Selected Problems of Knowledge Discovery Using Artificial Neural Networks

Keith Douglas Stuart[1] and Maciej Majewski[2]

[1] Polytechnic University of Valencia, Department of Applied Linguistics
Camino de Vera, s/n, 46022 Valencia, Spain
kstuart@idm.upv.es

[2] Koszalin University of Technology, Faculty of Mechanical Engineering
Raclawicka 15-17, 75-620 Koszalin, Poland
maciej.majewski@tu.koszalin.pl

Abstract. The paper describes the application of an artificial neural network in natural language text reasoning. The task of knowledge discovery in text from a database, represented with a database file consisting of sentences with similar meanings but different lexico-grammatical patterns, was solved with the application of neural networks which recognize the meaning of the text using designed training files. We propose a new method for natural language text reasoning that utilizes three-layer neural networks. The paper deals with recognition algorithms of text meaning from a selected source using an artificial neural network. In this paper we present that new method for natural language text reasoning and also describe our research and tests performed on the neural network.

1 Introduction

For linguistic research, there is a need for consciously created and organized collections of data and information that can be used to carry out knowledge discovery in texts and to evaluate the performance and effectiveness of the tools for these tasks. Knowledge discovery in text is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in unstructured textual data [14,15]. These patterns are unknown, hidden or implicit in semi-structured and unstructured collections of text. Below are some of the kinds of knowledge discovery tasks that many subject disciplines are interested in:

– Identification and retrieval of relevant documents from one or more large collections of documents.
– Identification of relevant sections in large documents (passage retrieval).
– Co-reference resolution, i.e., the identification of expressions in texts that refer to the same entity, process or activity.
– Extraction of entities or relationships from text collections.
– Automated characterization of entities and processes in texts.

D. Liu et al. (Eds.): ISNN 2007, Part III, LNCS 4493, pp. 1049–1057, 2007.

– Automated construction of ontologies for different domains (e.g., characterization of medical terms).
– Construction of controlled vocabularies from fixed sets of documents for particular domains.

The need to construct controlled vocabularies for subject domains has meant that terminological extraction from corpora has become an important process in tasks related to knowledge discovery in text [14].

The proposed system for knowledge discovery in text uses neural networks for natural language understanding in Fig. 1.

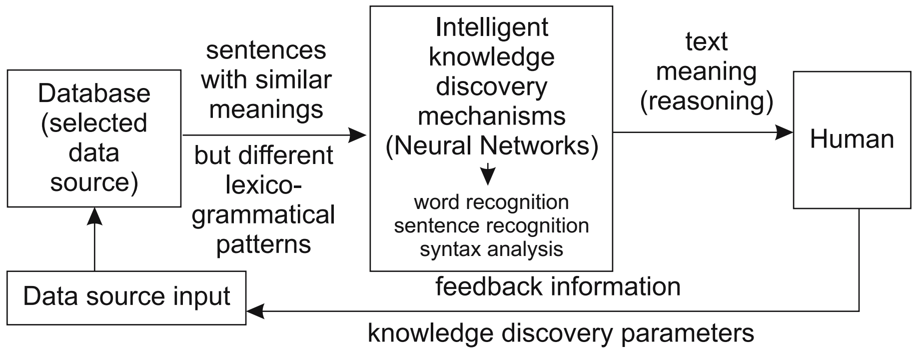

Fig. 1. Steps involved in proposed knowledge discovery in text

The system consists of a selected data source, 3-layer artificial neural networks, network training sets, letter chain recognition algorithms, syntax analysis algorithms, as well as coding algorithms for words and sentences.

2 The State of the Art

Knowledge discovery is a growing field: There are many knowledge discovery methodologies in use and under development. Some of these techniques are generic, while others are domain-specific.

Learning algorithms are an integral part of knowledge discovery. Learning techniques may be supervised or unsupervised. In general, supervised learning techniques enjoy a better success rate as defined in terms of usefulness of discovered knowledge. According to [1,2], learning algorithms are complex and generally considered the hardest part of any knowledge discovery technique. Machine discovery is one of the earliest fields that has contributed to knowledge discovery [4]. While machine discovery relies solely on an autonomous approach to information discovery, knowledge discovery typically combines automated approaches with human interaction to assure accurate, useful, and understandable results.

