We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Towards an Optimal Decision Support System

Witold Kosiński[1], Grzegorz Dziczkowski[2],
Bruno Golénia[3] and Katarzyna Węgrzyn-Wolska[4]
[1]Polish-Japanese Institute of Information Technology,
[2]Univ Lille Nord de France, Ecole des Mines de Douai, IA,
[3]University of Bristol,
[4] Ecole Superieur d'Ingenieurs en Informatique et Genie des Telecommunicatiom,
[1]Poland
[2,4]France
[3]United Kingdom

1. Introduction

Nowadays the internet is an essential tool for the exchange of information on a personal and professional level. The web offers us a world of prodigious information and has evolved from simple sets of static information to services which are more and more complex. These services cover making purchases, reading ones favorite newspaper, meeting the love of ones life, the possibility of discussion in many different forums and through a blog.

The internet contains a huge amount of information and for most of us it is the first place we look for information, book a plane or a hotel, to buy products or to consult the opinion of other consumers on the products we wish to buy, to read commentaries before choosing a film or going to the cinema, to see other peoples propositions before choosing wedding gifts etc. The principal problem is no longer knowing whether there is information on the web or not but finding it for the information flow is excessively abundant. Another problem which is not specifically linked to the internet but rather to society is the global invasion. We have access to more products than we can comprehend.

A human makes a decision every day; often he/she needs the intervention of a domain expert. Decision support system is concerned at making a decision, in purpose to replace the need of an expert. Traditionally, decision support systems are represented as a set of rules (Boolean of fuzzy) which fire to provide the decision. Currently, such systems are developed under heuristics.

Prediction engines are often developed to offer the user alternative products. People like to see other peoples opinions before forming there own. On line predictions are very useful for customers. Prediction engines algorithms are based on the experience and opinion of other users. The algorithms give very useful results if they can find users with similar results. In order to do this prediction engines need to have an extremely large user profile base.

The general objective of our research is to generate user profiles so that they can be used by predictive algorithms. The profiles concern cinema films and the general objective is to create an autonomous system which will serve as a support for prediction engines. The role of such a system is to find critics, to mark their sentiments automatically and to create final

profiles. The research activity concerns the notation of the opinions which is an ambiguous task. We propose approaches based on a deep linguistic treatment so as to improve the attribution of each mark to the intensity of an opinion.

In the first part of the chapter we describe the study and development of a system designed for the evaluation of sentiments within cinema reviews. In order to improve the application results of predictive algorithms the objective of this system is to supply a support system for the prediction engines analyzing users profiles. The system evaluates and automatically attributes a mark to the opinion expressed in the cinema reviews. Presented work in this part is in the realm of Opinion Mining. Our system uses three different methods for the classification of opinions. We present two new methods; one founded on pure linguistic knowledge and the other on a combination of statistic and linguistic analysis. We wish to show the advantages of deep linguistic analysis which is less commonly used than statistical analysis in the domain of sentiment analysis.

In the second part we show how the construction of a decision support system can be turned into the construction of an approximator of a function. We are using artificial neural networks and fuzzy inference system. To do so, we assume multidimensional data represented by attributes with their associated decision. Then, we gather them into a set TRE, called at other occasions the training set. Then different grouping procedures are applied to the set of data. With each element of three families of clusters membership functions are attached to. The creation of our reasoning system is performed in two stages. In the first stage feed-forward neural networks are constructed and trained on each cluster. After, learning the parameters of the feed-forward neural networks, we design a fuzzy inference system of Takagi-Sugeno-Kang type. Our fuzzy rules are built on each triple of clusters from those three families of clusters. Finally, the final decision of our system is an aggregation of outputs of the networks as the consequences of three-conditional fuzzy rules. We present results on a test set TES according to a standard benchmark. Our results show the benefits of parallel computation and comparison between different membership functions: polynomials of third and second degree, and generalized Gaussian functions. We also show the issues resting in our approach which should be solved to get an optimal decision support system.

2. Knowledge base for decision support system

Decision Support Systems (DSS) is a specialized information system including knowledge-based systems that support decision-making functions. Three fundamental components of DSS architecture are: the database (or knowledge base), the model (i.e., the decision context and user criteria), and the user interface. In this section we presented an input module of decision support system responsible of collecting and preparing the knowledge base for DSS.

2.1 Knowledge collecting system

System presented in this section carries out the automatic collection, evaluation and rating of the textual opinion written in natural language due to the preparation of the huge knowledge base for the DSS. First, the system searches and retrieves texts of opinions from the Internet. Subsequently, the system carries out an evaluation and rating of those texts. Finally, the system automatically creates users profile database. Our system uses linguistics and statistic methods for classifying opinions. All retrieved relevant information used by a second module responsible of opinion marking it means that after collecting the text, we

will assign notes by using the classifiers. The classifiers provide ratings from users' feelings. The classifier uses three different methods for assigning a mark to the text. Those methods are based on different approaches of corpus classification.

2.2 Data needs

An efficient Decision Support Systems requires a lot of effective information to analyze. The need to have an extremely large database is crucial for the DSS algorithms. Processes like collecting, analyzing, tracking and retrieving any piece of information, which may hold value for DSS is very important. The source of information also varies, information can comes from centrally maintained data systems, from organized file systems or remote storage, from networked systems and more and more frequently at present from external open web sources. We are interested in this last source, web that offers us a lot of prodigious information and which has evolved from simple sets of static information to services which are more and more complex.

2.3 Opinion detection

One of the kinds of the data that we are interested in is the profile of the human, it means all of the information which describes one person, like the taste, custom and habits. We can find and extract a lot of such information from the Web. The objective is to extract maximum of data concerning each person. While the internet becomes an essential tool for search and the exchange of information on a personal and professional level, we can find and extract from it a lot of information which describe the human custom and taste. In the goal of understanding the human opinions and sentiments written in natural language, Opinion Mining knowledge was necessary to implement. For this reason, we presented in this section our new approaches to automatically detect opinion from the text. We have proposed two kinds of classifications: first based on the group conduct and second linguistic. Then, we have compared our two approaches with the approach generally used in this field, it means, the statistical classification based on Naive Bayes classifiers.

3. Related works

Techniques used for sentimental analysis Das & Chen (2001), are known as Opinion Mining Dave et al. (2003). The research in this field covers different subjects, in particular the learning of words' or expressions' semantic orientation, the sentimental analysis of documents and opinions and attitudes analysis regarding some subjects or products Lewis & Haues (1994), Joachims & Sebastiani (2002).

3.1 Opinion mining and sentiments analysis

In order to determine the complexity of opinion marking, we are going to take an example of a review. The example is:

> "It's A Wonderful Life. I've only met 2 people in real life and 1 person on the IMDB who hates this one. My favorite film ever!"

As we have noticed, the review is composed of three phrases, which have opposite polarity. Even though, we can easily deduct that the first sentence is the movie title, *Wonderful life*, we will have two subjective phrases but hard to mark correctly. The last phrase is rather easy to

mark: *"My favorite film ever!"*. However, there is a problem for the marking of the phrase: *I've only met... who hates this this one*, because a statistical study shows us that the polarity is negative for this phrase but in fact the polarity is positive and with high intensity.

Sentiments can often be expressed in a subtle manner, which creates a difficulty in the identification of the document units when considering them separately. If we consider a phrase, which indicates a strong opinion, it is hard to associate this opinion with keywords or expressions in this phrase. In general, sentiments and subjectivity are highly sensitive to the context and dependent of the field.

Moreover, on the Internet, everyone is using its own vocabulary, which adds difficulties to the task; even though it is in the same field. Furthermore, it is very hard to correctly allocate the weight of phrases in the review.

It is not yet possible to find out an ideal case of sentiment marking in a text written by different users because it does not follow a rule and it is impossible to schedule every possible case. Moreover, frequently the same phrase can be considered as positive for one person and negative for another one.

3.2 Different approaches for the sentiment analysis

The semantic orientation of words has been elaborated first of all for the adjectives Hatzivassiloglou (1997), Whitelaw et al. (2005). The works on the subjectivity detection have revealed a high correlation between the adjective presence and the subjectivity of phrases Hatzivassiloglou & Wiebe (2000). This observation has often been considered as the proof that some adjectives are good sentiment indicators. A certain number of approaches based on the adjectives presence or polarity have been created in order to deduct the text subjectivity or polarity.

Turney's approach

One of the first approaches has been proposed by Turney (2002) and can be presented in four stages:

- First of all, there is a need to make phrase segmentation (part-of-speech)
- Then, we are putting together adjectives and adverbs in series of two words
- We apply afterwards SO-PMI (Semantic Orientation Using Pointwise Mutual Information) in order to calculate the semantic orientation of each detected series,
- Finally, we carry out a text classification as positive or negative by calculating the average of all orientations.

Results obtained by this approach are different compared to the field: for cars= 84%, for banking documents= 80% and for cinematographic reviews= 65%. The fact that adjectives are good opinion preachers is not diminishing the other words signification. Pang et al. (2002), in the polarity study of cinematographic criteria, have demonstrated that using only adjectives as characteristics gives result less relevant than using the same number of unigrams.

