Classification and clustering methods development and implementation for unstructured documents collections

Osipova Nataly

Department of Programming Technology,
Faculty of Applied Mathematics and Control Processes,
St. Petersburg State University,
Universitetsky pr. 35,
St. Petersburg, Russia
osipovanata@mail.ru

Abstract.

This paper is devoted to clustering and classification methods development and implementation. Here are discussed several clustering and classification methods based on words distributions analyzing approach. There was developed the Information Retrieval System which implements these methods. In this paper its architecture and features are described. Information Retrieval System methods were tested in classification environment. Testing methods and their results are described in this paper.

1. Introduction

The information search is a very serious modern problem. Document clustering and classification are very important methods of information search. They can help us to solve a lot of different problems.

Usually we work with the collection of unstructured documents, where there is no taxonomies. The information search problem in such collections comes from impossibility of documents accurate classification on topics, from complex interconnections between documents. One of the main problems of information search implementation in such collections is that we have to treat very large volumes of data. So, for solving this problem special information search methods are necessary.

In this paper several methods of document clustering and classification are presented. They are based on a new approach: words distributions analyzing. Every cluster is identified by a context word and the documents contained in the cluster have a high level of similarity. Document clustering methods are based on a distance measure which is based on the documents word distributions analyzing.

In contrast to such clustering methods as Information Bottleneck [3] or K-means [4] which try to provide a more compact data representation, the main goal is to identify documents which have a high level of relevance, to split semantically heterogeneous document collection into large number of semantically homogeneous clusters. Highly specialized words are called words with "narrow" contexts. So, after such words selection the mechanism of grouping together semantically related documents can be designed. This is the main idea of Contextual Document Clustering (CDC) approach and that was developed in joined project of Applied Mathematics and Control Processes Faculty, St. Petersburg State University and Northern Ireland Knowledge Engineering Laboratory (NIKEL), University of Ulster.

It is very important to stress that these words with "narrow" contexts are not described previously by any descriptions or training sets. They are selected automatically. It is a very important result because for most unstructured documents collections there are usually no predefined categories to sort them on. If there is a list of predefined topics then it is the classification but not the clustering task. For classification problem such methods are known as k-NN [5], Support Vector Machine [5] and others. In this paper methods based on a CDC are described.

2. Methods description

2.1. Definitions

In the paper while describing the information search methods the following definitions will be used:

Document – some named text (a book, paper, publication).

Terms dictionary – dictionary of specific words for the given area.

Dictionary – all words occurred in the collection.

Word context – a set of words which co-occur with the given word in documents. The number of words in a context determines its size.

Cluster – a group of documents, where all documents have a high level of similarity to each other.

Context or document probability distribution - a set of words probabilities which consist in the context or document. A word probability is defined as the ratio between word frequency in context (document) and context (document) size.

Entropy – uncertainty measure, here it is used to characterize commonness (narrowness) of the word context.

In this paper the following methods are be considered:

- document clustering method (unsupervised learning).
- document classification method using training sets (supervised learning).
- information retrieval methods:
  - ❑ keyword search methods

❑ cluster based search method
❑ similar documents search method

2.2. CDC algorithm description

The CDC algorithm fully described in paper [1]. Here the short overview is presented which is necessary in the following discussion.

Let us have the unstructured documents collection.

X - set of all documents in the collection.
Y - set of all words which occur in X.
x - arbitrary document from X, $x \in X$.
y - arbitrary word from Y, $y \in Y$.
Y_y - set of word which form y context.
Y_x - set of words from document x.
X_y - set of all documents the word y occurs in.
$df(y) = |X_y|$ - number of documents the word y occurs in.
$tf(x,y)$ - word y frequency, the number of the word y occurrences in the document x.
$p(y|x)$ - word y conditional probability in the document x.

$$p(y|x) = \frac{tf(x,y)}{\sum_{w \in Yx} tf(x,w)}$$

$p(V|x)$ - conditional probability distribution which describes document x. V is a random variable with values from Y_x
$p(Q|y)$ - conditional probability distribution which describes a context given with a word y. Q is a random variable with values from Y_y. The word y acts as a descriptor of the context.

$$p(w|y) = \frac{\sum_{x \in X_y} tf(x,w)}{\sum_{w \in Yy} \sum_{x \in Xy} tf(x,w)}$$

To find words with narrow contexts conditional probability distributions for all words in Y are calculated.