There are many different approaches that are classified as knowledge discovery techniques [16]. There are quantitative approaches, such as the probabilistic and statistical approaches. There are approaches that utilize visualization techniques. There are classification approaches such as Bayesian classification, inductive logic, data cleaning/pattern discovery, and decision tree analysis [2,4]. Other approaches include deviation and trend analysis, genetic algorithms, neural networks, and hybrid approaches that combine two or more techniques.

The probabilistic approach family of knowledge discovery techniques utilizes graphical representation models to compare different knowledge representations [7]. These models are based on probabilities and data independencies. The statistical approach uses rule discovery and is based on data relationships. An inductive learning algorithm can automatically select useful join paths and attributes to construct rules from a database with many relations [3]. This type of induction is used to generalize patterns in the data and to construct rules from the noted patterns.

Classification is probably the oldest and most widely-used of all the knowledge discovery approaches [3,7,16]. This approach groups data according to similarities or classes. There are many types of classification techniques e.g. the Bayesian approach, pattern discovery and data cleaning, and the decision tree approach. Pattern detection by filtering important trends is the basis for the deviation and trend analysis approach. Deviation and trend analysis techniques are normally applied to temporal databases [4,6].

Neural networks may be used as a method of knowledge discovery. Neural networks are particularly useful for pattern recognition, and are sometimes grouped with the classification approaches. A hybrid approach to knowledge discovery combines more than one approach and is also called a multi-paradigmatic approach. Although implementation may be more difficult, hybrid tools are able to combine the strengths of various approaches. Some of the commonly used methods combine visualization techniques, induction, neural networks, and rule-based systems to achieve the desired knowledge discovery. Deductive databases and genetic algorithms have also been used in hybrid approaches.

3 Method Description

In the proposed knowledge discovery system shown in Fig. 2, sentences are extracted from the database. The separated words of the text are the input signals of the neural network for recognizing words [5]. The network has a training file containing word patterns. The network recognizes words as the sentence components, which are represented by its neurons in Fig. 3. The recognized words are sent to the algorithm for coding words [12]. Then, the coded words are transferred to the sentence syntax analysis module. It is equipped with the algorithm for analysing and indexing words. The module indexes words properly and then they are sent to the algorithm for coding sentences [13]. The commands are coded as vectors and they are input signals of the sentence recognition module using neural networks. The module uses the 3-layer Hamming neural network in Fig. 4, either to recognize the sentence in order to find out its meaning or just does not recognize the sentence. The neural network is equipped with a training file containing patterns of possible sentences whose meanings are understood.