Pang's approach

Pang & Lee (2004) are proposing another approach for the polarity classification of cinematographic reviews. The approach is composed of two stages (first the detection of subjectivity is performed, then the detection of polarity is performed only on subjective sentences) Figure 1. The first goal is to detect the document's parts, which are subjective. Then, they are using the same statistical classifier to detect the polarity only on subjective

fragments detected previously. Instead of doing the subjectivity classification for each phrase separately, they admit that they can see a certain degree of continuity in the phrases subjectivity - a writer generally is not changing often between the fact to be subjective or objective. They give preferences in order to have proximity phrases, which have the same level of subjectivity. Every phrase in the document is then labeled as subjective or objective in the process of collective classification.
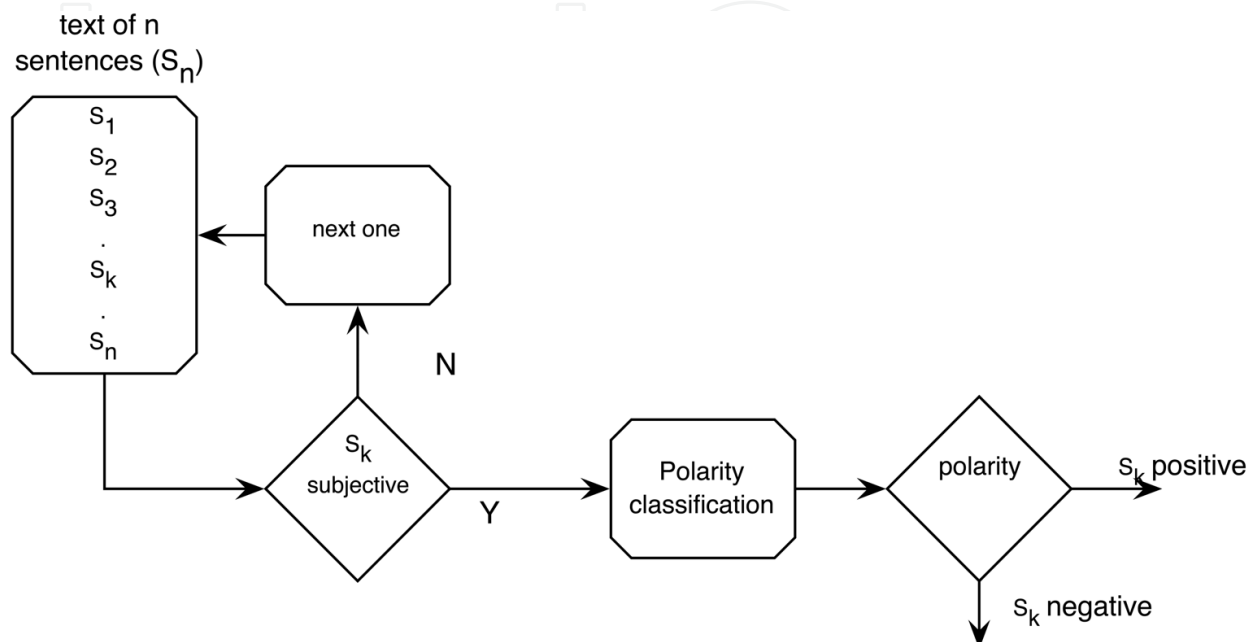

Fig. 1. Pang's approach - Utilization of the same classification technique for the detection of subjectivity and afterwards of phrases polarity labeled as subjective.

4. General system architecture

Our goal is to create the huge database of profiles which interacts with the taste of users. Principal modules of our architecture are [Figure 2]: research and collect of texts on the internet (Web Spider), text analyze, opinion detection, attribution of a mark for each text and storage of all the interesting information in database.
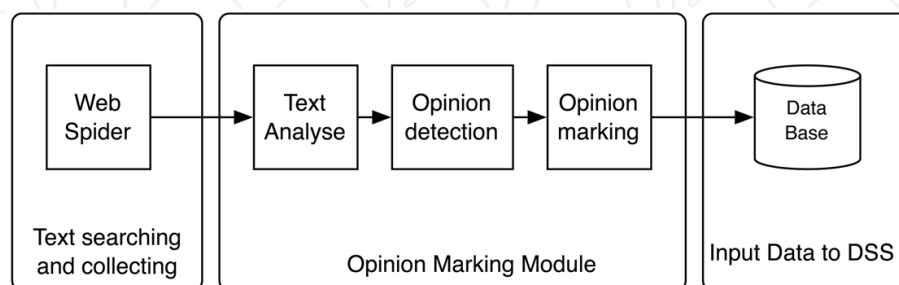

Fig. 2. Input Data architecture.

The first module collects information on the internet. In fact, we use spider retrieved the web pages for subsequent search purposes. Than all of these pages are analyze in the *Opinion marking module* described in [section 4.1]. The results of this analyze with all the appropriated information is stored in knowledge base.
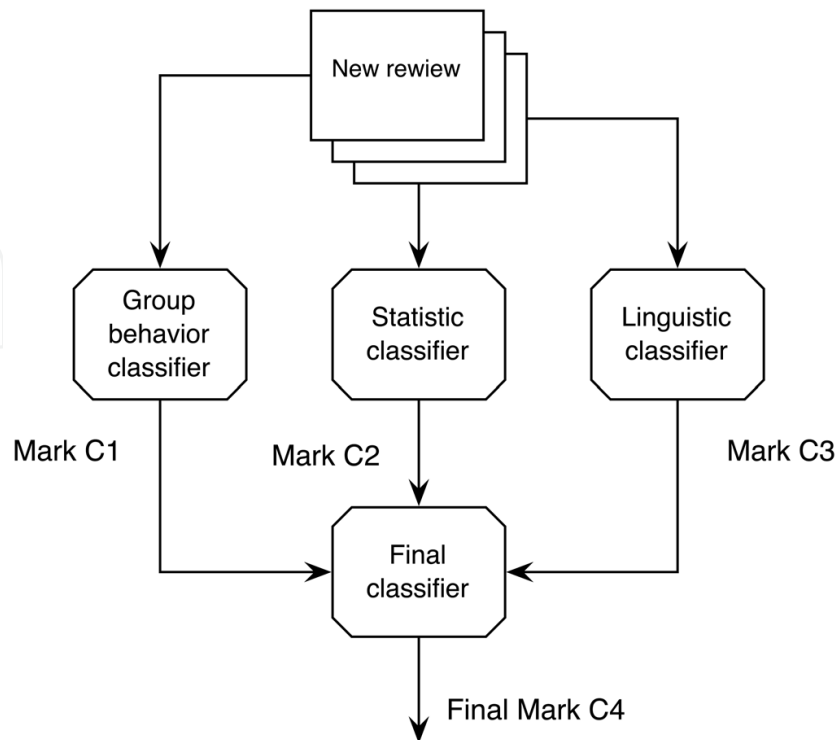
4.1 Opinion marking module

Fig. 3. The three principal marking modules

The Opinion Marking module [Figure 3] proceeds three different methods for the attribution of a mark:

- The group behavior classifier [section 4.2]
- The statistical classifier [section 4.3]
- The linguistic classifier [section 4.4]

Those measures are based on different approaches of document classification. Secondly, we have developed, for each method, a classifier, which assign separately the mark Dziczkowski & Wegrzyn-Wolska (2008b), Dziczkowski &Wegrzyn-Wolska (2007). We have obtained, therefore, three marks for each text, which can be different. We have used, finally, another classifier, which assign the final mark, based only on the three marks attributed previously in the classification process Dziczkowski &Wegrzyn-Wolska (2008a). For the calculation of the final mark, we have used the values of the three marks previously attributed and their probabilities.

On a research point of view, the most important part of the system conceived is the opinion marking module.

4.2 Group behavior classifier

In this section, we present the classifier used for the opinion marking. The general approach is based on the verification that opinions, having the same associated mark, have common characteristics. Then, we determine a behavior, for those having the same mark. We determine therefore, the general behavior of each group (5 groups corresponding to five different opinion marks, 5 groups correspond to the users' notation of our learning base). We have a data set composed of 300 opinions already marked (828 sentences for a group number 5, 588 sentences

for group 4, 657 sentences for 3, 431 for 2, and 1130 for the group number 1). We have gathered together all the opinions according to their mark. We obtain, then, five different groups. Afterwards, we have tried to determine typical characteristics of each group. We have defined all parameters, which can characterize the group behavior such as:

- Characteristic words,
- Characteristic expressions,
- The phrase length,
- The opinion size,
- The frequency of several words repetition,
- The negation,
- The number of punctuation signs (!, ;), ?)

The choice of criteria that we have kept for the analysis of the group behavior has been done in an empirical way. First of all, by analyzing the texts corpus, we have defined criteria that seems interesting and that could determine group behaviour. Then, we have tested those criteria on a training base containing a thousand of opinions. If results showed differences between groups, we considered those criteria as valid criteria for our research work. In this approach, we present the statistical study on linguistic data. The training base has been used for the opinion analysis, of those having the same mark, in order to find characteristics, which determine the behavior of each group. Each approach used in our research is based on different characteristics, in order not to repeat them in the classification process. However, we have borrowed semantic classes from the linguistic approach for the creation of the words list characteristics. The utilization of those data is different in those two groups. After having select criteria that characterize mark groups, we have analyzed the corpus in order to obtain statistical results. Results show huge differences between the characteristics of those groups. The creation of the global behavior of each groups, enable to determine the group in which a new opinion is. We have calculated for new opinions, the distance between its characteristics and those of the groups.

