MULTIDIMENSIONAL NETWORKS FOR HETEROGENEOUS DATA MODELING

Sergey Maruev, Dmitry Stefanovskyi, Alexander Troussov, John Curry, Alexey Frolov

***Abstract**: Big data frequently come in tabular form of rows and columns of numbers, special codes and short textual descriptions, in strict, structured, disciplined formats generated by a variety of transactional and operational business systems. In this paper we discuss the advantages of modeling heterogeneous data by multidimensional networks in line with the concept known as "Graph databases". Graph-based methods provide a powerful abstraction for mining such data; however, it is hard to achieve good results in mining using of the shelf methods. In this paper we show how empirical methods of fuzzy logic could be injected into abstract graph-based methods to achieve desirable results. We outline the wide range of applications of that modeling and mining, and present our results on the use of our methods of modeling and mining for processing of custom declarations for commercial goods. We examine several use cases, including recommendations to custom officers and participants of the international trade. The feasibility of the approach was tested by application to 2500 custom records collected during a continuous period of one month at eight border checkpoints between Russian Federation and two EU countries. In several use cases the algorithm achieved high accuracy under experimental conditions.*

***Keywords**: big data, graph-based methods, custom declarations.*

***ACM Classification Keywords**: Algorithms, Economics, Experimentation, Theory.*

Introduction

Big data frequently come in tabular form of rows and columns of numbers, special codes and short textual descriptions, in strict, structured, disciplined formats generated by a variety of transactional and operational business systems. In this paper we discuss the advantages of modeling heterogeneous data by multidimensional networks in line with the concept known as "Graph databases". Graph-based methods provide a powerful abstraction for mining such data; however, it is hard to achieve good results in mining using of the shelf methods. In this paper we show how empirical methods of fuzzy logic could be injected into abstract graph-based methods to achieve desirable results. We outline the wide range of applications of that modeling and mining, and present our results on the use of our methods of modeling and mining for important area of applications - processing of custom declarations for commercial goods.

International trade is one of the most important drivers of the global economy. Therefore, the study of impediments to this trade is of interest to the field of international economics. International trade is typically more costly than domestic trade due to the imposition of extra direct and indirect costs including tariffs, time costs due to border delays and processing costs that are exacerbated by differences in language, legal system and culture, see, for instance, [Zvetkov et al. 2013] in Russian.

We examine several use cases, including recommendations to custom officers and participants of the international trade. The feasibility of the approach was tested by application to 2500 custom records (which have 12043 items of goods) collected during a continuous period of one month at eight border checkpoints between Russian Federation and two EU countries, the same data set that was used in [Maruev et al. 2014].

We tested our approach on the use case of computing the code of custom goods based on the textual description provided in the declaration; the algorithm achieved high accuracy under experimental conditions.

The rest of the paper is organized as follows. Representing data in rows and columns probably has been the most pervasive formal method for data collection, representation and analysis in all areas of human activities, notably including the use in computer data bases. In section entitled “Network Modeling vs. Tabular Representation” we provide a brief description of modeling using multidimensional networks and comparison of such modeling with the table representation. We show how tables could be converted into multidimensional networks, and argue that such modeling naturally lends itself to the discovery of patterns. The bulk of the paper is the demonstration how real world data about custom declarations could be modeled by networks and explored using methods of graph mining.

In the next section - “Custom Declarations data” - we describe the data used in this paper. In section “Network Data Representation and Mining” we present a particular way of network modeling tailored to the task of prediction of the nomenclature code of goods from the textual description. The resulting network is a multidimensional network with two types of nodes: nodes corresponding to the nomenclature codes of goods, and nodes corresponding to the words used in natural language descriptions of goods. Assuming that all the data in our collection have correct nomenclature codes, we can consider the obtained network as an encapsulation of the knowledge about the relations between codes and words in the textual descriptions of the goods. New textual descriptions could be mapped into nodes of this network, the results of the mapping might be considered as a fuzzy set of nodes. Measuring “proximity” of this set to nodes representing nomenclature codes one can quantify the relevancy of the description to certain codes. In this paper we use the generic computational scheme on networks called spreading activation [Troussov et al., 2009] for this purpose. In section “Evaluation” we present the results of experimental validation of recommending codes based on the textual description of goods. Finally, section “Conclusions and Future Work” describes the conclusions and future work.