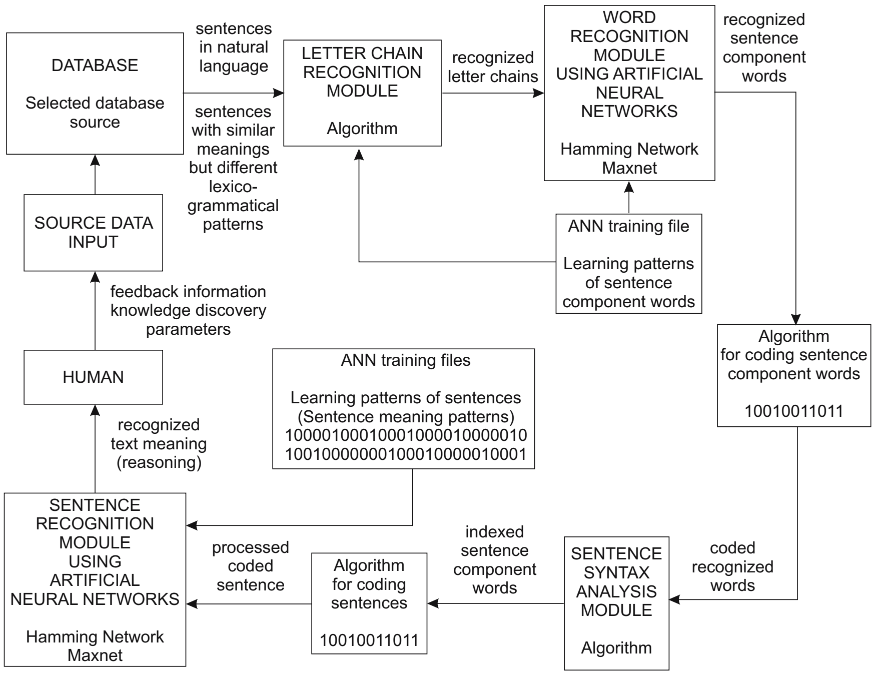

Fig. 2. Scheme of the proposed system for knowledge discovery in text

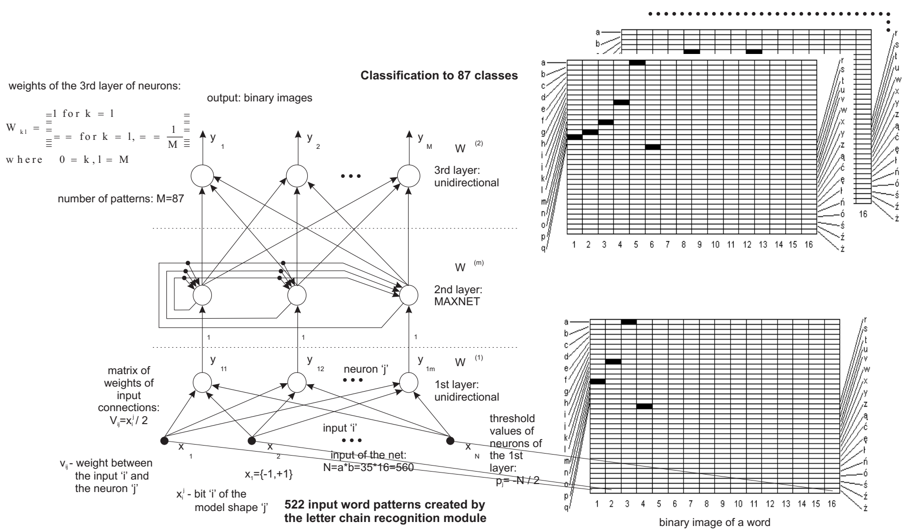

Fig. 3. Scheme of the 3-layer neural network for word recognition

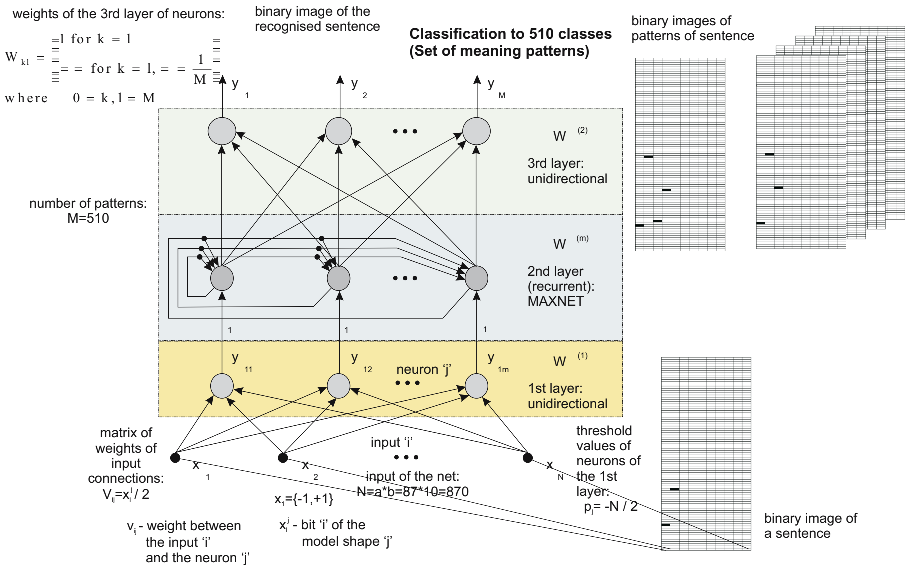

Fig. 4. Scheme of the 3-layer neural network for sentence recognition

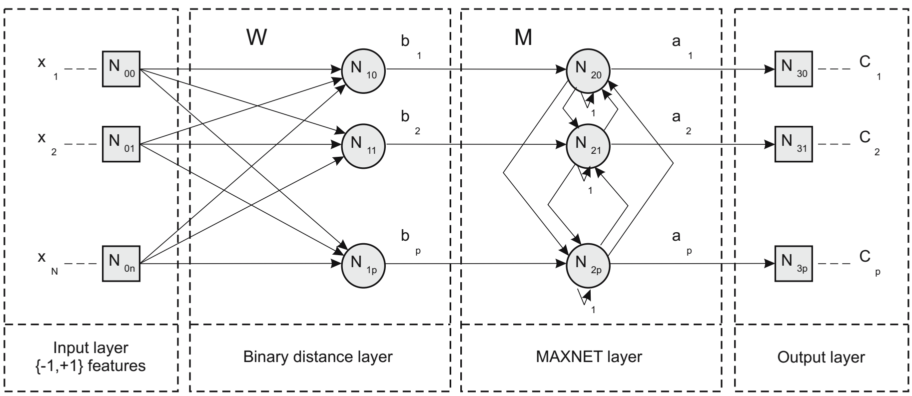

Fig. 5. Structure of the Hamming neural network as a classifier-expert module

Because of the binary input signals, the Hamming neural network is chosen in Fig. 5 which directly realizes the one-nearest-neighbour classification rule [9,10,11]. Each training data vector is assigned a single class and during the recognition phase only a single nearest vector to the input pattern x is found and its class C_i is returned. There are two main phases of the operation of the expert-network: training (initialization) and classification. Training of the binary neural network consists of copying reference patterns into the weights of the matrix W_{pn}, as follows (1):

$$w_i = x_i\,, \quad 1 \le i \le p \tag{1}$$

where p is the number of input patterns-vectors x, each of the same length n, w_i is the i-th row of the matrix W of dimensions p rows and n columns. For given n the computation time is linear with the number of input patterns p.

The goal of the recursive layer $N_{2.}$ is selection of the winning neuron.The characteristic feature of this group of neurons is a self connection of a neuron to itself with a weight m_{ii}=1 for all 1 $\le i \le p$, whereas all other weights are kept negative. Initialization of the $N_{2.}$ layer consists in assigning negative values to the square matrix M_{pp} except the main diagonal. Originally Lippmann proposed initialization [8] (2):

$$\begin{array}{l} m_{kl} = -(p-1)^{-1} + \xi_{kl} \quad for\ k \neq l\,, \quad 1\ for\ k = l \\ where \quad 1 \le k, l \le p,\ p > 1 \end{array} \tag{2}$$

where ξ is a random value for which $|\xi| \ll (p-1)^{-1}$. However, it appears that the most efficient and still convergent solution is to set equal weights for all neurons $N_{2.}$ which are then modified at each step during the classification phase, as follows (3):