4.3 Statistical classifier

In this section, we propose a general approach used in the sentiment analysis. We use this method to compare results of our approaches with the same training base. The way to carry out a classification is to find a characteristic of each category and to associate a belonging function. Among known methods, we can mention Bayes classifiers and the SVM method. We have obtained better results for the classifier of Naive Bayes, we are going therefore to based ourselves on this classifier. In our research work, we have used this classifier first of all to determine the subjectivity or objectivity of phrases, then in order to attribute a mark to subjective phrases of the opinion. The general process needs the preparation of training base for two classifiers to attribute a mark. The intermediate stages are the followings:

- Preprocessing and lemmatization,
- Vectorization and calculation of complete index,
- Constitution of training base for each classifier,
- Reduction of the index dedicate to the classifier,
- Addition of synonyms,
- Classification of texts

We are using, for the attribution of a mark to the sentiment via a statistical approach, two classifiers: a first one to filter the objective and the subjective phrases and a second one to

mark the opinion. The marking is done only on subjective phrases. Those classifiers rely on a vectorial representation of the text of the training base. This vectorial representation needs in a first time a linguistic preprocessing for the segmentation of the phrase, for the lemmatization and for the suppression of all words, which has no impact on the sense of the document. This preprocessing has been carried out for the linguistic classifier.

We carry out the preprocessing thanks to the application Unitex. We are already disposing of linguistic resources prepared for this task as, for example, the grammar of the phrase segmentation or dictionaries. Then, we take off term with no sense, such as defined or undefined articles or prepositions. We can conduct this task because those grammatical elements have a low impact on the text sense as, for example, on the opinion, contrary to adverbs, which give a high contribution to value judgment. Afterwards, on a training corpus, we calculate the dimension of the vectorial space of the text representation in order to carry out all lemma enumeration - the entire index. Each document is then represented by a vector, which contains the number of occurrences of each lemma present in the document. Every document of the training base is represented by a vector, which dimension corresponds to the whole index and components are occurrences frequencies of the index units in the document. Therefore, at this stage of the process, texts are seen as a set of phrases. Now, each phrase is labeled according to the construction of classifiers (the subjective classifier and the marking classifier). Labels correspond to subjective phrases (PS) or objective ones (PO) and the estimating mark attributed to those phrases (N from 1 to 5). A phrase j of the document i is marked as follows:

$$\vec{V}_{D_i P_j} = (f_{D_i P_j 1}, \ldots f_{D_i P_j k}, \ldots f_{D_i P_j |D|}, PS / PO, N) \tag{1}$$

where $f_{D_i P_j k}$ represents the occurrences number of the lemma k in the phrase j of the document i. The stage of the labeling was based on the opinions' marks of the training base and subjective phrases have been labeled manually. This is how we have built the set of training necessary to the determination of classifiers of subjectivity and of sentiment marking.

The last stage of the vectorial representation of the document corpus is the reduction of the entire index dedicate to the classifier. The reduction of the complete index consists in eliminating from the vectorial space of the training base, vectors, which have many components always null. This task enables us to eliminate the noise in the classifier calculation Cover & Thomas (1991). We have used the method of mutual information associated to each vectorial space dimension.

In our works, we have used two classifiers: the classification based on Bayes model and the classification using SVM. The two methods have been tested and the best results (F-score) have been obtained by the Bayes classifiers. It is, as a result, Bayes classifier who was used in the system. In the process of the statistical classification, we have at first classified subjective phrases and then we have attributed a mark.

Interesting phrases to carry out the opinion marking are subjective phrases because there are the only ones which contain the author point of view. For this reason, we have first of all carried out the filtration of subjective phrases. The diagram, which represents those tasks, is shown in the Figure 4.

The process presented enables to filter only subjective phrases, those expressing an opinion. The different stages are as follow:

- The preprocessing consists in carrying out the phrase segmentation, the lemmatization and the elimination in our research of words without sense.
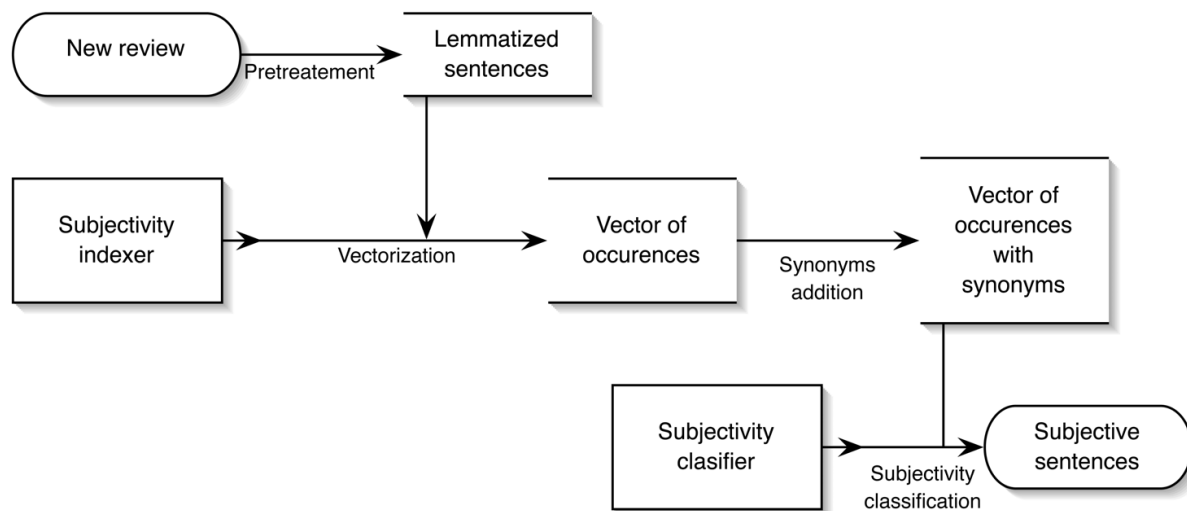
Fig. 4. Subjectivity classification - the classification steps

- The vectorization consists in putting all phrases in the form of vector of occurrences and to reduce the complete index.
- The addition of synonym consists to add terms (synonyms) in the vector of occurrences thanks to the linguistic analysis.
- The subjectivity classification consists in gathering together phrases in subjective or objective phrases. The classification is based on Bayes theorem. For the rest of the classification (marking), we keep only subjective phrases.

After carrying out the subjectivity classification, we only keep subjective phrases. We conduct a classification in order to be able to attribute a mark to those phrases of each analyzed opinion. The diagram representing those tasks is presented in Figure 5. The process presented enables to attribute a mark to phrases classified in the subjective phrases. The marking varies between 1 and 5. The stages are the following ones:

- The vectorization and the reduction of the complete index dedicated to the classification of the marking
- The addition of synonyms
- The marking classification, which consists in putting together phrases according to the sentiment intensity. Marks are between 1 and 5.

At this stage of the process, we obtain marks associated to every subjective phrase. The global mark of an opinion of the statistical classification is the arithmetical average of all the phrases of this opinion.

4.4 Linguistic classifier

We carry out marking on a scale going from 1 to 5. We have created for the linguistic approach a grammar rule for each of those groups. This grammar is based on texts' analysis of the training base, which contains approximately 2000 phrases for each mark (the same database than for the other classifiers). The principal goal of the linguistic classifier is the attribution of a mark according to sentiment. The marking is done phrase by phrase. The texts' study of the training base has been carried out in the aim of creating grammar rules for each mark (in this case, the mark is between 1 and 5). Five grammars have been created, one for each mark. Each grammar contains a huge number of rules taken from local grammar. For each grammar, more than thirty local grammars have been created. The
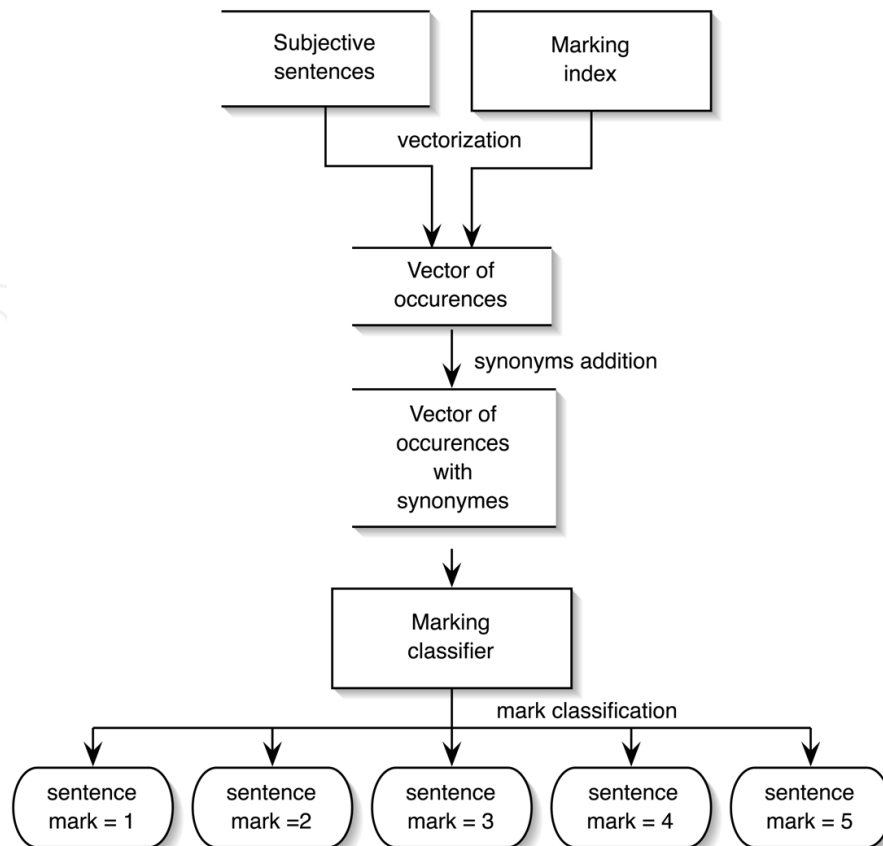
Fig. 5. Subjectivity classification - the classification steps

analysis is done phrase by phrase to attribute a mark to a new text in order to find a rule (from our rules base) corresponding to the studied phrase. At the end of this processing, we obtain phase of the new studied text with matching grammar rules. The final mark of this classification is the average of marks corresponding to general grammars.

