relation between conceptual model and NL has been analyzed in the paper of [10]. The authors noticed that efficiency and correctness of communication in the process of systems development could be improved when domain specialists can be confronted with NL phrases expressing the change of the conceptual model.

The importances to generate natural language from conceptual model have been analyzed in the paper of [4]. The author noticed that most people do not understand formal languages, but they understand natural languages, therefore it is desirable to have a tool, which automatically generates natural language from a formal specification.

Another area where conceptual modeling is related to the natural language processing is information extraction from textual corpora. In the paper of Embley et al. [6] has been shown that conceptual model can be useful for information extraction from the web. The conceptual model self-organizing map can be used in that context as well but such research is-out-of-scope in this paper.

We have found this in several Baltic and Scandinavian banks working with the IBM financial services data model (FSDM) [11], which is a domain specific model, based on the ideas of the experts from IBM banking solution centre. To boast the awareness and project-centric approach we integrated the model into the created data exploration and information extraction framework JMining [15], [16]. The model is shown to consist of a high level strategic classification of domain classes integrated with particular business solutions (e.g. Credit Risk Analysis) and logical and physical data entity-relationship (ER) models. In JMining Dialog system the user identifies concepts by using natural language on all conceptual models levels: the 'A' level identifies nine data concepts that define the scope of the enterprise model (involved party, Products, arrangement, event, location, resource items, condition, classification, business), the 'B' level contains with business concepts hierarchies (more than 3000 concepts), the 'A/B' business solutions (integrates more than 6000 concepts with more than 50 solutions) and 'C' level – entity relationship ER diagram with about 6000 entities, relationships and attributes.

In figure 1 we can see the small part from conceptual model. If the user brings the input, "show all arrangements with the type loan", the system activates the conceptual model graph paths with different probabilities for each concept e.g.: 1) Arrangement (0.59) -> Arrangement Family (0.42) -> Account Arrangements (0.40) -> Loan Arrangements (0.14), 2) Arrangement (0.59) -> Arrangement Family (0.42) -> Arrangements Type (0.25) -> Product Arrangements Type (0.23) etc.

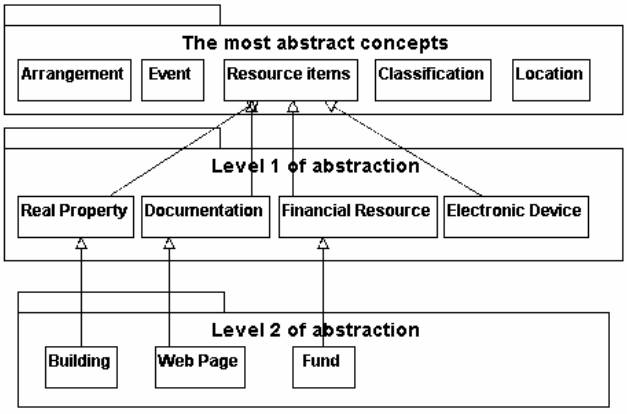

Figure 1. The small part from conceptual model.

As we see the user natural language input activates not just one concept but a path on conceptual graph. Then intelligent agents can act on that information e.g. agent responsible for SQL understanding can build the SQL sentences from identified databases, agent responsible for dialog handling can propose several options for user and ask to specify more accurately what the user has in mind.

VECTOR SPACE AND SELF-ORGANIZING MAP OF THE CONCEPTUAL MODEL

In the past ten years, self-organizing maps have been extensively studied in the area of text classification [13], [18]. The ideas and algorithms presented in those works we adapted in the context of the conceptual modelling.

Figure 2 shows the main activities of the conceptual model self-organizing map design. The vector space of the conceptual model has been build as follows.

1. Transform conceptual model. As the first step we transform conceptual model to the OWL [19] structure. The motivation behind this step is the in the future the OWL will be likely the most popular standard in describing the knowledge bases and we think to reuse such bases in the future research.

