ZASTOSOWANIA INFORMATYKI I ANALIZY SYSTEMOWEJ W ZARZĄDZANIU

Polska Akademia Nauk • Instytut Badań Systemowych

Seria: BADANIA SYSTEMOWE
tom 33

Redaktor naukowy:

Prof. dr hab. Jakub Gutenbaum

Warszawa 2003

ZASTOSOWANIA INFORMATYKI I ANALIZY SYSTEMOWEJ W ZARZĄDZANIU

pod redakcją
Jana Studzińskiego, Ludosława Drelichowskiego
i Olgierda Hryniewicza

Książka wydana dzięki dotacji KOMITETU BADAŃ NAUKOWYCH

Książka zawiera wybór artykułów poświęconych omówieniu aktualnego stanu badań w kraju w zakresie rozwoju modeli, technik i systemów zarządzania oraz ich zastosowań w różnych dziedzinach gospodarki narodowej. Wyodrębnioną grupę stanowią artykuły omawiające aplikacyjne wyniki projektów badawczych i celowych KBN.

Recenzenci artykułów:

Prof. dr hab. inż. Olgierd Hryniewicz
Prof. dr hab. inż. Janusz Kacprzyk
Dr inż. Edward Michalewski
Prof. dr hab. inż. Andrzej Straszak
Dr inż. Jan Studzinski
Dr inż. Sławomir Zadrożny

Komputerowa edycja tekstu: Anna Gostyńska

Wydawca: Instytut Badań Systemowych PAN
ul. Newelska 6, 01-447 Warszawa

Dział Informacji Naukowej i Wydawnictw IBS PAN
Tel. 836-68-22

Druk: Zakład Poligraficzny Urzędu Statystycznego w Bydgoszczy
Nakład 200 egz. ark. wyd. 25,2 ark. druk. 20,0

ISNB 83-85847-83-9
ISSN 0208-8028

Rozdział 4

Metody analizy systemowej w zarządzaniu

TEXT CATEGORIZATION USING SOME ELEMENTS OF FUZZY LOGIC

Janusz Kacprzyk and Sławomir Zadrożny
Systems Research Institute, Polish Academy of Sciences
<kacprzyk@ibspan.waw.pl, zadrozny@ibspan.waw.pl >

A linguistic quantifier guided aggregation for an automatic text document categorization is considered. Text categorization is a specific problem of information retrieval (IR) attracting recently a lot of attention and research. The applicability of fuzzy logic in IR has been already advocated and shown. However, the specific task of text categorization has not yet been directly addressed. Our approach is based on the calculus of linguistically quantified propositions proposed by Zadeh. Some computational results are presented.

Keywords: information retrieval, linguistic quantifiers, text categorization.

1. Introduction

Modern management is heavily based and relying on information which can be obtained from all possible sources. More and more, in addition to traditional sources of information that contain numeric data, information is derived from "non-conventional" sources which contain other forms of data, notably pieces of text, graphics, etc. Text type sources are considered here. As it is the case in all kinds of data, also text type data are pervaded with uncertainty and imprecision so that data (information) is imprefect. Fuzzy logic in a broad sense provides means for the processing of imperfect information. This includes the aggregation, fusion, etc. of such information that may be done using flexible operators, exemplified by *linguistic quantifiers*. Information retrieval tasks are perfectly amenable to such a treatment. The information retrieval process is characterized by such features as uncertainty (most often of the probabilistic nature), partial matching, incompleteness of queries, a vague concept of the *relevance*, etc. A lot of research has been done in the area of application of the elements of fuzzy logic for the purposes of information retrieval, cf., e.g., (Bordogna et al., 1995; Bordogna and Pasi, 2000; Bordogna et al., 2000; Kraft et al., 1999). Main issues of text documents representation and their querying have been addressed within this framework. The use of fuzzy logic related concept for query structure and interpretation is particularly promising. This is due to the fact that some elements of the classical IR system interface may be artificially precise and too rigid for a human user. Besides the main task of an IR system, i.e., retrieval of documents relevant for the user needs, there are many other related tasks. Among

them, quite an important task is that of *text document categorization*. Basically, it consists in assigning some thematic categories to documents. This may be, and often is, done manually. However, in the case of huge document sets, as the Internet, it becomes inefficient and ineffective. Thus, automatic approaches are more and more important.