Network Modeling vs. Tabular Representation

Network modeling is endemic throughout various domains of applications, including social and semantic web, communications. To introduce network method for custom declaration’s data representation, we juxtapose this method with the applications in computational linguistics, where we can visually show the difference between network and tabular representations.

Many data in office applications comes as “tables”, which could be processed by spreadsheet applications such as Microsoft Excel. Data which are viewed as networks, such as social networks, in many cases are converted to matrix form (to a “table”) as incidence matrix and processed using linear algebra methods (in mathematics, an incidence matrix is a matrix that shows the relationship between two classes of objects.). However, linear algebra provides only a subclass of useful graph-mining techniques.

In this paper we argue that the usefulness of graph-based methods for mining of unstructured heterogeneous data (usually represented as tabular data) is underappreciated. This statement is somewhat similar to the ideas which led to the coinage of the term “graph databases” [Rodriguez, 2011], although our emphasize is solely on the data representation and mining algorithms relying on the navigation through a network using links between neighbors, not on the methods of storage of graphs or spars matrix.

To illustrate our point let us consider computer dictionaries for natural languages.

Many lists of common English words starts from aardvark, aardwolf, abacus, … . To use such lists for solving crosswords in would be suitable to model the data in a tabular form like this:

a	a	r	d	v	a	r	k
a	a	r	d	w	o	l	f
a	b	a	c	u	s		

Fig. 1. A list of a few English words could be represented in a tabular form, which could be suitable for certain applications, like solving crosswords or computing of statistics of letters in certain positions

The same table could be redrawn as a graph where nodes represent letters.

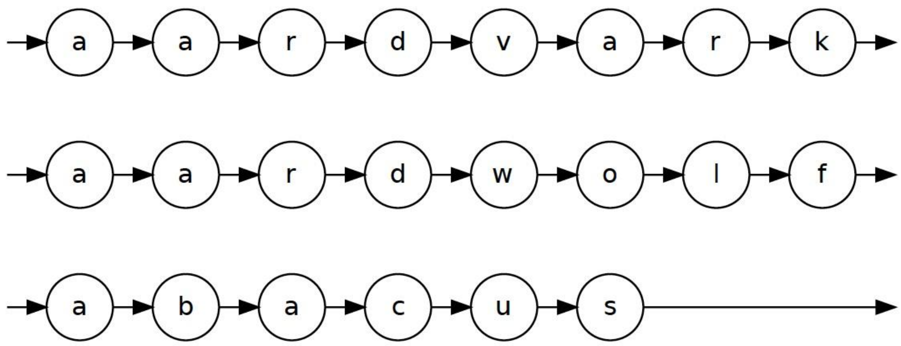

Fig. 2. The list of English words from the Fig. 1 could be visualized as a graph.

If this graph is used to construct a computer dictionary, it can be compactified to the following form:

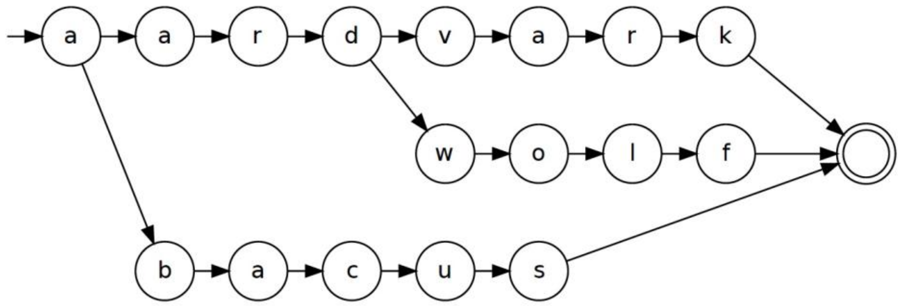

Fig. 3. This graph model has exactly the same data as the original list of common English words on the Fig. 1, and is usually called the Mealy finite-state machine.