The construction of local grammar has been carried out manually via phrases analysis of the texts having the same associated mark. Local grammar can not be too general because this tends to add ambiguity to results. However, if the grammar is too specific and complex, the use of this grammar is indeterminate because silence grows in a significant way. Grammars have been created to detect the opinion polarity and intensity in a phrase thanks to the local grammars form, which constitute a general grammar for each marking group. Research works are based only on local grammars form. Other characteristics purely statistical like words or characteristic expressions, phrase size, words frequency, words repetition, the number of punctuation signs and so on, are not taken into account. Of course, characteristic words are in dictionaries with semantic categories and in local grammar, but this approach is a linguistic processing (grammar is necessary) not a statistical one (like the two other classifiers).

The creation of local grammar is a tiresome task. Grammars used in our system have been created in an empirical way. We have carried out in the following way: first of all, we have constructed general grammars, then we added a complexity level to the linguistic analysis and we have made tests. After those tests, we have repeated the process (addition of a complexity level). For each level, we have conducted tests and calculated the F-score. The final result of grammars rules forms have been chose in order to obtain the best result of F-

score. Unfortunately, we can not be sure of the fact that our choice is the most coherent one. We have taken into account the fact that each classifier presented in our system should have its own criteria and characteristics. It is important to mention that the linguistic classifier provide the best results. We can observe, in particular, that the precision parameter is better than the one obtained by using other approaches.

5. Final classifier

Until now, we have presented three different methods to attribute a mark. Thus, we obtain three different estimations (one for each classifier). The marking is carried out each time in a different way. Marks are therefore not always the same. As we are obtaining three different marks, another problem consists in conducting the final marking in order to attribute only one mark to the text. We need a final classification to obtain the final mark.

5.1 Neural network classifier
We have observed that, if we are calculating the final average obtained by the three classifiers, results are less efficient than those obtained by the linguistic classifier.
We have also observed that often a classifier in specific situations gives best results, whereas in other circumstances, it would be another one. For example [Figure 6], we have observed that often when the first classifier gives a mark equal to 2 and the last two ones give a mark of 1, the correct results is 2. As a consequence, the first classifier is determinant in this case. By implementation of neural networks for this stage and by taking into consideration each probability for each score for each classifier we improved our results for 3 to 7% depending on the class.
We are using, for this reason, a final classifier. For this classification we are applying a neural network. The choice of this classifier is justified by the presence of a data base, already annotated, which will be useful for the training base. Moreover, it is easy to implement those data, for it to be used in the training base. The classifier takes into account only the probability of the mark of each classifier. No other characteristics are taken in consideration. This choice is acceptable because we think that we have used all other possible characteristics in the marking process (by using the three classifiers mentioned previously) and we do not wish to repeat those characteristics in the classifications. Furthermore, the utilization of a characteristic of an opinion marking classifier in the final classification can influence the choice of this classifier.
For the entries of the final classifier, we have used marks of the previous classifiers. The marks of each classifier represented by the belonging probability of one of the five marks categories. For example, the linguistic classifier attributes the mark in the following way: the probability that the mark is:
- equal to 5 is $p_5=0,6$
- equal to 4 is $p_4=0,2$
- equal to 3 is $p_3=0,1$
- equal to 2 is $p_2=0,1$
- equal to 1 is $p_1=0$

We have used a neural network to determine the correlation between marks obtained by the three classifiers. We are using the neural network of multilayer perceptron with the algorithm of back propagation with 3 layers. The set of TRE is composed of 1000 reviews
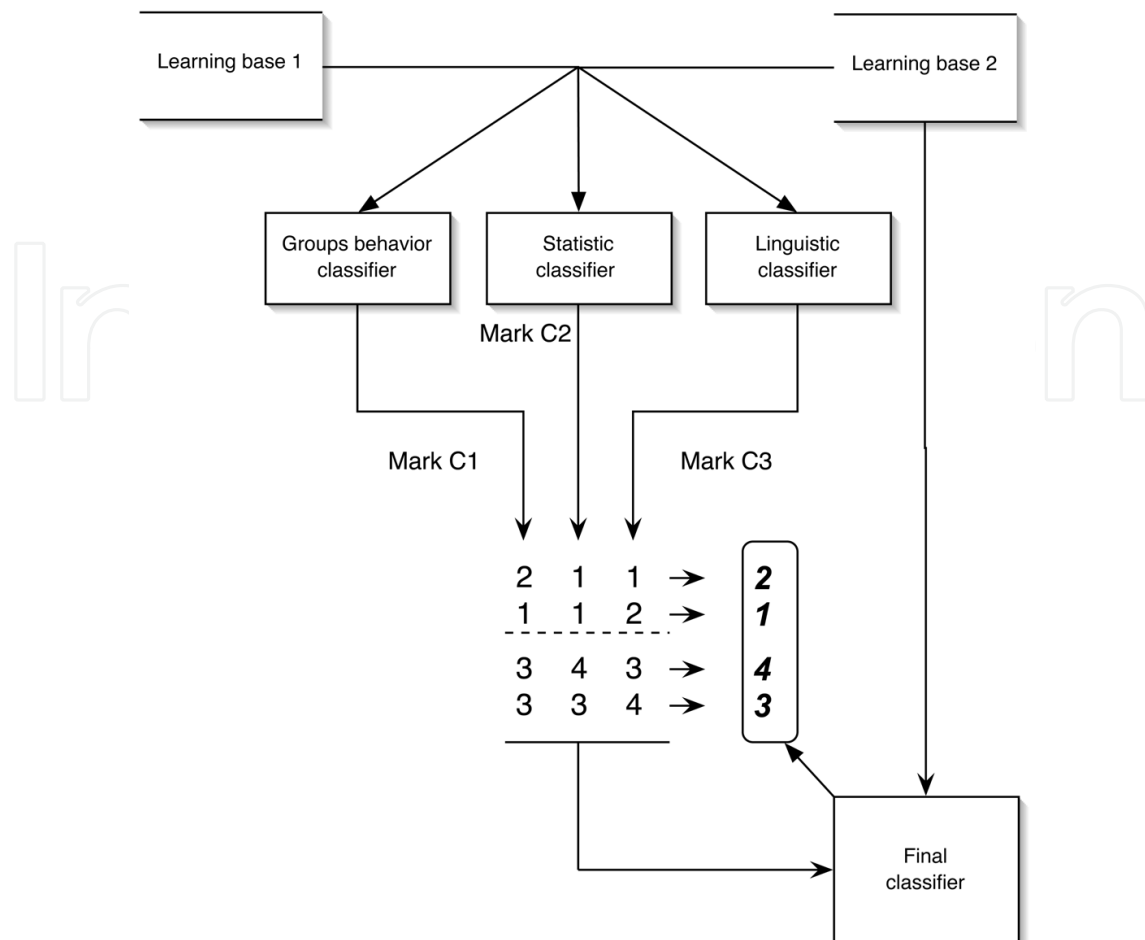
Fig. 6. Final classification- the marks behavior shows the presence of a determinant classifier in some situations

already annotated by the authors (200 reviews for each mark). We have one output (final mark) and 15 inputs (3 marks, each mark is composed by the probabilities of each mark p_1 – p_5). The meaning of theses probabilities are different for each classifier because the calculation of theses probabilities change depending of classifier. For linguistic classifier the mark probabilities are calculated like the sum of the sentences finding by the local grammars annotated like grammars of this mark, taking into account the complexity of linguistic analysis (the local grammars the most complexes so those with higher precision and less recall have the weight more important than the generals local grammars - low precision, high recall). For the group behavior classifier the mark probabilities are calculated according to distance of characteristic of a new review to the characteristics of each group. For statistic classifier the probabilities are calculated like the frequencies of the words for each mark.

We use another training set for learning 3 classifiers and another training set for training our neural network [Fig. 6]. We cannot use the same training set for entire system because the results of each classifier are based on this set. For example the form of local grammars is based on the reviews sentences from training set. In the case of using the same training set the results of classifiers would be incorrectly good, even ideal. And the neural network would learn on wrong examples.

Our system has a parallel architecture and the neural network is the final stage to combine three marks of each classifier. We notice that we can't implement a sequential architecture because the classifiers are based on different characteristic and for this reason it is not possible to approve or improve the results of one classifier by another one.

Temporary results

After carrying out tests, we can observe that we have succeeded to implement an innovative method based on a linguistic classifier. The results obtained after this classification give the better results. We can, therefore, conclude that the deeper linguistic analysis is an important issue in the field of Sentiment Analysis.

We have observed that the best results find for the three approaches were those expressing extreme opinions. Knowing the principle that it is an obligation to dispose of grammars more complex, we have demonstrate that the linguistic classifier gives better results than the statistical or the group behavior ones [Table 1]. The corpus of movie reviews in presented test used by three classifiers contains 2264 sentences for a mark equal to 5, 1957 sentences for 4, 1308 sentences for 3, 1925 sentences for 2 and 1835 sentences for 1. We present in our results the percent value of F-score.

	mark 5	mark 4	mark 3	mark 2	mark 1
Linguistic classifier	85 %	77.6 %	72.9 %	69.6 %	77.8 %
Group behavior classifier	73.8 %	71 %	70.8 %	66.1 %	68.3 %
Statistic classifier	70 %	70.7 %	66.1 %	63.3 %	73 %
Final classifier	83.1 %	81.2 %	74.5 %	72.2 %	81.4 %

Table 1. Classifiers results

Despite the fact that the linguistic classifier enables to obtain the best results, its utilization cannot be universal. Its application to a new field requires the creation of a new linguistic resource base and it is necessary to carry out the deep linguistic analysis again. Those processing are unavoidable because the language is highly dependent of the field.