Text categorization task exhibits some imprecision. Even a human being may be unsure as to a clear cut classification of a document to just one category. Moreover, it is quite natural to consider a degree of belongingness to a category. This becomes even more apparent in case of an automatic classification procedure. We may easily expect that the results of classification may be ambiguous. The fuzzy logic approach has proved to be useful in such a context. We have implemented (Zadrożny et al., 2002) a pilot version of an Internet oriented IR system featuring some elements of fuzzy logic built into the user interface and making it more human consistent. Here, we investigate some possible application of fuzzy logic related concepts to the very classification process. Our approach is mainly based on the use of linguistically quantified propositions in the sense of Zadeh to model some intuitively compelling rules of classification.

We start with a brief introduction to Zadeh's calculus of linguistically quantified propositions. Then, fundamentals of IR are discussed with an emphasis on the task of documents categorization. In the next section we present our proposed text document classifier employing Zadeh's calculus of linguistically quantified propositions. We conclude with some results of computational experiments.

2. A Calculus of linguistically quantified propositions

The linguistically quantified propositions may be exemplified by: "*Most of the properties of two objects are matching*". In the context of comparison of two objects, such a proposition makes it possible to express in a flexible way the requirement as to the compatibility (matching) of two objects. It may be interpreted as a certain scheme of aggregation of partial compatibility conditions of these objects with respect to the particular properties characterizing them. Thus, a linguistic quantifier ("most" in the above example) replaces basic classical aggregation operators related to logical connectives (AND, OR), which may be too strict. It often happens that these all/some conditions quantified are of a gradual type, and hence both the conditions and quantifier are best modeled within the framework of fuzzy logic.

Zadeh (Zadeh, 1983) introduced two types of *linguistically quantified propositions*:

$Q\ X$'s are G's (type I) (1)

$Q\ B's$ are G's (type II) (2)

where Q is a linguistic quantifier, and G and B are fuzzy sets in the universe X. Fuzzy linguistic quantifiers are represented by fuzzy sets defined in an appropriate universe. The *proportional* linguistic quantifiers such as "most", "almost all", etc. are represented by fuzzy subsets, Q, on the interval [0,1]:

$$\mu_Q: [0, 1] \rightarrow [0, 1] \tag{3}$$

Zadeh proposed an interpretation for proportional linguistic quantifiers such that the truth degree T of proposition (1) is computed using:

$$T = \mu_Q(\text{card}(G)/\text{card}(X)) = \mu_Q\left(\sum_i \mu_G(x_i)/n\right) \tag{4}$$

where μ_Q is the membership function of quantifier Q and n is the cardinality of the universe X. For propositions of type (2) we have:

$$T = \mu_Q(\text{card}(G \cap B)/\text{card}(B)) = \mu_Q\left(\sum_i (\mu_G(x_i) \wedge \mu_B(x_i))/\sum_i \mu_B(x_i)\right) \tag{5}$$

Thus, the truth of a proposition of type (1) is proportional to the fraction of elements of the universe X that belong to its subset G. An exact form of this relationship is determined by the Q's membership function, which may be of the following, piece-wise linear, form for "most":

$$\mu_Q(y) = \begin{cases} 1 & \text{for } y \geq 0.8 \\ 2y - 0.6 & \text{for } 0.3 < y < 0.8 \\ 0 & \text{for } y \leq 0.3 \end{cases}$$

On the other hand, the truth of a proposition of type (2) is proportional to the fraction of elements of a (fuzzy) set $B \subseteq X$ belonging at the same time to $G \subseteq X$.

3. Fundamentals of information retrieval and document categorization

Let us assume the following notation for our discussion of the text categorization task:

$D = \{d_i\}_{i=1,N}$	- a set of text documents
$C = \{c_i\}_{i=1,S}$	- a set of categories
$\Xi: D \times C \rightarrow \{0, 1\}$	- assignment of categories to documents
$D_1 \subset D$	- a set of training text documents, i.e., $\Xi(d,c)$ is known for $d \in D_1$ and any $c \in C$

Often, some additional assumptions are made about the set C and/or the assignment function Ξ. Namely, often C contains just two categories (with the meaning in the spirit of „interesting/relevant" versus „uninteresting/irrelevant").