For computer science problems, such as designing the data structure for dictionaries which support search operation, a standard solution using hash tables could be used as well as the Mealy finite-state machine shown on the Fig. 3; and one can argue about advantages and disadvantages of both methods. However, the situation drastically changes when we move from lists of random character strings to the lists of words from the vocabulary of a natural language.

When the strings are words from a natural language, graph-based representation of data has at least one crucial advantage. Firstly, the graph-representation becomes very compact since common prefixes and affixes of words are conflated, and this leads to non-functional advantages (in memory footprint and processing speed). More importantly, even when such compactification is provided by formal mathematical methods, which are unaware of the morphology, effectively they produce a representation of the initial list of strings in a graph form which shows patterns of the morphology of the language; therefore it becomes possible to process out-of-vocabulary words, like *trichloroisocyanuric*, and to construct morphological guessers [Jurafsky and Martin, 2009]. For instance, a morphological guesser might infer that the word *ontologization* is a well formed English noun, and to find that it is related to the noun *ontology*. Moreover, using graph-representation one can infer that the relation between the pair of words *ontology-ontologization* is the same as the relation between words *industry-industrialization* (see, for instance, [Troussov and O'Donovan, 2003]).

The procedure of the processing out of vocabulary words, like the word *ontologisation*, might be summarized in the following diagram:

```
Input: a new word: ontologization  → Network model
           of existing words   →   Patterns which the input follows
```

When the network model of custom declaration has been already constructed, the scheme of processing new declarations is the same as it is in the above mentioned morphological applications:

```
Input: a new custom declaration record →
             → Network model of custom declaration data →
                          → Patterns which the input follows
```

There is also a significant difference between these two use cases. In the computational linguistic all the patterns of the language could be discovered using the human insight and the methods of the computational linguistics. In case of mining custom declarations, on the fly and on demand discovery of emerging patterns moves to the foreground.

Custom Declarations Data Description

The data used in this paper are the same as in the paper [Maruev et al. 2014]. In this section we provide a short overview of this data.

The raw data for this study originates with traders shipping goods into the Russian Federation through land borders. They comprise a description of the itemized contents of a shipment of goods in a particular vehicle (always a truck in this study). These data are used to produce custom goods declarations and to compute taxes. When the truck crosses the border, the data become part of the custom service's electronic data archive.

Each item record describes a specific type of goods, and has several numeric and alphanumeric fields, including identification numbers of consignee, consignor, and carrier; gross weight, invoiced cost; currency code and currency rate. Fields relevant to this paper are:

- GoodsTNVEDCode – ten digits code for the commodity. This nomenclature is used in the Customs Union of Belarus, Kazakhstan, and Russia and is also consistent with the codes used in the European Union
- GoodsDescription – goods description;

Table 1 shows an example of an item record which uses Russian language description of goods related to printing machinery.

Table 1. An example of goods item with the nomenclature code (GoodsTNVEDCode) 8443999009.

№	DI	E	C1	C2	C3	Goods-TNVEDCode	GoodsDescription	GW	InvoicedCost	CC	CR
1	1	1	385	389	785	8443999009	ЧАСТИ И ПРИНАД-ЛЕЖНОСТИ ПЕЧАТНЫХ МАШИН	7482.320	531495.03	USD	33.2474

Network Data Representation and Mining

We represent these data as multidimensional networks (see [Troussov et al., 2011]). Nodes and links are typed. Nodes represent complete data fields or, as in the case of the “GoodsDescription”, particular words from that textual field. An example of such a network is given on Fig. 4 below. This network and the methods of its construction will be described in Subsection “Network Construction”.