To select words with the narrow contexts the entropy for every word context is calculated. The entropy is found on the word context conditional probability distribution by formula:
$H(y) = H[p(Q|y)] = -\sum_{w \in Yy} p(w|y) \log(p(w|y))$ - word y entropy.
$H(y)$ is used as a criteria for narrow contexts selection. Entropy achieves its maximum for a uniform distribution when every $p(w|y)$ equals each other. Then the entropy equals to the $\log(|Y_y|)$.

It is known that the dictionary size of document collection is of the order $O(n^{\beta})$, where n is the total text size and $\beta < 1$. As the text size for documents where y occurs $\approx df(y) \cdot k$, where k is an average size for these documents, then $|Y_y| = O((k \cdot df(y))^{\beta})$ and $H(y) = O(\log(df(y)))$. Taking into account the dependency between $H(y)$ and $df(y)$ the set of words Y is divided into groups according to their $df(y)$. Then all words from one group have document frequencies from the same interval.

$$Y = \bigcup_{i=1}^{r} Y_i$$

$$Y_i = \{y : y \in Y, df_i \le df(y) < df_{i+1}\}, i = \overline{1,r}$$

$$df_{i+1} = \alpha \cdot df_i,$$

where

r - number of groups,

$\alpha > 1$ - some constant, which is known from experiments according to the collection size.

Define

C_i - number of words in a group i.

Then from every group i as the words with narrow contexts the number of words with the minimal entropy according to the group size C_i is selected.

$TC_i = N \frac{C_i}{\sum_{i=1}^{r} C_i}$ - number of words to take from each group to add to the set of narrow context words, where N is the number of narrow contexts. It is the method parameter and is selected experimentally according to the collection size. In some experiments with CDC algorithm N is equal to 1000. So, after selecting words with narrow contexts from each group the set of narrow context words Z which describe clusters is created.

Then every document in the collection have to be assigned with the closest narrow word context. The distance between document and word context is measured using Jensen-Shannon divergence of the word context probability distribution and document distribution.

Let $p1$ and $p2$ be two probability distributions. Then the Jensen-Shannon divergence of $p1$ and $p2$ is

$$JS_{\{\pi 1, \pi 2\}}[p1, p2] = H[\overline{p}] - \pi 1 H[p1] - \pi 2 H[p2],$$

where $\pi 1 \ge 0, \pi 2 \ge 0, \pi 1 + \pi 2 = 1, \overline{p} = \pi 1 p1 + \pi 2 p2$

JS is non-negative bounded function of $p1$ and $p2$, which is equal to zero if and only if $p1 = p2$. It is a concave function of $\pi 1$ and $\pi 2$ with unique maximum in $\{0.5, 0.5\}$.

So, the document x is assigned to the cluster described by narrow context word y if

$$y = \arg\min_{w \in Z}\{JS_{\{0.5,0.5\}}[p(W \mid w), p(V \mid x)]\}$$

2.3. Distance calculation method

For distance between two probability distributions calculation the formula which helps to simplify JS divergence computation was derived. It shows that JS divergence for two conditional probability distributions depends only on words which occur in both distributions with non-zero probabilities.

Theorem.

Let $px = p(V \mid x), py = p(Q \mid y)$ - two conditional probability distributions,
$V \in D, Q \in W$,
$p(w \mid x) = p(V = w \mid x), w \in D$
$p(w \mid y) = p(Q = w \mid y), w \in W$
$\bar{p} = \frac{1}{2}(px + py)$
then

$$JS(px, py) = \log(2) + \frac{1}{2}\sum_{w \in D \cap W}\left(p(w \mid x) * \log\left(\frac{p(w \mid x)}{2}\right) + p(w \mid y) * \log\left(\frac{p(w \mid y)}{2}\right)\right) -$$
$$- \sum_{w \in D \cap W}\frac{p(w \mid x) + p(w \mid y)}{2} * \log\left(\frac{p(w \mid x) + p(w \mid y)}{2}\right)$$