Often, it is assumed that exactly one category is assigned to a document. We adopt here the most general case of multiclass multilabel text categorization that does not assume these restrictions either on C or on Ξ. Basically, the task of automatic text document categorization consists in extending Ξ from D_1 to the whole set D. Thus, text categorization is a typical example of a classification task. More precisely, the process consists of two phases:

- learning of classification rules (explicit or implicit; building a classifier) from examples of documents with known class assignment (*supervised learning*),
- classification of documents unseen earlier using derived rules.

In order to build a classifier we have to assume some representation of the documents. We follow here the classical IR approach referred to as the bag of words (index terms) representation of the document. Let us assume additionally the following notation:

$T = \{t_j\}_{j=1,M}$ - a set of index terms

Then, as in the vector space model, the documents are usually represented via a function F:

$$F: D \times T \rightarrow [0, 1] \tag{6}$$

i.e., a document is represented as a vector:

$$d_i \rightarrow [w_1,...,w_M] \qquad w_j = F(d_i,t_j) \qquad d_i \in [0,1]^M \tag{7}$$

where each dimension corresponds to an index term and the value of w_j (*weight*) determines to what extent a term $t_j \in T$ is essential for the description of contents of the document. Most often, the index terms are just some terms that appear in the training set of documents. Then, a popular version of function F is the one based on some statistical considerations and often referred to as a *tf* × *idf* function:

$$F(d_i,t_j) = (f_{ij} / \arg\max_j f_{ij}) * (\log(N/n_j) / \arg\max_j \log(N / n_j)) \tag{8}$$

where f_{ij} is the frequency of a term t_j in a document d_i, N is the number of all documents in the set (collection) D and n_j is a number of documents from D where term t_j appears (*document frequency*). Thus, the first factor is the normalized frequency of term t_j (*tf*, *term frequency*) in document d_i, while the second factor is the normalized inverted frequency (*idf*, *inverted document frequency*) in the collection D of documents in which term t_j appears at least once. Other normalization schemes may be employed, too.

Thus, we usually start with a numerical representation of documents as formalized by (7). Then, any of numerous classifier construction algorithms may be

applied, including rule-based systems, decision trees, artificial neural networks, etc. One of classical algorithms developed in the area of IR is that of Rocchio (Rocchio, 1971; Joachims, 1997). The learning phase consists in computing a *centroid* vector for each category of documents. Then, in the classification phase, a document is classified to a category whose centroid is most similar to this document. The similarity may be meant in several ways – in the original Rocchio's approach it corresponds to the Euclidean distance.

The type of classifiers considered here produce for a document to be classified and each category a matching degree expressing the extent to which a document possibly belongs to given category. Obviously, if just one category is to be assigned to a document we choose the category with the highest matching degree (rank). However, in case of multilabel categorization the situation is not so clear and the classifier has to decide how many of those top ranked categories should be assigned to a document under consideration. This is referred to as a thresholding strategy (Yang, 2001; Sebastiani, 1999; Yang, 1999). Usually (Yang, 2001), the following strategies are considered: rank-based thresholding (RCut), proportion based assignment (PCut) and score-based local optimization (SCut). The first strategy consists in choosing r top categories for each document. Parameter r may be set by the user or automatically tuned (learned) using a part of the training set of documents. The next strategy works for "batch categorization" (i.e., where a set/batch of documents has to be classified at once) and assigns to each category such a number of documents from a batch of documents to be classified so as to preserve a proportion of the cardinalities of particular categories in the training set. The last method assigns a document to a category only if a matching score of this category and document is higher than a threshold. Thresholds are tuned using a part of the training set of documents, separately for each category. In the next section we propose other strategies using the concept of a linguistic quantifier.

4. A Rocchio style classifier employing linguistic quantifiers

The main idea of a Rocchio style classifier is to compute a centroid (profile) for each category and then to base the categorization decision for a document on its distance (more generally, some measure of similarity) from centroids of all categories. For the further discussion let us assume the following notation:

$$P = \{p_i\}_{i\in[1,S]}$$

is a set of centroids, one for each of S categories. Each centroid is represented by a vector:

$$p_i = [u_{i1},\ldots,u_{iM}] \tag{9}$$

where M denotes, as previously, the number of terms used to index the documents.