5.2 Fuzzy inference classifier

In data-driven DSS the main module deals with data, from which knowledge and then rules are constructed. In this part of our chapter we are concerned with the presentation of stages necessary to construct a decision making engine driven by mutidimensional numerical data base.

General idea

At the final stage the engine takes a form of an approximator. For its construction methods developed by the authors and their coworkers are used Weigl & Kosiński (1996), Kosiński et al. (1997), Kosiński, Weigl & Michalewicz (1998), Kowalczyk (1999), Kosiński & Weigl (2000), Kosiński & Kowalczyk (2007), Kosiński et al. (2007), Kosiński & Golénia (2008), Golénia et al. (2009)). In the construction process an unknown function relationship is looked for when a set of training data TRE relevant for the wanted relationship is given.

In the first part of our Chapter we have constructed the classifiers of opinions, having the same associated mark, i.e. common characteristics. We determine therefore, the general behavior of each group (5 groups corresponding to five different opinion marks). We have gathered together all the opinions according to their mark. We obtain, then, five different

groups. We have, for example, defined in subsection 4.2 all parameters, which can characterize the group behavior. However, we have applied three different methods for the attribution of a mark: the group behavior classifier, the statistical one and the linguistic classifier. Those three different methods correspond to some extend to three methods of grouping of the whole data set. Attributed mark can be compared with what will be called here a specified decision of expert. Then the final classifier has been constructed with the use of neural networks in which all three classifiers have their contribution. Notice that in the first part of our chapter different groups of data correspond to different marks attached to each opinion. Grouping procedures proposed here can be applied when output values are continuous even, not necessarily discrete.

In this part we are going to propose a construction of a little more complex module of the finial classifier in which not only neural networks will be present but also a module of multiconditional If–Then fuzzy rules. Those rules will be fuzzy in their premise parts, while their consequent parts will be of functional type. In this way the final classifier will be of a generalized Takagai-Sugeno-Kang type (Jang (1993),Jang & Sun (1993), Weigl & Kosiński (1996), Kosiński & Weigl (2000)). Three grouping procedures applied to the same data set will play the role of three classifiers, which have been described in the first part of our chapter.

The fuzzy sets which appear in each rule are constructed based on characteristic features of each group of data (previously, each was characterized by the same marks). Three different procedures will lead to three families of fuzzy sets which encompass groups of data of common characteristics of output values. Hence the fuzzy rules will be three-conditional. The level of belonging (membership) of individual data to each triple sets will results in the contribution of their output values, as the consequence of the rule, in the overall, final value of the system.

For our purpose and thinking on the construction of decision support information system, we assume that an expert has supplied us with a number, say P, of examples which form TRE, called at other occasions the *training set*. Each element of TRE is an ordered pair: given data vector and given by an expert a specified decision related to it. Each data vector has been encoded in an n-dimensional vector from $\mathbf{R}^n$ while the decision - in a number, so we deal with pairs (x,y) that form the database TRE which is a subset $\mathcal{Z}$ of $\mathcal{X} \times \mathcal{Y}$, n-dimensional input space, say $\mathcal{X}$, and 1D output one, say $\mathcal{Y}$.

Together with that partitioning of the data base we perform the projection on the space $\mathcal{X}$ to get a splitting of the input domain into subdomains called groups of clusters.

The partitioning is the first stage of construction procedure then the next one appears in which on each cluster a feed–forward neural network (FNN) is designed and trained. Moreover, to each cluster a fuzzy set is attached with corresponding membership function in the form of a generalized Gaussian function or polynominal one. Each function depends on scalar variable that measures the distance of the running point from each centroid of the cluster, and possesses in its definition two characteristic features of the cluster: its covariance matrix S and the centroid a. Then a module of three-conditional rules It–Then for a fuzzy inference system is constructed, consequent parts of which are convex combinations of outputs of artificial feed-forward neural networks already constructed. In each premise part of the rules a triple of fuzzy sets attached to the triple of clusters from three partitions (cluster coverings) appear. In general the number of rules is equal to the product of the numbers of clusters of those three coverings. However, one can try to prune some rules during the learning process.

The proposed type of procedure of triple covering can greatly reduce the discrepancy and noise contained in the numerical data of TRE. Then the overall output of the information system is defined as a convex combination of all outputs given by consequent parts of all rules, where the coefficients of the combination are (normalized) levels of activity of individual rules calculated from weighted, aggregated values of their membership functions.

Two parameters characterize each Gaussian membership function. In the case of polynomial functions one parameter is free only. To fix parameters on which membership functions depend the next learning process is performed on the whole set TRE. In this way the membership functions involved in the fuzzy sets of the fuzzy rules can be tuned. The constructed information system has grounded rather well the term a fuzzy–neural system.

Grouping of data

The set TRE is composed of so-called training pairs:

$$\text{TRE} = \{ \, p^q = (\, x^q, y^q) \in \mathbf{R}^{n+1} : q = 1, 2, ...P \} \tag{2}$$

that represent a discrete number of points. Each value x can be regarded as an input value (or independent variable) to which a desired valued y, regarded as an output value, is given. If the data are given as numerical vectors (points) from a subset $\mathcal{Z}$ of $n + 1$-dimensional Euclidean space, then the concept of the similarity can be defined in terms of Euclidean distance function (Euclidean metric) — a very popular and commonly used metric. There are other possible distance functions, like l^p, with $p = 1, 2, ..$, sup-norm and Mahalanobis metrics. The latter one is defined for any two individuals (vectors) u, w form $\mathcal{Z}$ as follows:

$$d_W(\, u, w) := \{(\, u - w) \cdot W^{-1}(\, u - w)\}^{\frac{1}{2}}, \tag{3}$$

where W is a symmetric, positive define matrix and the dot denotes a scalar product between the vectors $u - w$ and $W^{-1}(u - w)$. Note that if $W = I$, where I is the identity matrix, we get the classical Euclidean metric. However, if W is the so-called scatter matrix[1] Duran & Odell (1974), then the Mahalanobis metric is invariant under any non-singular affine point transformation of the set $\mathcal{Z}$. Notice that any normalization procedure made on the numerical data can be represented by a non-singular affine point transformation. However, in the case of training pairs a more complex distance function should be used than Euclidean one, to balance the influence of the independent (x) and dependent (y) variables in the grouping procedure. In our previous publications (Koleśnik et al., 1999; Kosiński, Weigl & Michalewicz, 1998), Kosiński et al. (1997) we have described seed growing approach and

---

[1] If a discrete set of points $\mathcal{Z}$ is given, then its scatter matrix W_Z is defined by

$$W_Z = L^{-1} \sum_{p=1}^{L} (z_p - z^h) \otimes (z_p - z^h),$$

where z_h is the mean point (centriod) of $\mathcal{Z}$ and L is the number of points in $\mathcal{Z}$.

evolutionary method of clustering of elements from the set TRE. Here, we omit those descriptions and assume that they have been already done.

Coverings

Hence, we assume that we have for our disposal three groupings (coverings[2]) of TRE by groups (clusters), i.e. three families of clusters $\{\mathcal{K}_{a1}, \mathcal{K}_{a2}, ..., \mathcal{K}_{aM_a}\}$, $a = 1, 2, 3$ such that

$$\text{TRE} = \bigcup_{h=1}^{M_a} \mathcal{K}_{ah} \text{ for each } a = 1, 2, 3. \tag{4}$$

For each cluster $\mathcal{K}_{ah} \subset \mathbf{R}^{n+1}$ its centroid $p^{ah} = (a^{ah}, d^{ah}) \in \mathcal{Z}$ is defined

$$p^{ah} = \frac{1}{N_{ah}} \sum_{j=1}^{N_{ah}} p_j^{ah}, \ p_j^{ah} \in \mathcal{K}_{ah}, \ \text{for each } j = 1, \ldots, N_{ah}, \tag{5}$$

with N_{ah} as the size of $\mathcal{K}_{ah}$.

Now for each $a = 1, 2, 3$ the projection of each $\mathcal{K}_{ah} \subset \mathbf{R}^{n+1}$ on the input space $\mathcal{X} \subset \mathbf{R}^n$ forms three families $\{\mathcal{X}_{a1}, \mathcal{X}_{a2}, .., \mathcal{X}_{aM_a}\}$, $a = 1, 2, 3$ of subdomains (input clusters, groups) that forms three coverings of the input data $x's$ from $\mathcal{X}$. To each cluster $\mathcal{X}_{ah}$ we relate its scatter (variance-covariance) matrix S_{ah} of dimension $n \times n$ calculated according to the formula Anderberg (1973):

$$S_{ah} = \frac{1}{N_{ah}} \sum_{j=1}^{N_{ah}} (x^{aj} - a^{ah}) \otimes (x^{aj} - a^{ah}), \forall \ x^{aj} \in \mathcal{X}_{ah} \tag{6}$$

where $\otimes$ denotes the tensor product of two vectors. The scatter matrix can be used to measure the efficiency of the grouping in the definition of the fitness function Kosiński, Weigl & Michalewicz (1998).