In this paper we focus on a particular task – to automatically detect the code of the commodity based on its textual description. The same as in the paper [Maruev et al., 2014], we can consider this task as a task of supervised machine learning. We split the data set into two halves, learn the rules based on the first half, and apply the rules to the second half (not used in the procedure of learning). We then validate the results against the known outcomes to assess the predictive power of our rules.

In our approach, learning is done in two distinctive stages. Firstly, we build the network from the data. Secondly, we use graph algorithms to find patterns in the network.

Fig.4 shows the fragment of the network constructed in our validation experiment. The network was generated from the original tabular data in the reduced feature space where only nomenclature codes of the commodity and words from textual descriptions are used.

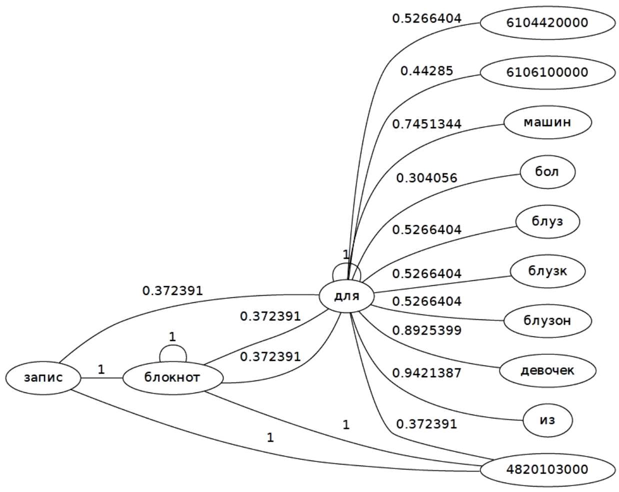

Fig. 4. A fragment of the network generated from the original list of goods items. Nodes labeled with numbers represent goods codes; nodes labeled with strings of alphabetical characters represent words used in textual descriptions. Links are weighted. This network shows, for example, that the word “для” (“for” in English) has been met in many goods descriptions and with many goods codes. Most of the formal methods, used to compute structural importance of the nodes in the network, will rate this word very high; one can also wrongly assume that this word could be a good predictor for many different codes. The mining method described in this paper allows preventing such nodes to dominate the results.

NETWORK CONSTRUCTION

The construction and the use of this network consist of the following generic steps which could be adjusted for use in other scenarios.

1. The first 8000 items description rows were used to construct the network
2. We removed all the fields except the commodity codes (‘GoodsTNVEDCode”), and the goods description (“GoodsDescription”).

The removal is not a necessary step, but we have done it in order to reduce the volume of learning data and test our algorithm under harsh conditions.

3. Each word has been reduced to its normalized form by the procedure known as stemming, see, for instance, [Jurafsky and Martin 2009]. Stemming is an empirical natural language processing procedure allowing to map inflected and derived words into one index form; for instance, to map "fishing", "fished", and "fisher" to the form "fish". We used Porter stemmer for Russian to perform this procedure.
4. As the result, we obtained 4352 entities, which were merged into one network with 4352 nodes.

If the two entities are met at least once in one same shipment document, the corresponding pair of nodes is connected by an arc. The weight of that arc represents how frequently the two entities corresponding to the pair of nodes are met in a shipment document, i.e. the number of co-occurrences divided by the number of items.

MODELING NEW DOCUMENTS AS SETS OF NODES ON THE NETWORK

The network represents the learning data. Each new document could now be modeled as a set of nodes on this network. I.e. for each new item we need to perform steps 1-3 described above. Each new entity is mapped into a corresponding node on the network. If such a node is not found, the entity is ignored. It could not be usefully present in the model because our “learning” has no knowledge about such entities.