The most popular similarity measure is the cosine of both vectors:

$$\mathrm{sim}(d,p) = \sum_{i=1}^{M} w_i u_i \Big/ \sqrt{\sum_{i=1}^{M} u_i^2} \sqrt{\sum_{i=1}^{M} u_i^2} \tag{10}$$

where $d = [w_1,\ldots,w_M]$ and $p = [u_1,\ldots,u_M]$.

Our approach assumes, classically, the computation of centroids for all categories. These centroids are, however, constructed in a different way than in the typical Rocchio style classifier. Namely, weights u_{ij} are not calculated directly as the averages of the weights of all training documents belonging to a given category but according to the following formula:

$$c_{ij} = (f_{ij} * \log(S/n_j)) \,/\, \arg\max_j(\log(S/n_j) * f_{pj})) \tag{11}$$

where f_{ij} is a frequency of term t_j in all documents belonging to category c_i and n_j is the number of categories in documents of which term t_j appears (*category frequency*). By analogy to (8) it may be called a $tf \times icf$ representation where *icf* stands for an *inverted category frequency*. A document to be classified d is represented, similarly as previously, by a vector:

$$d = [w_1,\ldots,w_M] \tag{12}$$

but this time

$$w_j = (f_j * \log(S/n_j)) \,/\, \arg\max_j(\log(S/n_j) * f_j)) \tag{13}$$

where f_j is a frequency of term t_j in the document d and n_j is the category frequency of this term, cf. (11). Now, we base our decision on the classification of document d as pertinent to category c_i on its similarity to a corresponding centroid p_i. Let us observe that, in given context, the similarity of a query and a centroid intuitively means that terms representing them have comparable weights, relatively or absolutely. As the categories (their centroids) represent many documents, then one should not expect a match between a centroid and a document along all dimensions of their representation. More reasonable is to formulate the requirements that along *most* of these dimensions there is a match. This may be formalized as follows using the calculus of linguistically quantified propositions:

"A document matches a category if *most* of the *important* terms present in the document are also *present* in the centroid of the category"

The idea refers directly to our previous experiences with a database fuzzy querying (Kacprzyk and Zadrożny, 2001; Kacprzyk, Ziółkowski, 1986). The above

linguistic expression is formalized using Zadeh's calculus of linguistically quantified propositions by, cf. (2):

$Q\ B's$ are $G's$

where X, the universe considered, is a set T of all index terms, B is a fuzzy set of terms important for the document d, i.e., $\mu_B(t_j) = w_j$ cf. (13) and G is a fuzzy set of terms present in centroid p_i of category c_i, i.e., $\mu_G(t_j) = u_{ij}$. Thus

$$\text{sim}(d,p) = \text{truth}(Q\ B's \text{ are } G's) \tag{14}$$

Due to a high dimensionality of the space considered and known deficiencies of linguistic quantifiers in the sense of Zadeh, we also tested a modified version where only terms weighted in the document higher than a certain threshold or only, say, 10 top ranked terms, are considered. For the comparison, we present also some computational results for the classical cosine-based approach (10).

Concerning the thresholding strategy we propose an approach also based on fuzzy linguistic concepts. The underlying idea may be expressed as follows:

"Select such a threshold r that *most of the important categories had a number of sibling categories similar to r in the training data set*"

Thus, for each $r \in [1,R]$ we compute the truth degree of the italicized clause above (R is a parameter). This is again formalized using Zadeh's calculus of linguistically quantified propositions as:

$$Q\ B's \text{ are } G's \tag{15}$$

where X, the universe considered, is a subset of C of 10 categories with the highest matching score, B is a fuzzy set of important categories for a given document d, i.e., $\mu_B(c_i) = \text{sim}(d,c_i)$ where sim $(\cdot,\cdot)$ is the matching function (14) used and d is a document to be classified. G is a fuzzy set of categories, that, on average, had in the training set the number of sibling categories similar to r for which truth value of (15) is calculated. This similarity is modeled by a similarity relation which is another parameter of the method. For the purposes of this strategy, for each category the number of average sibling categories in the training data set is first computed. By the sibling category for a category c_i we mean a category that is assigned to the same document as the category c_i. This strategy is referred to later as T.I.