Proof:

$$JS(px, py) = H(\bar{p}) - \frac{1}{2}H(px) - \frac{1}{2}H(py)$$
$$H(px) = -\sum_{w \in D} p(w \mid x) * \log(p(w \mid x))$$
$$H(py) = -\sum_{w \in W} p(w \mid y) * \log(p(w \mid y))$$
$$H(\bar{p}) = -\sum_{w \in D \cup W}\left(\frac{p(w \mid x) + p(w \mid y)}{2}\right) * \log\left(\frac{p(w \mid x) + p(w \mid y)}{2}\right)$$

$$H(\bar{p}) = -\left(\sum_{w \in D}\frac{p(w \mid x)}{2} * \log(\frac{p(w \mid x)}{2}) - \sum_{w \in D \cap W}\frac{p(w \mid x)}{2} * \log(\frac{p(w \mid x)}{2}) + \right.$$
$$\left. + \sum_{w \in W}\frac{p(w \mid y)}{2} * \log(\frac{p(w \mid y)}{2}) - \sum_{w \in D \cap W}\frac{p(w \mid y)}{2} * \log(\frac{p(w \mid y)}{2}) + H^1_{D \cap W}\right)$$

where
$$H^1_{D \cap W} = \sum_{w \in D \cap W}\left(\frac{p(w \mid x) + p(w \mid y)}{2}\right) * \log\left(\frac{p(w \mid x) + p(w \mid y)}{2}\right)$$

$$\begin{aligned}H(\overline{p}) = &-\left(\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) - \frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(2)\right)+\\ &-\left(\frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y)) - \frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(2)\right)+\\ &-\left(\ \text{-}\sum_{w\in D\cap W}\frac{p(w\mid x)}{2}*\log(\frac{p(w\mid x)}{2}) - \sum_{w\in D\cap W}\frac{p(w\mid y)}{2}*\log(\frac{p(w\mid y)}{2}) + H^{1}_{D\cap W}\right)\end{aligned}$$

$$\begin{aligned}H(\overline{p}) = &-\left(\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) - \frac{1}{2}\log(2)*\sum_{w\in D} p(w\mid x)\right)+\\ &-\left(\frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y)) - \frac{1}{2}\log(2)*\sum_{w\in W} p(w\mid y)\right)+\\ &-\left(\ \text{-}\frac{1}{2}\sum_{w\in D\cap W}\left(p(w\mid x)*\log(\frac{p(w\mid x)}{2}) + p(w\mid y)*\log(\frac{p(w\mid y)}{2})\right) + H^{1}_{D\cap W}\right)\end{aligned}$$

$$H^{2}_{D\cap W} = \sum_{w\in D\cap W}\left(p(w\mid x)*\log(\frac{p(w\mid x)}{2}) + p(w\mid y)*\log(\frac{p(w\mid y)}{2})\right)$$

As $\sum_{w\in D} p(w\mid x) = 1, \sum_{w\in W} p(w\mid y) = 1$, then

$$\begin{aligned}H(\overline{p}) = &-\left(\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) - \frac{1}{2}\log(2)\right)+\\ &-\left(\frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y)) - \frac{1}{2}\log(2)\right) - \left(\ \text{-}\frac{1}{2}H^{2}_{D\cap W} + H^{1}_{D\cap W}\right)\end{aligned}$$

$$H(\overline{p}) = \log(2) - \left(\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) + \frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y))\right) - \left(\ \text{-}\frac{1}{2}H^{2}_{D\cap W} + H^{1}_{D\cap W}\right)$$

$$\begin{aligned}JS(px, py) = &\log(2) - \left(\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) + \frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y))\right) - \left(\ \text{-}\frac{1}{2}H^{2}_{D\cap W} + H^{1}_{D\cap W}\right)+\\ &-\left(-\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x))\right) - \left(-\frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y))\right)\end{aligned}$$