MINING

When we encode a set of data as a network, such as described in the previous subsection, mining now can be done by various graph-based methods. For the recommendation tasks, one can model the situation as a set of nodes on this network, and based on the graph-topology discover other nodes which might be relevant (close) to the initial conditions. Specifically for our task - computing GoodsTNVEDCode based on the textual description – we model the description as a set of nodes (corresponding to individual orthographic words in the description) on the network, and using graph-methods find the most relevant nodes representing GoodsTNVEDCodes. To find and rank related nodes, we used the set of operations based on the Spreading Activation Method, described in [Troussov et al. 2009], and its generalization in the paper [Troussov et al. 2011].

The Spreading Activation Method has its origin in neurophysiology: “In neurophysiology interactions between neurons is modeled by way of activation which propagates from one neuron to another via connections called synapses to transmit information using chemical signals. The first spreading activation models were used in cognitive psychology to model this processes of memory retrieval.” – [Troussov et al. 2009]. Later this framework was exploited in Artificial Intelligence as a method for searching associative, neural or semantic networks; see, for example,[Crestani 1997], [Aleman-Meza et al. 2003], [Rocha et al. 2004].

In terms of the spreading activation, our mining could be explained as follows: we put the initial activation at those network nodes which correspond to words used in the description, and compute how much activation comes to the nodes corresponding GoodsTNVEDCodes. Spreading activation serves as a search method in the work, which also allows to compute the cumulative strengths of connections between the words in the description and the GoodsTNVEDCode.

Depending on the task of mining and the structural properties of the network, a few up to several dozen iterations of spreading activation are normally sufficient to achieve the goal. We found that one iteration of spreading activation is enough for our purposes. In other words, the number of arcs and their weights between the model and the node GoodsTNVEDCode is a good predictor that the goods code is consistent with the given textual description. The larger the weights, likelier it is that the goods code is a correct one. The effect of the number of arcs here is much less evident, because, logically, a high number of arcs with small weights indicate that the goods code is wrong.

To aggregate weights one can use the arithmetic mean of the weights of arcs. However, for the use case of recommendations to assign an armed convoy for the shipment, [Maruev et al. 2014] found that arcs with high weights close to *1.0* were important, while arcs with small weights were not reliable predictors. Therefore, instead of arithmetic mean for n real numbers x_1, x_2, …, x_n representing weights, [Maruev et al. 2014] used the mean computed as the L^p –norm of the vector { x_1, x_2, …, x_n } with the parameter p empirically taken with the value *3.5* to favor links with high weights and to ignore links with very small weights:

$$\| \mathrm{x} \|_{\mathrm{p}} = (| x_1 |^{\mathrm{p}} + | x_2 |^{\mathrm{p}} + ... + | x_n |^{\mathrm{p}})^{1/p} \tag{1}$$

In this paper we use the same formula 1 for the aggregation; however, we found that the parameter p in our case should be different.

Evaluation

We used 8000 chronologically first data records to model the data as a network of custom goods codes and words used in goods description; and the rest of the data (4043 records) to test our algorithm. Each new record was broken into words and mapped onto the network, and the strength of its connection to various codes was computed using the spreading activation method. The output of this algorithm is the list of the nodes corresponding to the strength of the connection with the set of words, the strength of the connection in this context is called the level of activation (see [Troussov et al., 2009]). If two or more nodes have the same activation, the order in which they appear in the list is random.

If the most activated node is the same as the node corresponding to the code in the record in question, the code is considered to be predicted correctly. We investigated the accuracy of the prediction under the various values of the parameter p in formula 1.

If the recall is measured in top two goods codes (that is the result of the prediction is considered as to be correct if either the most activated node or the second most activated node is the node corresponding to the code in the record), the recall is 100%. In all cases where the correct answer was the second most activated node, the textual description was not sufficient to predict the nomenclature code; for instance, the code for tomato fruits depends on the additional information absent in the data we used, such as the season when the cargo enters the country. One can easily fix this particular problem with “hard rules”, but the goal of our pilot project is to create proof-of-the-concept scalable technology platform for mining custom declarations for cases not covered by hard rules of custom regulation, as opposed to another people-intensive manual solution.