Another approach exploiting the concept of sibling categories works as follows. Only categories whose matching score is higher than a certain parameter (in our experiments usually 0.2 is assumed) are taken into account. Their scores are normalized (divided by the sum of their scores) and then the weighted sum of the average number of siblings is taken as a threshold cut (rounded to the nearest integer value). This strategy is referred to later on as T.II. For the comparison we also tested

simple RCut strategy with a threshold rank equal 2, i.e., two top scored categories are assigned to each document. This strategy is referred to later as T.III.

5. Results of a computational experiment

In our general setting for computational experiments we are following Yang and Liu's work (Yang and Liu, 1999). The text corpus used is Reuters-21578 as made available over the Internet by Lewis (Lewis). More precisely we are using the Modified Apte ("ModApte") split of the data, i.e., for the training phase a subset of news characterized by the attributes LEWISSPLIT="TRAIN" and TOPICS="YES" and for testing phase a subset LEWISSPLIT="TEST"; TOPICS="YES". In both cases, we use only news that actually contains topics and body of the text or at least the title. This gives rise to 7728 training, 3005 test documents and 114 categories. The title of the document and its body are concatenated to produce the document. The documents are preprocessed by removing stop words (Stop words list) and numbers. Stemming is done using the standard Porter's algorithm (Porter, 1980). The terms space dimensionality reduction is done using a simple approach based on document and category frequencies of terms. Namely, only terms with a document frequency higher than 3 and a category frequency lower than 75% are used. This rule yields 5565 index terms.

Table 1. Comparison of matching schemes for T.II. thresholding strategy.

Matching scheme	micro-averaging			macro-averaging			11-point average precision
	precision	recall	F1	precision	recall	F1	
Method1	0.3914	0.8215	0.5302	0.4038	0.5322	0.4592	0.8311
Method2*	0.4226	0.6765	0.5203	0.3416	0.6174	0.4398	0.7673
Cosine	0.2226	0.6462	0.3311	0.1235	0.4943	0.1976	0.6511

* only terms weighted above 0.2 are considered in matching degree computation

Table 2. Comparison of different thresholding strategies for the Method1 matching scheme.

Thresholding strategy	Micro-averaging			macro-averaging			11-point average precision
	precision	Recall	F1	precision	Recall	F1	
T.I.	0.5531	0.7765	0.6460	0.4891	0.4785	0.4837	0.8311
T.II	0.3914	0.8215	0.5302	0.4038	0.5322	0.4592	0.8311
T.III.*	0.4642	0.7478	0.5728	0.4776	0.4309	0.4530	0.8311

An evaluation of the particular approaches tested has been carried out by using standard measures of recall, precision, F1 measure and 11-point average precision. Both micro- and macro- averaging results are presented. These measures are expressed by the following formulae:

- micro-averaging

$$\text{precision} = \frac{\text{number of correct classifications made by the system}}{\text{total number of all classifications made by the system}} \tag{16}$$

$$\text{recall} = \frac{\text{number of correct classifications made by the system}}{\text{total number of all categories indicated in test documents}} \tag{17}$$

$$F1 = 2*\text{precision}*\text{recall}/(\text{precision}+\text{recall}) \tag{18}$$

- macro-averaging

First, the precision, recall and F1 measure are calculated separately for each category using formulae (16)-(18) and then the arithmetic mean of them is calculated.

In Tables 1 and 2 we present some of the results of our experiments, where method1 refers to our matching scheme defined with (14) and method2 to its variation, as explained below the table. In Table 2 we test various thresholding strategies for the linguistic guided aggregation Method1.

6. Concluding remarks

Our starting point is the fuzzy querying of crisp databases. We try to adapt some of the ideas we have proposed earlier, notably of a linguistically guided aggregation of partial matching degrees, for the purposes of text document categorization. In the paper we propose to employ a linguistic quantifier guided aggregation for the purposes of text document categorization. It is characterized by a flexibility of produced (implicit) rules of the classification that may easier take into account an inherent imprecision and vagueness of the task considered. We have presented some preliminary computational results on a testbed document corpora. The results are encouraging though an improvement is still needed. The method is highly parametric and should be better tunable than in our preliminary experiments.