$$\begin{array}{l} m_{kl} = \varepsilon_k(t) = -(p-t)^{-1} \quad for\ k \neq l\,, \quad 1\ for\ k = l \\ where \quad 1 \le k, l \le p,\ p > 1 \end{array} \tag{3}$$

where t is a classification time step. In this case the convergence is achieved in $p-1-r$ steps, where $r > 1$ stands for the number of nearest stored vectors in W.

In the classification phase, the group $N_{1.}$ is responsible for computation of the binary distance between the input pattern z and the training patterns already stored in the weights W. Usually this is the Hamming distance (4):

$$b_i(z, W) = 1 - n^{-1} D_H(z, w_i)\,, \quad 1 \le i \le p \tag{4}$$

where $b_i \in [0, 1]$ is a value of an i-th neuron in the $N_{1.}$ layer, $D_H(z, w_i) \in \{0, 1, ..., n\}$ is a Hamming distance of the input pattern z and the i-th stored pattern w_i (i-th row of W).

In the classification stage, the $N_{2.}$ layer operates recursively to select one winning neuron. This process is governed by the following equation (5):

$$a_i[t+1] = \varphi\left(\sum_{j=1}^{n} m_{ij} a_j[t]\right) = \varphi\left(a_i[t] + \sum_{j=1,\, i \neq j}^{n} m_{ij} a_j[t]\right) \tag{5}$$

where $a_i[t]$ is an output of the i-th neuron of the $N_{2.}$ layer at the iteration step t, φ is a threshold function given as follows (6):

$$\varphi(x) = x \quad for\ x > 0\,, \quad 0\ \ otherwise \tag{6}$$

Depending on the chosen scheme (2)-(3) of the m_{ij} weights in (5), we obtain different dynamics of the classification stage. The iterative process (5) proceeds up to a point where only one neuron has value different than 0 - this neuron is a winner.

4 Research Results

The dataset for the tests carried out contained a database of 1500 sentences, files consisting of 522 letter chains, 87 word training patterns and 510 sentence meaning training patterns. The first test measured the performance of the sentence meaning recognition with the sentence recognition module using artificial neural networks as a set of words recognised earlier in Fig. 6.