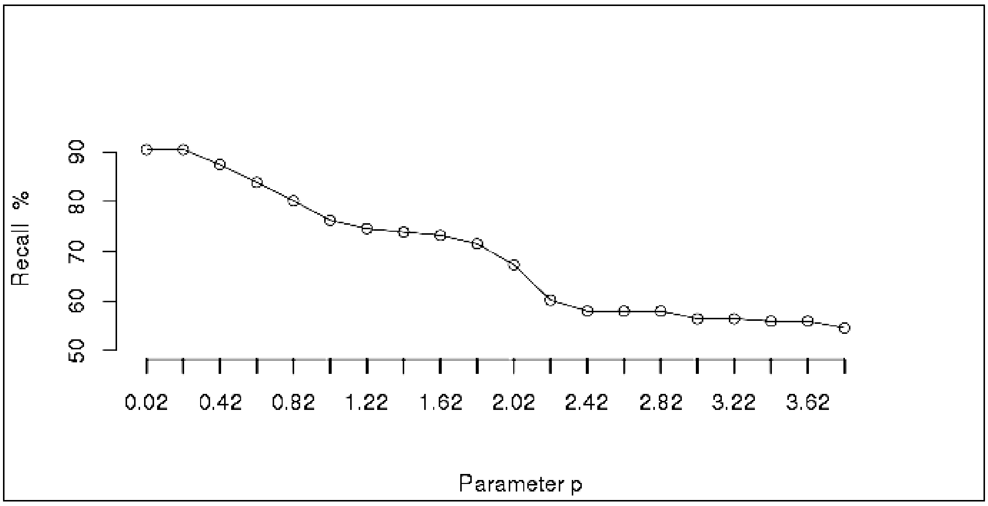

Fig. 5. The recall of the prediction of goods nomenclature codes in testing 4023 custom records based on the network created using different set of 8000 records. With the parameter p close to zero the recall is 90.5 %.

To understand if the achieved results are good or not in the paradigm of the machine learning, we need to take into consideration that the amount of the data used for learning (that is the data used to construct the network) is small (8000 records) as contrasted with the number of possible goods codes (about ten thousands). In addition, the data which we consider as the gold standard, might contain factual errors (such as the wrong codes), and definitely have certain number of misspellings in goods descriptions.

We conclude that the results of the experiments are the best possible to achieve under what might be considered as “harsh” conditions.

The most interesting theoretical result for us was the following. In this paper we used spreading activation as mining method, the same as was used in the paper [Maruev et al, 2014], where this method has been applied for a different use case of mining, namely recommendation of assigning armed convoy to the track when it crosses the border. Spreading of the activation algorithms are based on the iterative re-computation of activation of the network nodes. On each iteration the new level of activation is computed based on current activation in the node and the activation transferred to the node by its neighbors; in [Troussov et al., 2009], this stage is called “Computation of the New Level of Activation”.

Firstly, we discover that the same formula (formula 1) of re-computation used in [Maruev et al. 2014] works well for our task, however good results are achieved with different value of the parameter *p*. Formula 1 with paremeter p>=1 is used in fuzzy logic to implement logical operation AND using Jager’s *t*-norms (see [Chen, 1996]).

In [Maruev et al., 2014], the parameter *p* has been empirically taken with the value *3.5*. The rationale of this could be explained as follows. During the re-computation stage, the activation at each neighbor node around the node representing assigning of armed convoy could be considered as a predictor of convoy assignment; some predictors are more important than other. Formula 1 effectively aggregates these predictors into one real number using a specific type of logical operation AND. If this number is bigger than a certain threshold the algorithm recommends assigning the armed convoy. Aggregation of the several numbers into one could be done in many different ways, for instance one can take the arithmetic mean of these numbers. When the aggregation is done using formula 1 with $p=2.0$ the result become more dependent on the most important predictors; with the parameter $p=3.5$ the result become heavily dependent on the most important predictors, while less important predictors are practically ignored. Indeed, the escort is assigned to the track, so the fact that some of the goods in the track previously were not provided by the escort is not important, but the fact that some of the goods previously were escorted, is very important.