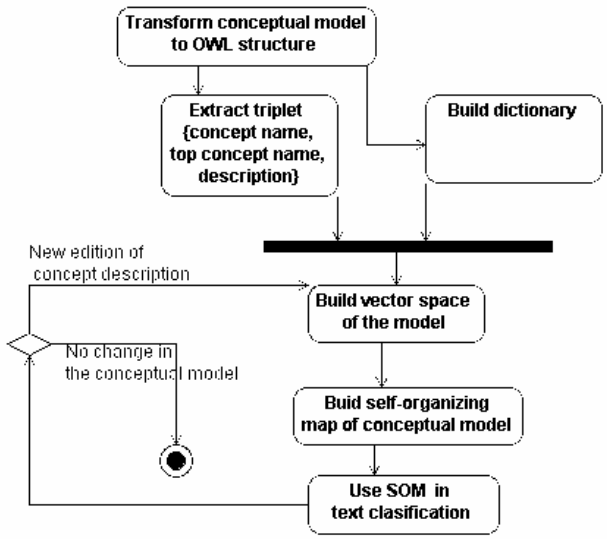

Figure 2. Main activities of the conceptual model self-organizing map design.

2. *Build dictionary.* At this step 900 nouns has been extracted from all explanatory descriptions of the concepts. Ordered set of extracted nouns formed dictionary that is used to form vector space of conceptual model.

3. *Extract triplet.* The triplet: concept name, the most abstract parent concept name, and description of the concept are extracted.

4. *Build vector space of the model.* The best way to explain this step is to present example. Lets assume that we have the concept "arrangement" with the following explanatory text:

"Arrangement represents an agreement, either potential or actual, involving two or more Involved Parties, that provides and affirms the rules and obligations associated with the sale,… ,"

and if the dictionary of words is the set that looks like:

{ acceleration, acceptance, accessibility, accident, accommodation, … arrangement, …, sale, … }

then not normalized vector representation of the concept will look like:

{ 0 , 0 , 0 , 0, 0, … 2, …,1, … }.

Such naïve representation of the conceptual model can be used as the benchmark for more sophisticated transformations. In addition to this representation, we used a method that utilizes the WordNet ontology bases [17]. The WordNet bases have been used a broad range of experiments on textual corpora (see, for example, [18]). In our case, we use semantic relationships in a set of synonyms representing distinct concepts. Hypernym-hyponym relationships are used to determine whether we can obtain fewer, more general concepts. In the example above, the concept arrangement as well as concepts {prearrangement, collusion, formation etc.,} will be mapped as one single concept. In addition the GATE was used to extend semantic mapping of the WordNet initially used by others researches [18]. With GATE toolboxes some natural language processing techniques, such as tagging, parsing, and word sense disambiguation can be integrated with statistical word knowledge.

In our experiment we extracted 1200 concepts explanatory descriptions and they formed a sample base that we transformed to the vector space representation using method described above. In addition, we labelled each concept with the labels that represent top parent root nodes of the conceptual model. For example, concepts customers, department, group, employees etc. have the same label of top parent concept involved party.

The question that we ask is how well we can separate concepts from each other by they explanatory descriptions and can we find some clusters that resemble conceptual model structure.

Figure 3 shows the self-organizing map for IBM financial warehouse conceptual model [11]. Each neuron of the map is labelled with the wining root concept label. For example, if top left neuron mapped 5 concepts that have top parent concept "Involved party" and 2 concepts with top parent concept "Arrangement", then the neuron will be labelled "Involved party" ("invol" in the figure 3).

It has been expected that if the conceptual model vector space has some clusters that resembles conceptual model itself, then we can expect that the model will be easer understood compared with the model of more random structure. We can se from figure 3 that there are some clusters of the concepts e.g. arrangement (labelled arran). On the other hand some concepts are not well clustered.

Self-organizing maps can be used as exploratory tool in the modelling process. The next section gives some details about an experiment, when the number of users queried conceptual model using natural language to identify the concepts and to get information form database.