Let us notice that the matrices S_{ah} are symmetric and positive semi-definite. The eigenvectors corresponding to the vanishing eigenvalues determine the directions of the vanishing „thickness" of the cluster. These observations can be used in reducing the data.

In the previous publications Kosiński & Weigl (1998); Kosiński, Weigl & Michalewicz (1998); Kosiński & Kowalczyk (2007); Kosiński & Golénia (2008) we assumed for simplicity that matrices S_{ah} are nonsingular. Here, we are not going to adapt this assumption. We assume, that in general, the number of positive eigenvalues of particular scatter matrix S_{ah} is less or equal to the dimension n.

Let us take one of those matrices, say S_α corresponding to the cluster $\mathcal{X}_\alpha$ where $\alpha = ah$. If the number m_α of its positive eigenvalues $\lambda_{\alpha i}$ is equal to n, then the standard inverse matrix exits S_α^{-1}, and satisfies the identity

$$S_\alpha S_\alpha^{-1} = S_\alpha^{-1} S_\alpha = I. \tag{7}$$

---

[2] Coverings may differ by the number of groups (clusters) M_a. More coverings may be constructed and then multi-conditional fuzzy rules can be used later on, instead of (17).

On the other hand, if $\lambda_{\alpha 1}, \ldots \lambda_{\alpha m_\alpha}$, with $m_\alpha < n$, are only positive eigenvalues of S_α with the corresponding orthonormal eigenvectors $e_{\alpha i}$, $i = 1 \ldots m_\alpha$, then we may define its pseudo–inverse $\tilde{S}_\alpha^{-1}$ by the formula

$$\tilde{S}_\alpha^{-1} = \sum_{i=1}^{m_\alpha} \lambda_{\alpha i}^{-1} \, e_{\alpha i} \otimes e_{\alpha i}. \tag{8}$$

Later on we will use one symbol S_α^{-1} only, to denote each of them: the standard inverse (7) and the pseudo-inverse (8) of the scatter matrix S_α.

To measure the distance between the placement of a point x from the centroid a^α of the cluster, $\mathcal{X}_\alpha$ we use the Mahalanobis metric (3) based on the scatter matrix

$$d_{S_\alpha}(x, a^\alpha) = \{(x - a^\alpha) \cdot S_\alpha^{-1}(x - a^\alpha)\}^{\frac{1}{2}} =: d_\alpha(x). \tag{9}$$

It is obvious that for each $x \in \mathcal{X}$ we have

$$d_\alpha(a^\alpha) = 0 \text{ and } (d_\alpha(x))^2 = \sum_i \lambda_{\alpha i}^{-1} [(x - a^\alpha) \cdot e_{\alpha i}]^2. \tag{10}$$

It is possible to estimate the maximal value of each d_α on $\mathcal{X}_\alpha$ due to the definition (6), by the inequality, for each $x \in \mathcal{X}_\alpha$

$$d_\alpha(x) = \{\sum_i \lambda_{\alpha i}^{-1} [(x - a^\alpha) \cdot e_{\alpha i}]^2\}^{1/2} \leq \{\frac{\max\limits_j \lambda_{\alpha j}}{\min\limits_i \lambda_{\alpha i}}\}^{1/2} =: d_{\alpha 0}, \tag{11}$$

where N_α denotes the number of elements (points) in the cluster $\mathcal{X}_\alpha$, while m_α is the number of positive (non vanishing) eigenvalues of the scatter matrix S_α and $i, j = 1, \ldots, m_\alpha$; of course we have $m_\alpha \leq n$, in general. This inequality will be used in defining the membership functions of fuzzy sets.

Neural networks

On each cluster from families: $\{\mathcal{X}_{a1}, \mathcal{X}_{a2}, \ldots \mathcal{X}_{aM_a}\}$, $a = 1, 2, 3$, we construct a single mapping neural network which is *Feed-forward Neural Network* (FNN). The α-th FNN is composed of one hidden layer. The number of neurons in the input layer is n, while the number of neurons in the hidden layer is l_α. We restrict ourself to a perceptron type neural network, which is a universal approximator Cybenko (1989); Hornik (1991). Hence, in the α-th FNN each neuron of hidden layer is equipped with an activation function, say σ_α, which is not a polynomial. The activation function can be taken from the family of two–parameter generalized sigmoidal functions, implemented by the present authors in a number of publications (cf. Kosiński, Weigl & Michalewicz (1998); Kosiński et al., (1998); Kowalczyk (1999); Weigl & Kosiński (1996)):

$$\sigma_\alpha(z) = \frac{r_\alpha}{1 + \exp(-\delta_\alpha z)}. \tag{12}$$

The family of parameters r_α and δ_α give more flexibility in the adaptation process. Moreover, their appearance has given a possibility to design a corrected adaptation algorithm for neural network weight vector Gołąbek et al. (1999). Another type of activation functions was proposed in Duch & Jankowski (1997).

More complex neuron networks are also possible. However, in the present paper output layer nodes (neurons) have the identity activation function, and hence neurons can be characterized by constants, only. Hence the output from the α-th network, denoted by y_α can be written as

$$y_\alpha = f_\alpha(x, \Omega_\alpha) = \sum_{j=0}^{l} \omega_{\alpha j}^{ll} \sigma_\alpha (\sum_{i=0}^{n} \omega_{\alpha ji}^{l} x_i), \tag{13}$$

where $\omega_{\alpha j}^{ll}$ and $\omega_{\alpha ji}^{l}$ wIaji, j =1, ..., l_α are constant components of the weight vectors ω_α^{ll}, $\omega_{\alpha j}^{l}$. Here the zero component x_0 of the input variable x is equal to 1 and was introduced to incorporate the bias $\omega_{\alpha j0}^{l}$ under one summation sign. The vector Ω_α incorporate all above components of weight vector together with parameters r_α and δ_α of the activation function.

Each FNN is trained on the data from TRE belonging to the corresponding cluster, i.e. each f_α is trained on the cluster $\mathcal{K}_\alpha$.

5.3 Fuzzy inference system of Takagi-Sugeno-Kang

The next stage of construction of the our information system is to define three families of fuzzy sets: A_{1h} B_{2k} and C_{3l} corresponding to the family of clusters $\mathcal{X}_{1h}$, $\mathcal{X}_{2k}$ and $\mathcal{X}_{3l}$, respectively, in the input domain.

Membership functions

Previously Koleśnik et al. (1999); Kosiński et al. (1997); Kosiński, Weigl & Michalewicz (1998); Kosiński et al., (1998); Kosiński & Kowalczyk (2007) we have used generalized Gaussian functions

$$\mu_\alpha(x) = d^\alpha \exp(-0.5((x - a^\alpha) \cdot S_\alpha^{-1}(x - a^\alpha))^{b^\alpha}), \tag{14}$$

with some parameters d^α and b^α. In our publications (Kosiński & Weigl (1998)) Weigl & Kosiński (1996)) it was shown that by introducing two additional adaptable parameters d^α and b^α one makes the system more flexible. The parameter d^α has to be non-negative and such that the maximum value of the membership function does not exceed 1. A crucial adaptive features is contained in exponent b^α. Depending on its value (i.e. whether it is smaller or bigger than 1, or even non-negative) we can reach for a particular membership function practically a constant value or a singleton. The negative exponent b^α is also possible. In the last paper Golénia et al. (2009) the membership functions are assumed as polynomial of third degree of the form

$$\mu_\alpha(x) = a_\alpha d_\alpha(x)^3 + 1 \tag{15}$$

with an appropriate constant a_α. Now we can suggest the next family of polynomial functions, namely

$$\mu_\alpha(x) = 1 - 2\left(\frac{d_\alpha(x)}{d_{\alpha 0}}\right)^2, \quad \text{for } 0 \le d_\alpha(x) \le \frac{d_{\alpha 0}}{2},$$

$$\mu_\alpha(x) = 2\left(\frac{(1-2\delta)d_\alpha(x) - (1-\delta)d_{\alpha 0}}{d_{\alpha 0}}\right)^2, \quad \text{for } \frac{d_{\alpha 0}}{2} \le d_\alpha(x) \le \frac{1-\delta}{1-2\delta}d_{\alpha 0}, \tag{16}$$

$$\mu_\alpha(x) = 0 \text{ for } \frac{1-\delta}{1-2\delta}d_{\alpha 0} \le d_\alpha(x),$$

where $\delta = \sqrt{\frac{\epsilon}{2}}$. For the center (centroid) of each cluster we assume the highest level of membership, i.e. $\mu_\alpha(a^\alpha) = 1$. However, if we move out of the center the membership level should diminish, and outside of the cluster $\mathcal{X}_\alpha$ the membership level should be zero.

It is acceptable to make the main, universal assumption[3] about the decay of the membership level, and take a small number $\epsilon << 1$, which gives the membership level of points at the boundary of each cluster. Due to our estimation (11) of the maximal distance d_α of points of each cluster, we put

$$a_\alpha d_{0\alpha}^3 + 1 = \epsilon.$$

From here we derive for the family of membership functions (15) the expression $a_\alpha = (\epsilon - 1)/d_{0\alpha}^3$. Similar assumption is made for the quadratic case (16). Membership functions from the last family (16) have the value ϵ at $d_{\alpha 0}$ and $1/2$ at $d_{\alpha 0}/2$. Notice, that all parameters of membership polynomial functions (15,16) are determined in terms of one parameter ϵ and the characteristic features of cluster contained in the scatter matrix S_α and $d_{\alpha 0}$.