$$\begin{aligned}JS(px, py) = &\log(2) - \frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) - \frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y)) + \frac{1}{2}H^{2}_{D\cap W} - H^{1}_{D\cap W} +\\ &+\frac{1}{2}\sum_{w\in D} p(w\mid x)*\log(p(w\mid x)) + \frac{1}{2}\sum_{w\in W} p(w\mid y)*\log(p(w\mid y))\end{aligned}$$

$$JS(px, py) = \log(2) + \frac{1}{2}H^{2}_{D\cap W} - H^{1}_{D\cap W}$$

$$JS(x,y)=\log(2)+\frac{1}{2}\sum_{w\in D\cap W}\left(p(w\,|\,x)*\log(\frac{p(w\,|\,x)}{2})+p(w\,|\,y)*\log(\frac{p(w\,|\,y)}{2})\right)+$$
$$-\sum_{w\in D\cap W}\left(\frac{p(w\,|\,x)+p(w\,|\,y)}{2}\right)*\log\left(\frac{p(w\,|\,x)+p(w\,|\,y)}{2}\right)$$

theorem proofed.

2.4. The collection dictionary construction.

While treating large documents collections one of the most serious problems is that very large volumes of data have to be treat. Having nearly 60,000 documents and 50,000 different words in the collection the contexts for words may consist of 15,000 words. Storing and retrieving such volumes of information is quite difficult. Calculating probability distributions and entropies for words having such sizes of contexts may take a lot of time. Also, calculating entropy it is very important first of all to take into account words with narrow contexts, because they influence on the word entropy most of all. And the words with wide contexts, common words do not affect the word entropy very much. So, these words removal can only help us and does not greatly influence on the results.

Working with the specific documents collections there is the possibility to use the collection dictionary of special collection terms. For example, medical collections have such dictionaries. The dictionary have to be build beforehand. But when working with the collections of documents devoted to different topics such dictionary may not exists. Then it have to be build automatically. There are several methods of dictionary building. First of all the list of all words in the collection is created, which does not contain words from special stop-words list of common words such as pronouns. For every word its frequency is calculated ($tf(y)$). Also it is counted in how many documents the word occurs ($df(y)$). On the base of these parameters the dictionary is build. There are three options:

1). Words which have very high or very low frequency are deleted from the main list.
2). Words with very high or very low $df(y)$ are deleted from the main dictionary.
3). Method consists of both 1) and 2) methods.

Thresholds on $df(y)$ and $tf(y)$ are set according to experimental data.

2.5. Retrieval algorithms

After clusters are build they may be used as basis for implementing different information retrieval methods. In this paper the following information retrieval methods are described:

- keyword search method
- cluster based search method
- similar documents search method

Keyword search method result is a list of documents which contain words from the request.

Cluster based search method result may contain documents which does not contain words from the request. It returns documents from the cluster which are semantically related to the request.

For implementing this method for every cluster the set of words related to the cluster is build. It is build taking words which consist in the cluster context. Then if the words from the request occur in this set, then the cluster is returned as the search result. These clusters can be ranked according query word weights in the cluster context.

For implementing search of documents which are related to the given one distances between documents in every cluster using Jensen-Shannon divergence are calculated. So the full graph for every cluster is generated. Then using Greedy algorithm the Minimal Spanning Tree (MSP) is build. It has the property, that the average distance between its neighbor elements is minimum. The Greedy algorithm is the most optimal and fast algorithm for MSP generation. So, when it is necessary to find documents related to the given one the neighbor documents of the selected one are returned.

2.6. Document classification using training sets.

Sometimes the list of topics on which it is need to classify the documents is already given. Then it is the classification problem. The topics are characterized by a training set. The training set is a number of documents for which it is known which topic they are related to.

The following methods of document classification were developed:

1). First method (CM1)

- clusters of test documents are build
- topics contexts are formed using the documents from training set
- the distances between topics and clusters contexts are calculated
- every cluster is assigned to the most related topic
- all cluster documents are assigned to the topic, the cluster is assigned to

2). Second method (CM2)

- documents from the training set are mixed with all documents in the test collection
- clusters for all documents set are build
- every cluster is assigned to a topic. If the cluster contains single document from training set which is assigned to a topic, then the cluster is assigned to this topic. If there are several documents in a cluster from training set which are assigned to different topics, then the cluster is assigned to the most popular topic.