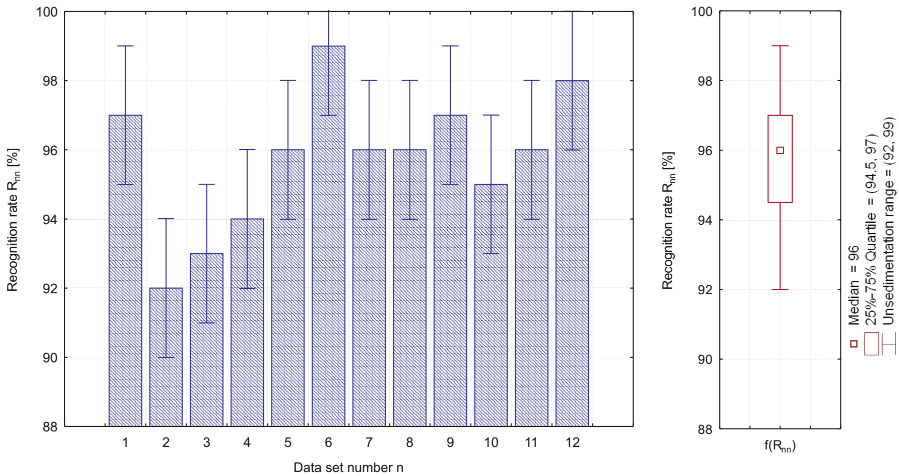

Fig. 6. Sentence meaning recognition rate as a set of words recognised earlier

As shown in Fig. 7a, the ability of the implemented neural network for word recognition to recognise the word depends on the number of letters. The neural network requires a minimum number of letters of the word being recognized as its input signals. As shown in Fig. 7b, the ability of the neural network for sentence meaning recognition to recognise the sentence depends on the number of sentence component words. Depending on the number of component words of the sentence, the neural network requires a minimum number of words of the given sentence as its input signals.

5 Conclusions and Perspectives

Knowledge discovery is a rapidly expanding field with promise for great applicability. Knowledge discovery purports to be the new database technology for the coming years. The need for automated discovery tools had caused an explosion in research.

The motivation behind using the binary neural networks in knowledge discovery comes from the possible simple binarization of words and sentences, as well as very fast training and run-time response of this type of neural networks.