Fuzzy rules

Constructing the fuzzy inference system for our problem we consider a family $\{\mathcal{R}_m : m = 1, 2, ...,Q\}$ of three-conditional rules of the form

$$\textbf{if } x \text{ is } A_{1h} \textbf{ and } x \text{ is } B_{2k} \textbf{ and } x \text{ is } C_{2l} \text{ then } y \text{ is } C_{hkl}(x), \tag{17}$$

with $Q = M_1 \cdot M_2 \cdot M_3$, since all possible triple (h, k, l) are admitted, in which the consequent part C_{hkl} is not a fuzzy set but a weighted combination of three functions f_{ah} with $a = 1, 2, 3$, namely

$$C_{hkl}(x, \Omega, \epsilon) = \{\mu_{1h}(x)f_{1h}(x, \Omega_{1h}) + \mu_{2k}(x)f_{2k}(x, \Omega_{2k}) + \mu_{3l}(x)f_{3l}(x, \Omega_{3l})\}\gamma_{hkl}(x), \tag{18}$$

where

$$\gamma_{hkl}(x) = \{\mu_{1h}(x) + \mu_{2k}(x) + \mu_{3l}(x)\}^{-1}.$$

We can see that the consequence of each rule is a weighted (convex) combination of individual outputs of the neural networks. This type of generalized Takagi-Sugeno-Kang's fuzzy rule will appear in the final construction stage.

---

[3] In order to diminish the number of parameters to be tuned later on.

Aggregation

We define the aggregation of all $Q = M_1 \cdot M_2 \cdot M_3$ rules. Now we calculate the activation level of each three-conditional fuzzy rule using the multiplication method (cf. Weigl & Kosiński (1996); Kosiński et al. (1997)). Activation level, denoted by v_{hkl} of the typical rule (17), will be

$$v_{hkl}(\, x) = \mu_{1h}(x) \cdot \mu_{2k}(x) \cdot \mu_{3l}(x). \tag{19}$$

Then we normalized all activation levels in such way, that their sum up to 1, i.e. the normalized level of activation of the rule (17) will be

$$\bar{v}_{hkl}(x) = \frac{v_{hkl}(x)}{S(x)} \text{ , where } S(\, x) = \sum_{h'=1}^{M_1}\sum_{k'=1}^{M_2}\sum_{l'=1}^{M_3} v_{h'k'l'}(x). \tag{20}$$

Hence the overall output of the FUZZy-Neural Inference System (FUZZNIS) will be

$$z = f(x,\Omega,\epsilon) = \sum_{h=1}^{M_1}\sum_{k=1}^{M_2}\sum_{l=1}^{M_3}\bar{v}_{hkl}(x)C_{hkl}(x,\Omega_{1h},\Omega_{2k},\Omega_{3l},\epsilon). \tag{21}$$

Here Ω is a collection of all individual vectors $\Omega_{1h},\Omega_{2k},\Omega_{3l}$. When the generalized Gaussian functions appear additional to Ω the extra weight vector Θ appears, which is a collections of all parameters of Gaussian membership functions (14), namely (d^α, b^α) with α as a multi-indices $1h$, $2k$, $3l$. In the case of both polynomial functions (15, 16) only one parameter ϵ needs adaptation.

It is worthwhile to mention that when more different fuzzy domain coverings are constructed, one can assume multi-conditional rules in the module Kosiński et al. (1997); Kowalczyk (1999).

Final adaptation

Constructed in the last sections the information system presented in the form of (21) needs the last stage of adaptation of the parameter ϵ influencing all polynomial membership functions or parameters (d^α, b^α) in the case of Gaussian functions, and then the convex combination (21). To end this we have to define a new error function

$$E(\epsilon,\text{TRE}) := \frac{1}{Q}\sum_{x,y \in \text{TRE}} |f(x,\Omega,\epsilon) - y_{dis}|^2, \tag{22}$$

where y_{dis} is desired output value (i.e. decision) to the input value x. Now the terminal stage of the construction follows in which the error function (22) will be minimized over all points (x,y) taken from TRE. The gradient descent method or a genetic algorithm can be implemented for this purpose, since the error function is non-quadratic in the variables ϵ. As initial values some small, comparing to 1 positive value for ϵ could be taken, e.g. 10^{-2}.

The presented algorithm has been implemented in C++ and is in a testing stage for 4-D input data. We have adapted our system using 216 elements of training data in TRE and other 125 elements as testing data TES. The training and testing pairs were chosen randomly from the graph of the real-valued function of 3 variables

$$y = F(x_1, x_2, x_3) = (1 + x_1^{0.5} + x_2^{-1} + x_3^{-1.5})^2 \tag{23}$$

where x_1, x_2, x_3 were randomly taken from the interval [1,6]. Output values y were in the interval [5.101, 22.049].

First we make a comparison of computational results when the membership functions (14,15,16) are taken, on a distributed memory architecture of a supercomputer with 8 processors at the University of Bristol. Each covering was discovered (formed) in the grouping phase which evolved during 2000 iterations. Each neural network f_α was adapted over 10000 iterations with a test set corresponding to half of the size of $\mathcal{K}_\alpha$. The test set for f_α was formed with the maximum of its associated membership function on TES set.

Level	APE			RMSE		
	Min	Avg	Max	Min	Avg	Max
Low-level: Neural networks						
Neural network	2.7046	4.0375	5.8498	0.3801	0.8084	2.2332
	2.5328	25.6537	67.4742	0.2680	3.0134	6.3280
Medium-level: Fuzzy rules						
Polynomial	2.7046	14.6289	26.1748	0.3801	1.9577	3.8075
	12.9252	23.3371	38.4051	1.9672	3.2252	4.5419
Quadratic	2.7046	9.5942	15.1276	0.3801	1.4735	2.7539
	6.1638	18.9901	38.4051	0.9787	2.7037	4.5419
Gaussian	2.7046	17.3242	40.7523	0.3801	2.2424	5.0633
	13.4063	25.8251	38.4051	2.5151	3.5423	4.5419
Top-level: Overall output						
Polynomial		17.6714			2.56067	
		18.5190			2.67368	
Quadratic		11.0716			1.81720	
		12.4913			1.90090	
Gaussian		22.8443			3.18962	
		23.3547			3.31610	

Table 2. Error analysis of FUZZNIS

The absolute error function (AE) was defined as follows, on each cluster $\mathcal{X}_\alpha$ on TRE:

$$AbsoluteError(X_\alpha, \text{TRE}) = \frac{1}{size(\text{TRE})} \sum_{(x,y) \in \text{TRE}} | f_\alpha(x) - y_{des} |, \tag{24}$$

and similarly on TES.

In Table 2, we observe the errors inside the FUZZNIS for different levels of granularity: Low level (the first stage of construction of the system of FNN's): neural networks, Medium level

(the second stage of construction of the system): fuzzy rules, and Top level (the final stage of construction - aggregation): overall output. That has been done with different membership functions (Polynomial of 3ed degree, Quadratic and Gaussian). The errors are given in Absolute Percent Error (APE) and Root Mean Square Error (RMSE) with eq.(25) and eq.(26) for each level using a local TRE and TES. In Table for each level and each membership function, the errors in the first row corresponds to the set TRE whereas in the second row to TST. In both equations below the word TREST refers to either the set TRE or to TES, while $net(x)$ refers to the real output of the network, and y_{des} denotes the desired output. We have used $f_\alpha(x, \Omega_\alpha)$ from(13) for the low-level, $C_\alpha(x)$ from(18) for the medium-level and $f(x, \Omega, \epsilon)$ from(21) for the top-level.

In the low-level, for each neural network, the set TRE was taken as a common part of the whole set TRE and the corresponding cluster. For this level, the accuracy was relatively good for a small TRE with 4% (± 1.8%). However, a high value of error was noticed for TES. In the medium-level, the fuzzy rules were evaluated by pair of clusters. In the top-level, the overall output of the FUZZNIS, the TRE and TES were taken completely.

Independently of the membership functions the following trend was remarked:

For the medium-level, we established for the TRE that the error is increasing whilst decreasing for the TES in average compared to the low level.

For the overall output at the top-level, the error was higher for the TRE and lower for the TES than for the medium-level up to the point to be regularly closed.

Among the membership functions the results clearly showed that the Quadratic function was the best membership function, followed by the Polynomial and finally by the Gaussian.

$$\text{APE}(x, y, \text{TREST}) = \frac{1}{size(\text{TREST})} \sum_{k=1}^{size(\text{TREST})} \left| \frac{net(x^k) - y_{des}^k}{net(x^k)} \right| \cdot 100\% \qquad (25)$$

$$\text{RMSE}(x, y, \text{TREST}) = \sqrt{\frac{1}{size(\text{TREST})} \sum_{k=1}^{size(\text{TREST})} (net(x^k) - y_{des}^k)^2} \qquad (26)$$

In Table 2 for each case of membership functions the double lines contain errors on TRE in the upper line, and similarly on TES in the lower line.

In Table 3 we show for each case of membership functions the minimal absolute error on individual clusters on TRE and TES, respectively.

Membership function	AE on TRE	AE on TES
Polynomial 3ed degree	0.153063	0.200833
Quadratic	0.070613	0.0765449
Gaussian	0.222665	0.235378

Table 3. Absolute error (AE) on individual clusters for the membership functions

We can see that our quadratic membership function makes pretty well. The results for the Gaussian function obtained with new overall output function are better than in the previous case Golénia et al. (2009) when different overall output function appeared.

Moreover, we found out that the utilization of the supercomputer is really useful for working with an information decision systems in comparison to several hours in the previous case Kosiński & Golénia (2008).