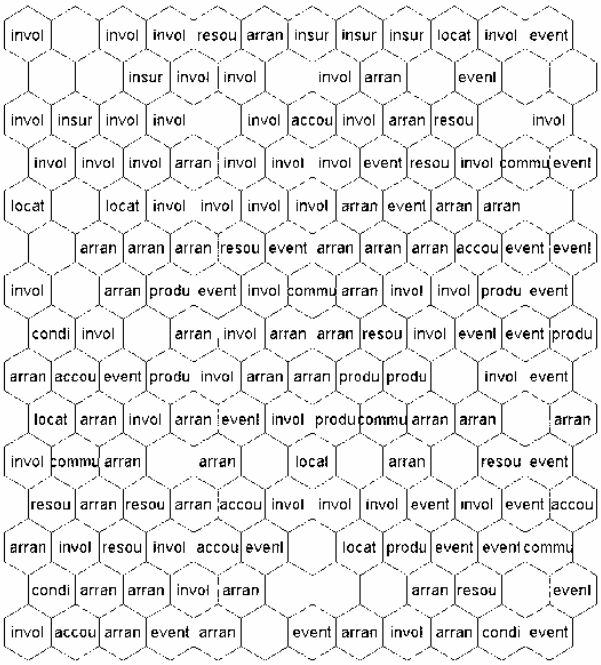

Figure 3 Conceptual model vector space self-organizing map. Labels: invol, accou, locat, arran, event, produ, resou, condi represents concepts involved party, accounting, location, event, product, resource, condition.

SELF-ORGANIZING MAP FOR NATURAL LANGUAGE UNDERSTANDING

The objective of this research was to find the techniques and the tools for concepts identification in textual corpora. The state-of-the-art natural language understanding (NLU) systems are still in the infancy stage end experiment below confirms this statement (see table 1). We have made primary evaluation of WebSphere Voice Server NLU toolbox, which is a part of the IBM WebSphere software platform. From IBM presentation [12] it appeared that the system is primarily intended for telecommunication market. It was a challenging task to test it on more a complex system e.g. a full conceptual model for financial services. The IBM NLU system uses statistically based models, which as they claim, provide more flexibility and robustness compared with traditional grammar-based methods. Much of the algorithm is unknown because the product is proprietary. Self-organizing map has been used as the alterative to IBM solution. In the present research the black box approach was used for both solutions: put the training data, compile and test the system response to the new arriving data.

The sets of pairs were constructed to train the IBM NLU model. The same set has been used for self organizing map. Each element of the set is represented by one concept. The firs part of the element is the label of the concept (e.g. arran – arrangement, etc.). In the case of IBM NLU toolbox, it is used as the class label that the system identifies from user's text input. In the case of self-organizing map it is used to label the wining neuron. The second part of the element is the set of sentences that describes the meaning of the concept. After the teaching process has been finished, the following experiment has been conducted with the IBM NLU toolbox and conceptual model self-organizing map.

A group consisting of 3 students was instructed about the above data model. They queried the system with about 20 questions and tried to identify the "Involved Party" concept. We increase the number of concepts that we put into the model for teaching to identify them. At the beginning only 9 top 'A' level concepts were considered. In this case for training data a description of these concepts were extracted from the original IBM model. At the second stage, the descriptions from child concepts were added to the training data for these 9 top parent concepts (see the second row in the table). Next the number of concepts was increased to 50 and finally 500 concepts with their descriptions

were extracted. Table 1 shows the results of the experiment. To detect the classification error the proportion of the correct identified concepts was used.

Table 1. Concept identification comparison between IBM NLU toolbox and self-organizing map of conceptual model.

	CN=9	CN=50	CN=500
1. IBM NLU	0.3682	.1726	0.0822
2. Self-organizing map of conceptual model.	0.4590	0.2814	0.1874

We were faced with a critical scalability problem. There were several instances in training when the system diverged from any reasonable acceptance level. While it was possible to make the training successful through manual intervention by adding more training data, the problem of divergence remained when the number of concepts increased up to the full conceptual model. The present research has shown that there is a lack of descriptive power for entities identification when training data include only brief descriptions of the conceptual model entities (as in IBM FDWM).

CONCLUSIONS

In this paper we presented the self-organizing map of conceptual model. The subject of integrating natural language processing techniques in software development is difficult and challenging task. Our results shows that self-organizing maps can be useful for this purpose but more research are needed in that direction.

On of the next extensions will be integration of Internet web pages with conceptual model for building more robust self-organizing maps. Our first attempts with the Google Internet search engine in that direction shows promising results.