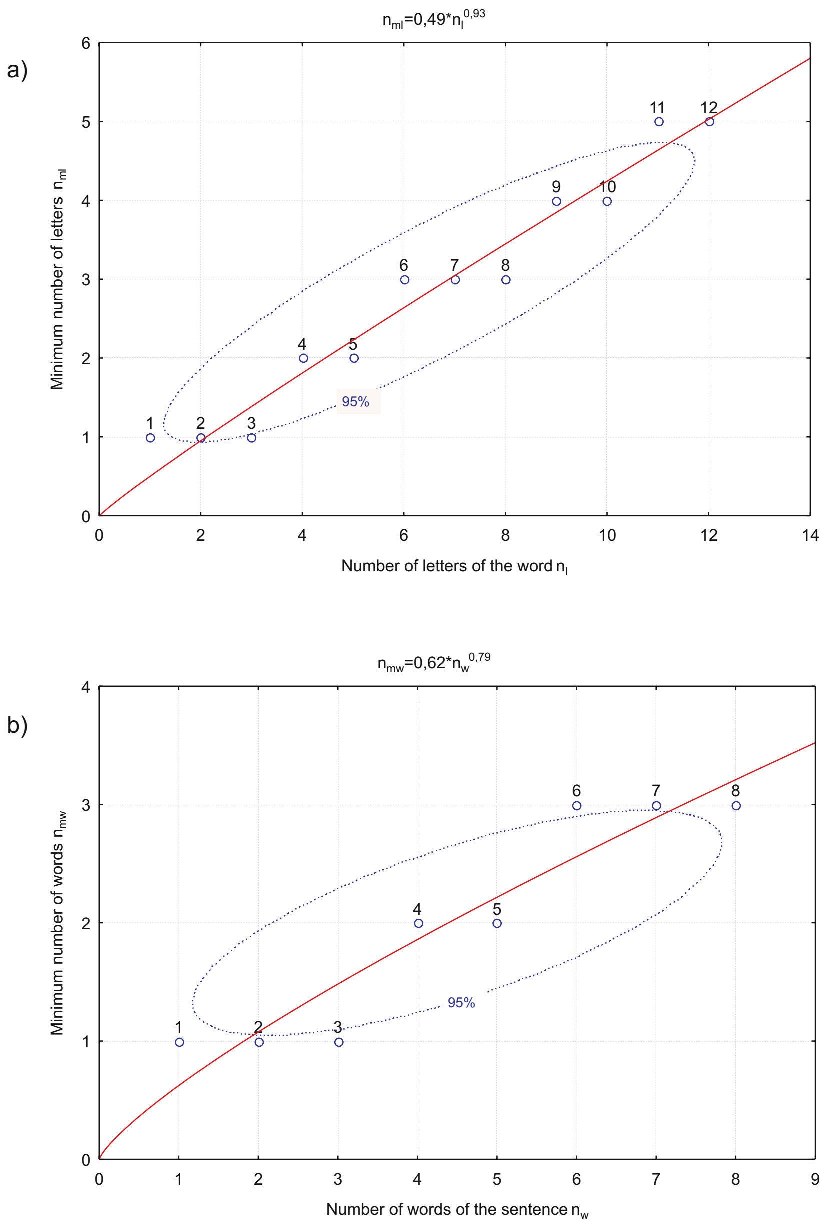

Fig. 7. a) Sensitivity of word recognition: minimum number of letters of the word being recognized to number of word component letters; b) Sensitivity of sentence meaning recognition: minimum number of words of the sentence being recognized to number of sentence component words

Application of binary neural networks allows for recognition of sentences in natural language with similar meanings but different lexico-grammatical patterns, which can be encountered in documents, texts, vocabularies and databases. The presented methods can be easily extended.

It is anticipated that commercial database systems of the future will include knowledge discovery capabilities in the form of intelligent database interfaces. Some types of information retrieval may benefit from the use of knowledge discovery techniques. Due to the potential applicability of knowledge discovery in so many diverse areas there are growing research opportunities in this field.

References

1. Berry, M., Linoff, G.: Mastering Data Mining. John Wiley & Sons (2000)
2. Dunham, M.H.: Data Mining Introductory and Advanced Topics. Prentice Hall (2003)
3. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. 2nd edition. Morgan Kaufmann (2006)
4. Hand, D.J., Mannila, H., Smyth, P.: Principles of Data Mining. MIT Press (2000)
5. Kacalak, W., Douglas Stuart, K., Majewski, M.: Intelligent Natural Language Processing. International Conference on Natural Computation ICNC2006. Xi'an, China. Lectures Notes in Artificial Intelligence **4221** (2006) 584-587
6. Kantardzic, M.: Data Mining: Concepts, Models, Methods, and Algorithms. Wiley-IEEE Press (2002)
7. Larose, D. T.: Discovering Knowledge in Data: An Introduction to Data Mining. John Wiley & Sons, Inc. (2004)
8. Lippman, R.: An Introduction to Computing with Neural Nets. IEEE Trans. Acoustic, Speech, and Signal Processing **3** (1987) 4-22
9. Majewski, M., Kacalak, W.: Intelligent Layer of Two-Way Voice Communication of the Technological Device with the Operator. Lectures Notes in Artificial Intelligence **3683** (2005) 930-936
10. Majewski, M., Kacalak, W.: Intelligent Human-Machine Voice Communication System. Engineering Mechanics International Journal for theoretical and applied mechanics **12** (2005) 193-200
11. Majewski, M., Kacalak, W.: Automatic Recognition and Evaluation of Natural Language Commands. Lecture Notes in Computer Science **3973** (2006) 1155-1160
12. Majewski, M., Kacalak, W.: Natural Language Human-Machine Interface using Artificial Neural Networks. Lecture Notes in Computer Science **3973** (2006) 1161-1166
13. Majewski, M., Kacalak, W., Douglas Stuart, K.: Selected Problems of Intelligent Natural Language Processing. International Journal of Research on Computing Science **20** (2006) (ISSN 1665-9899)
14. Stuart, K.: Corpus Linguistics: Theory, Method and Applications. XXIV International AESLA Conference, Language Learning, Language Use and Cognitive Modelling: Applied Perspectives across Disciplines. Madrid, Spain (2006)
15. Stuart, K.: A Software Application for Text Analysis. International Conference on Engineering Education, Valencia, Spain (2003)
16. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Pearson Addison Wesley (2005)