The experimental results depicted at the Fig. 5 show that the best results are achieved with small value of *p*, which essentially means that the number of the predictors for the particular goods code is important, while giving more importance to strong predictors quickly degrades the results. At the moment we can't explain this phenomenon in a rigorous way. Tentatively, we speculate along the following two lines.

Firstly, in our experiments to predict the code based on the textual description, we filter out some words using an empirical technique known in information retrieval as the removal of "stop words"; typically these are functional words forming so called closed word classes (see [Jurafsky and Martin, 2009]), like prepositions, conjunctions, particles, etc. However, after the removal of stop words we still have a big number of certain words (like *weight*, *mkm*) which describe quantity, weight and methods of packaging of goods, and are applied to many different goods codes. Such words technically become strong predictors to many completely unrelated goods codes. To improve the quality of graph mining one can manually remove such words and/or to develop graph-based methods which prevents these words from dominating the outcome of the algorithm.

Secondly, although some words are strong predictors of particular codes, and the noun phrases used in goods descriptions are mostly semantically compositional (that is the meaning of the expression is composed from the meaning of individual words (see [Jurafsky and Martin, 2009]), only the particular combination of all words in the description eventually determines the code. Based on the results of our experiments, it seems that our method of modeling captures this peculiarity very well in the topology of the network, and our mining method - spreading activation with formula 1 – could be quite sensitive to such combinations, and is capable to implicitly find such combinations and use them for the correct prediction of the goods code. To illustrate this let us consider the real example from our data – the item described as "*jacket for computational devices*" (*čexól dljz vuchisliteljnuch ustrojstv* in Russian, Russian equivalent for *jacked* actually means removable or replaceable protective or insulating cover for an object). Both words *computational* and *devices* are very strong predictors for the goods codes related to electronics. However "*jacket for computational devices*" is not an electronics goods item.

Conclusions and Future Work

We introduced a generic method for modeling tabular data as a multidimensional network, where nodes represent various codes and alphanumeric fields, as well as the terms extracted from the fields containing natural language phrases. The network form of representation provides ease of merge of heterogeneous information; external knowledge could be added on the top of the network obtained from data as new nodes and new weighted arcs. We validated our approach on a task of finding patterns in 2500 custom records, containing 12043 items of goods, collected during a continuous period of one month at eight border checkpoints between Russian Federation and two EU countries.

In this paper we described the application of this modeling method to a network form of data representation of custom goods declarations. The experimental results of the paper, in conjunction with the results of the paper [Maruev et al. 2014], demonstrate the applicability of spreading activation based algorithms for mining this data for two polar use cases: the assignment of an armed escort to a shipping vehicle in cases of elevated risk of theft, and the prediction of the nomenclature code of goods based on the textual description. Obtained results show potential applications of our method for building recommender systems for use by customs officers, traders, carriers and insurers.

Our modification of the traditional spreading activation technique [Troussov et al., 2009] allows the explicit injection of fuzzy logic into the computational scheme to address specific problems of mining. Future development of network process methods might be driven by "physics-inspired" (see [Troussov et al., 2011a]) and "logic-inspired" (including cellular automata and fuzzy logic) approaches, which will allow synthesizing algorithms with desirable outcomes.

Bibliography

[Aleman-Meza, B., Halaschek, C., Arpinar, I., Sheth, A., 2003]. Context-Aware Semantic Association Ranking. Proceedings of SWDB'03, Berlin, Germany, 33-50.

[Chen, C.H. (ed.), 1996]. Fuzzy Logic and Neural Network Handbook. McGraw-Hill, 700 pp.

[Crestani, F., 1997]. Application of Spreading Activation Techniques in Information Retrieval. Artificial Intelligence Review, 11(6), 453-482.