REFERENCES

[1] Androutsopoulos, I., Ritchie, G.D., Thanisch, P.: Natural Language Interfaces to Databases – An Introduction. Natural Language Engineering, 1(1):29–81, (1995).

[2] Androutsopoulos, I., Ritchie, G.D., Thanisch, P.: Experience Using TSQL2 in a Natural Language Interface. In J. Clifford and A. Tuzhilin, editors, Recent Advances in Tem- poral Databases – Proceedings of the International Workshop on Temporal Databases, Zurich, Switzerland, Workshops in Computing, pages 113–132. Springer-Verlag, Berlin, (1995).

[3] Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Wilks, Y.:Experience of using GATE for NLP R/D. In Proceedings of the Workshop on Using Toolsets References 200 and Architectures To Build NLP Systems at COLING-2000, Luxembourg, (2000).

[4] Dalianis. H.. Concise Natural Language Generation from Formal Specifications., Ph.D. Thesis, (Teknologie Doktorsavhandling), Department of Computer and Systems Sciences, Royal Institute of Technology/Stockholm University, June 1996. Report Series No. 96-008, ISSN 1101-8526, SRN SU-KTH/DSV/R--96/8--SE.

[5] Darke, P, Shanks, G. Understanding Corporate Data Models, Information and Management 35 19-30, (1999).

[6] Embley, D.,W., Campbell, D.,M., Jiang, Y.,S., Liddle, S.,W., Lonsdale, D.,W., Ng, Y.,K., Smith, R.,D. Conceptual-model-based data extraction from multiple-record web pages. Data & Knowledge Engineering, 31:227-251, (1999).

[7] Gaizauskasand. R., Wilks., Y. InformationExtraction: Beyond Document Retrieval. Journal of Documentation, 54 1 :70105, (1998).

[8] Hertzum, M., Pejtersen, A., M. The information-seeking practices of engineers: searching for documents as well as for people. Journal of Information Processing and Management, Vol(36),pp.761-778, (2000).

[9] Honkela, T. Von Foerster meets Kohonen - Approaches to Artificial Intelligence, Cognitive Science and Information Systems Development. Kybernetes, Vol. 34, 1/2, p. 40 – 53, (2005).

[10] Hoppenbrouwers, J., van der Vos, B., and Hoppenbrouwers, S. NL Structures and Conceptual Modelling: The KISS Case. In van de Riet, R., Burg, J., and van der Vos, A., editors, Application of Natural Language to Information Systems, pages 197--209. (1996)

[11] IBM IBM Banking Data Warehouse General Information Manual. Available from on the IBM corporate site http://www.ibm.com (accessed July 2005).

[12] IBM. An Introduction to IBM Natural Language Understanding. An IBM White Paper. Available from on the IBM corporate site http://www.ibm.com (accessed July 2005).

[13] Kohonen, T.,. Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43, 59-69, (1982).

[14] Kohonen, T. Self-Organizing Maps, Springer-Verlag, 2001.

[15] Laukaitis, A., Vasilecas, O., Berniunas, R. JMining - information delivery web portal architecture and open source implementation // Edited by O. Vasilecas et al. Information Systems. Development: Advances in Theory, Practice and Education., Springer, 2005.

[16] Laukaitis, A., Vasilecas, O. An architecture for natural language dialog applications in data exploration and presentation domain. ADBIS 2005 m.

[17] Miller, G.A.: WordNet: A Dictionary Browser, Proc. 1st Int'l Conf. Information in Data, pp. 25–28, (1985).

[18] Wermter, S.: Hybrid Connectionist Natural Language Processing, Neural Computing Series, Chapman & Hall, (1995).

[19] Web Ontology Language (OWL), http://www.w3.org/2004/OWL/.

ABOUT THE AUTHOR

Assoc.Prof. Algirdas Laukaitis, Department of Information Systems, Vilnius Gedinimas Technical University, Phone: +370 52744860, E-mail: algirdas@fm.vtu.lt

Prof. Olegas Vasilecas, PhD, Department of Information Systems, Vilnius Gedinimas Technical University, Phone: +370 52744860, E-mail: olegas@fm.vtu.lt.