The proposed approach is strongly connected with other results obtained earlier by other authors in the area of applications of fuzzy logic for IR. Specifically, the concepts of an extended fuzzy Boolean IR model are directly applicable for our approach. A further discussion of this topic will appear in our forthcoming journal paper.

In our further research we will focus on efficient tuning schemes for the proposed methods of matching a document against centroids as well as on the

thresholding strategy. Moreover, we will investigate formal properties of the proposed approach. More computational experiments will also carried out in order to test the applicability of the approach.

References

Bordogna G., Carrara P., Pasi G. (1995) Fuzzy approaches to extend Boolean information retrieval, in: P. Bosc, J. Kacprzyk (Eds.), *Fuzziness in Database Management Systems*, Physica Verlag, Heidelberg, 231-274.

Bordogna, G. Pasi, G. (2000) Application of fuzzy sets theory to extend Boolean information retrieval, in: F. Crestani, G. Pasi (Eds.), *Soft Computing in Information Retrieval*, Physica Verlag, Heidelberg New York, 21-47.

Bordogna G., Bosc P., Pasi G. (2000) Extended Boolean information retrieval in terms of fuzzy inclusion, in: O. Pons, M.A. Vila, J. Kacprzyk (Eds.), *Knowledge Management in Fuzzy Databases*, Physica Verlag, Heidelberg, 234-246.

Joachims T. (1997) A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization, in: *Proceddings of ICML-97.*

Kacprzyk J., Zadrożny S. (2001) Computing with words in intelligent database querying: standalone and Internet-based applications, *Information Sciences*, 134, 71-109.

Kacprzyk J., Ziółkowski A. (1986) Database queries with fuzzy linguistic quantifier, *IEEE Trans. on Systems, Man and Cybernetics. SMC*, 16, 474-479.

Kraft D.H., Bordogna G., Pasi G. (1999) Fuzzy set techniques in information retrieval, in: J.C. Bezdek, D. Dubois, H. Prade (Eds.) *Fuzzy Sets in Approximate Reasoning and Information Systems*, 3, The Handbook of Fuzzy Sets Series, Kluwer Academic Publishers, Norwell.

Kraft D.H., Buell D.A. (1992) Fuzzy sets and generalized Boolean retrieval systems, in: D. Dubois, H. Prade, R.R. Yager (Eds.) *Readings in Fuzzy Sets for Intelligent Systems*, Morgan Kaufmann Publishers, San Mateo.

Lewis D.D. Reuters-21578, Distribution 1.0, http://www.research.att.com/~lewis.

Porter M.F. (1980) An algorithm for suffix stripping, Program, 14 (3), 130-137.

Rocchio J. (1971) Relevance feedback in information retrieval, in The SMART Retrieval System: Experiments in Automatic Document Processing, Prentice-Hall Inc., 313-323.

Sebastiani F. (1999) A tutorial on automated text categorisation, in: A. Amandi, A. Zunino (eds.), Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence, Buenos Aires, AR, 7-35.

Stop words list, http://www.indiana.edu/ cgi-bin-local/doIsearch.pl?Stopwords.

Yager R.R. (1987) A note on weighted queries in information retrieval systems, *JASIS*, 38, 23-24.

Yang Y. (1999) An evaluation of statistical approaches to text categorization, *Journal of IR* **1**(1/2), 67-88.

Yang Y. (2001) A study on thresholding strategies for text categorization, in: Proceedings of ACM SIGIR Conference on Research and Devclopment in Information Retrieval (SIGIR'01).

Yang Y., Liu X. (1999) A re-examination of text categorization methods, in: Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99), 42-49.

Zadeh L.A. (1983) A computational approach to fuzzy quantifiers in natural languages, Computers and Maths with Appls. 9, 149-184.

Zadrożny S., Ławcewicz K., Kacprzyk J. (2002) Intelligent linguistic characterization and retrieval of textual documents: an Internet-based application, in: Proceedings of the IPMU'2002 conference, Annecy, France, 1223-1230.