[Jurafsky, D. and Martin, J.H., 2009]. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition (2ed.), Prentice Hall.

[Maruev, S., Stefanovskyi, D., Frolov, A., Troussov, A., and Curry, J.] Deep mining of custom declarations for commercial goods. To appear at Procedia Economics and Finance, 2014.

[Rocha, C, Schwabe, D., Poggi de Aragao, M., 2004]. A Hybrid Approach for Searching in the Semantic Web. Proceedings of the 13th international conference on World Wide Web, May 17-20, 2004, New York, NY, USA, 374-383.

[Rodriguez, M. Knowledge Representation and Reasoning with Graph Databases, 2011]. http://markorodriguez.com/2011/02/23/knowledge-representation-and-reasoning-with-graph-databases/

[Troussov, A., and O'Donovan, B]. "Morphosyntactic Annotation and Lemmatization Based on the Finite-State Dictionary of Wordformation Elements". Proceeding of the International Conference Speech and Computer (SPECOM' 2003), October 27-29 2003, Moscow, Russia

[Troussov, A., Darena, F., Zizka, J., Parra, D., and Brusilovsky, P., 2011a]. "Vectorised Spreading Activation Algorithm for Centrality Measurement". Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis. sv. LIX, no. 7, s. 469--476. ISSN 1211-8516.

[Troussov, A., Jundge, J., Alexandrov, M., and Levner, E., 2011]. Social Context as Machine-Processable Knowledge. Proceedings of the International Conference on Intelligent Information and Engineering Systems INFOS 2011, Rzeszów - Polańczyk, Poland, pp. 104-114, ISBN: 978-954-16-0053-5.

[Troussov, A., Levner, E., Bogdan, C., Judge, J., Botvich, D., 2009]. Spread of Activation Methods. In Dynamic and Advanced Data Mining for Progressing Technological Development, Y. Xiang and S. Ali (eds) IGI Global.

[Zvetkov, V, Zoidov, K, and Medkov, A., 2013]. In Russian. Цветков, В, Зоидов, К., Медков, А. О возможности и целесообразности организации транзита через Россию грузов между странами Тихоокеанского региона и Европы. Депонирована в системе Соционет, февраль 2013 г. Retrieved February 27, 2014, from http://www.cemi.rssi.ru/mei/articles/tsvetkov-and13-01.pdf

Authors' Information

Sergey Maruev – Professor, The Russian Presidential Academy of National Economy and Public Administration, Prosp. Vernadskogo 82, bld. 5, Moscow, 119571, Russia; e-mail: maruev@gmail.com

Major Fields of Scientific Research: mathematical modeling, economics of education, business process modeling, social network analysis

Dmitry Stefanovskyi – Assoc. Prof., Ph.D, The Russian Presidential Academy of national economy and public administration; Prosp. Vernadskogo 82, bld. 1, Moscow, 119571, Russian Federation; e-mail: dstefanovskiy@gmail.com

Major Fields of Scientific Research: mathematical modeling, world economy

Alexander Troussov – Director of the International Research Laboratory for Mathematical Methods for Social Network Mining, The Russian Presidential Academy of National Economy and Public Administration (RANEPA), Prosp. Vernadskogo 82, bld. 5, Moscow, 119571, Russian Federation; e-mail: troussov@gmail.com

Major Fields of Scientific Research: Natural language processing, Information Retrieval, Social and semantic web, social network analysis, graph-based methods

John Curry – Office of the Revenue Commissioners, Central Revenue Information Office, Cathedral Street, Dublin 1, Ireland, e-mail: curry.john@gmail.com

Major Fields of Scientific Research: Data mining, Software technologies

Alexey Frolov – The Federal Customs Service of Russia, ROSTEK-Pskov, Director, Ul. Vorovskogo, bld.20, 180000, Pskov, Russian Federation, e-mail: info@rostek-pskov.ru

Major Fields of Scientific Research: Economics, World economy