It can be mentioned that our membership functions (14,15,16) of fuzzy sets can be generalized to include ordered fuzzy numbers, recently invented by the first author W.K. and his coworkers, cf. Kosiński (2006); Kosiński et al. (2003).

6. Conclusions

In the goal of understanding the opinions written in natural language, an Opinion Mining knowledge was necessary to implement. For this reason, we presented in this chapter new approaches to automatically detect opinion from the text. The two classifications (group conduct and linguistic) have been proposed by us. Then, we have compared our approaches with the approach generally used in this field (the statistical classification, which is based on Naive Bayes classifiers). After carrying out tests, we can observe that we have succeeded to implement a first innovative method based on a linguistic classifier. The results obtained after this classification give us satisfaction. We can, therefore, conclude that the linguistic analysis, which is deeper, is an important research path in the field of Sentiment Analysis.

The final classifier can be constructed as a FUZZy-Neural Inference System (FUZZNIS) by copying the method known in approximation of multivariant functions. The designing procedure of FUZZNIS has been presented in Section 5.2. The results concerning its application to an approximation of a benchmark function of 3 variables (23) allow us to say that by applying FUZZNIS as the final classifier an optimal decision support system can be obtained.

7. Acknowledgement

8. References

Anderberg, M. (1973). Cluster analysis for applications, Probability and Mathematical Statistics. Mathematical Statistics. New York.

Cover, T.M., & Thomas, J.A. 1991. Elements of Information Theory. John Wiley.

Cybenko, G. (1989). Approximation by superpositions of sigmoidal function., *Mathematics of Control, Signals, and Systems* 2: 303–314.

Das, S. & Chen, M. (2001). Yahoo! for amazon: Extracting market sentiment from stock message boards., *the 8th Asia Pacific Finance Association Annual Conference (APFA)*.

Dave, K., Lawrence, S., & M., Pennock D. 2003. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. Pages 519–528 of: Proceedings of WWW.

Duch, W. & Jankowski, N. (1997). New neural transfer functions, *Int. J. Appl. Math. Comput. Sci.* 7: 639–658.

Duran, B. S. & Odell, P. (1974). Cluster analysis – a survey, Lecture Notes in Economics and Mathematical Systems, Vol. 100, Springer-Verlag, Berlin.

Dziczkowski, G. & Wegrzyn-Wolska, K. (2007). Rcss - rating critics support system purpose built for movies recommendation., *Advances in Intelligent Web Mastering.*

Dziczkowski, G. & Wegrzyn-Wolska, K. (2008a). An autonomous system designed for automatic detection and rating of film. extraction and linguistic analysis of sentiments., *In Proceedings of IEEE/WIC/ACM inter. conference on Web intelligence and intelligent agent technology* WI: 847–850.

Dziczkowski, G. & Wegrzyn-Wolska, K. (2008b). Tool of the intelligence economic: Recognition function of reviews critics., *In ICSOFT 2008 Proceedings.*

Gołąbek, B., Kosiński, W. & Weigl, M. (1999). Adaptation of learning rate via adaptation of weight vector in modified m-delta networks., *Computational Intelligence and Appli cations* P.S. Szczepaniak (Ed.), Studies in Fuzziness and Soft Computing, Vol. 23, Physica-Verlag, c/o Springer-Verlag,: 156–163.

Golénia, B., Kosiński, W. & Weigl, M. (2009). *Recent Advances in Intelligent Information Systems*, EXIT, Warsaw, ISBN 978-83-60434-59-8, chapter An Evolutionary Method for Designing Decision Support, pp. 275–288.

Hatzivassiloglou, V., & McKeown, K. 1997. Predicting the semantic orientation of adjectives., *In Proc. of the Joint ACL/EACL Conference.*

Hatzivassiloglou, V. & Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity., *In Proceedings of the International Conference on Computational Linguistics (COLING).*

Hornik, K. (1991). Approximation capabilities of multilayer feedforward, *Neural Networks4 251–257.*

Jang, J.-S. (1993). Anfis: adaptive-network-based fuzzy inference system, *Systems, Man and Cybernetics* 23: 665–685.

Jang, J. & Sun, T. C. (1993). Functional equivalence between radial basis function networks and fuzzy inference system., *IEEE Transaction on Neural Networks 156–159.*

Joachims, T. & Sebastiani, F. (2002). Guest editors introduction to the special issue on automated text categorization., *Intell. Inform. Syst.*

Koleśnik, R., Koleśnik, L. & Kosiński, W. (1999). Genetic operators for clustering analysis, *Intelligent Information Systems VIII* Proc. of theWorkshop held in Ustroń, Poland: 203– 208.

Kosiński, W. (2006). On fuzzy number calculus., *Int. J. Appl. Math. Comput. Sci.* 16: 51 57.

Kosiński, W. & Golénia, B. (2008). Double covering of data for multivariate approximation, *Development in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Sets and Related Topics. Foundations* K. Atanasov et al. (Eds.), EXIT, Warsaw: 97–111.

Kosiński, W. & Kowalczyk, D. (2007). Adaptive decision support system using web-users profile data,, *Advances in Intelligent Web Mastering* K. M. Węgrzyn-Wolska and P.S.

Szczepaniak (Eds.), Proc. of 5th AWIV'2007, Fontainebleau, France, ASC, vol. 43, Springer-Verlag, Berlin Heidelberg: 180–185.

Kosiński, W., Kowalczyk, D. & Weigl, M. (2007). Multivariate data approximation with preprocessing of data, *Computer Assisted Mechanics and Engineering Sciences* 14: 651–656.

Kosiński, W., Prokopowicz, P. & Ślęzak, D. (2003). Ordered fuzzy numbers, *Bulletin of the Polish Academy of Sciences, Sér Sci Math* 51: 327–338.

Kosiński, W. &Weigl, M. (1998). General mapping approximation problems solving by neural networks and fuzzy inference systems., *Systems Analysis Modelling Simulation* 30: 11– 28.

Kosiński, W. & Weigl, M. (2000). Adaptive information systems for data approximation problems., *Fuzzy Control Theory and Practice* Hampel R., Wagenknecht M., Chaker N.(Eds); Physica-Verlag, Springer, Heidelberg New York,: 109–120.

Kosiński, W., Weigl, M. & Kieś, P. (1997). Fuzzy domain covering of an inference system, *Proceedings of the Third National Conference on Neural Networks and Their Applications* Kule n. Częstochowa, PNNS, Częstochowa: 310–321.

Kosiński, W., Weigl, M. & Michalewicz, Z. (1998). Evolutionary domain covering of an inference system for function approximation., *Evolutionary Programming VII; V.M. Port, N. Saravanam, D. Waagen and A. E. Eiben (Eds.)* Proceedings of the 7th International Conference, EP'98, San Diego California, NCS, vol. 1447, New York: 167–180.

Kosiński, W., Weigl, M., Michalewicz., Z. & Koleśnik, W. (1998). Genetic algorithms for preprocessing of data for universal approximators., *Intelligent Information Systems VII* Proceedings of the Workshop held in Malbork, Poland, Instytut Podstaw Informatyki, PAN,Warszawa: 320–331.

Kowalczyk, D. (1999). *An adaptive fuzzy inference system as a fuzzy controller. rozmyty adaptacyjny system wnioskujący jako sterownik rozmyty*, Master's thesis, WSP w Bydgoszczy, Wydział Matematyki, Techniki i Nauk Przyrodniczych, Specjalność: Technika Komputerowa, Bydgoszcz.

Lewis, D. & Haues, P. J. (1994). Guest editorial for the special issue on text categorization., *ACM Trans. Inform. Syst.*

Pang, B. & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts., *In Proceedings of the Association for Computational Linguistics (ACL).*

Pang, B., Lee, L. & Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques., *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP).*

Turney, P. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews., *In Proceedings of the Association for Computational Linguistics (ACL).*

Weigl, M. & Kosiński, W. (1996). Approximation of multivariate functions by generalized adaptive fuzzy inference network., *Proceedings of the 9-th International Symposium on Methodologies for Intelligent Systems* ISMIS'96, Zakopane, IPI PAN, Warszawa,: 120–133.

Whitelaw, C., Garg, N & Argamon, S. (2005). Using appraisal groups for sentiment analysis.,
         *In Proceedings of the ACM SIGIR Conference on Information and Knowledge Management
         (CIKM).*

Decision Support Systems Advances in

Edited by Ger Devlin

This book by In-Tech publishing helps the reader understand the power of informed decision making by covering a broad range of DSS (Decision Support Systems) applications in the fields of medical, environmental, transport and business. The expertise of the chapter writers spans an equally extensive spectrum of researchers from around the globe including universities in Canada, Mexico, Brazil and the United States, to institutes and universities in Italy, Germany, Poland, France, United Kingdom, Romania, Turkey and Ireland to as far east as Malaysia and Singapore and as far north as Finland. Decision Support Systems are not a new technology but they have evolved and developed with the ever demanding necessity to analyse a large number of options for decision makers (DM) for specific situations, where there is an increasing level of uncertainty about the problem at hand and where there is a high impact relative to the correct decisions to be made. DSS's offer decision makers a more stable solution to solving the semi-structured and unstructured problem. This is exactly what the reader will see in this book.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Witold Kosinski, Grzegorz Dziczkowski, Bruno Golenia and Katarzyna Wegrzyn-Wolska (2010). Towards an Optimal Decision Support System, Decision Support Systems Advances in, Ger Devlin (Ed.), ISBN: 978-953-307-069-8, InTech, Available from: http://www.intechopen.com/books/decision-support-systems-advances-in/towards-an-optimal-decision-support-system

INTECH
open science | open minds