3. Information retrieval system

3.1. System architecture

All previously described methods were implemented in Information Retrieval System (IRS).

Structurally IRS consists of the following parts:

- data base server
- client

Documents and other information used in clustering and classification methods is stored in the data base (DB). DB is managed by Data Base Management System (DBMS). DBMS organizes documents store, algorithms implementation, users' access to the DB. In IRS Microsoft SQL Server 2000 is used which provide:

- very fast data processing (retrieve, insert, indexing).
- storing complex data structures
- system scalability
- access division to data
- data integrity by using transactions

Microsoft SQL Server 2000 DBMS is a high-performance, scalable and secure system. It can maintain and process a very large amount of data. High performance is due to the use of programming language T/SQL. All algorithms are implemented as stored procedures which run inside the kernel of DBMS in the most efficient order and provide fastest execution.

Client implements the user's interface. For IRS the "thick" client was implemented. In the "thick" client variant a special program written on C# is installed on a user's computer. The program forms the query according to data which user inserted and transmits it to the DB server. After getting the result it represents it in the most efficient manner. The "thick" client is much better to use for local area networks (LAN).

3.2. IRS features

In the IRS the following problems are solved:

1. documents clustering
2. keyword search method
3. cluster based search method
4. similar documents search method
5. documents classification with the use of training sets

The data base of the IRS consists of the following tables:

- documents
- dictionary of all words
- dictionary
- table of relations between documents and words: document-word
- words contexts
- words with narrow contexts
- clusters
- topics
- table of relations between topics and words: topic-word
- intermediate tables for main tables build and for retrieve implementation

1. Document clustering

For documents clustering the table documents, dictionary of all words and document-word tables are build by the application written on C#.

Using this initial data for every word its frequency and its document frequency are calculated. Based on this information the collection dictionary is created.

The special intermediate document-word table which contains only words from the dictionary is created. It has the same structure as the main document-word table. Later only the words from dictionary are taken into account.

Using this intermediate table, documents and dictionary tables with the stored procedures the table of contexts is build. It contains contexts of all words from the collection dictionary.

Using contexts table the entropy for every word is calculated.

The table of groups is created and every word is assigned to a group. The group sizes are calculated. From every group the necessary number of words with the least entropy are taken. They form table of words with narrow contexts.

The distances between all documents and probability distributions of found words with narrow contexts are calculated. Every document is assigned to the nearest cluster. The clusters table is build.

2. Keyword search method

The keyword search method is implemented using document-word table. If the document contains words from the request, then it is put in the result document list.

3. Cluster based search method

For the cluster based search the centroid table is build. It is the table which for every cluster contains the set of words which occur in the documents of this cluster. This words set may be different from the word which describes cluster context, because it may happen that there are documents in the cluster which do not contain the word, which define the context. So, the words which occur in the documents of the cluster are used in cluster based search. If words from the request are presented in such group of words for a context, then the documents of the corresponding cluster are returned as the search result.

4. Similar documents search method

To retrieve documents which are related to the given one the next steps are made:

- distances between all documents in every cluster are calculated
- distances table is build, the MST for every cluster is build and tree table is created
- for the user's request, neighbor documents using tree table for the given document are found and reply is returned to the user according to distances between documents

5. Document classification using training sets

When solving the problem of document classification on given topics first of all the table of topics is created. If there is a training set, then the contexts table for every topic is build. In this paper two document classification methods are described:

1). Clusters for the documents from test collection are build. Distances between clusters and topics probability distributions are calculated. Every cluster is assigned to the nearest topic. All documents from the cluster are assigned to the same topic.

2). All documents from the training set are put in the documents and document-word tables which contain documents from the main (test) collection. The clusters are build for all documents. Every cluster is assigned to some topic if there are documents in it from the training set which are assigned to the topic.

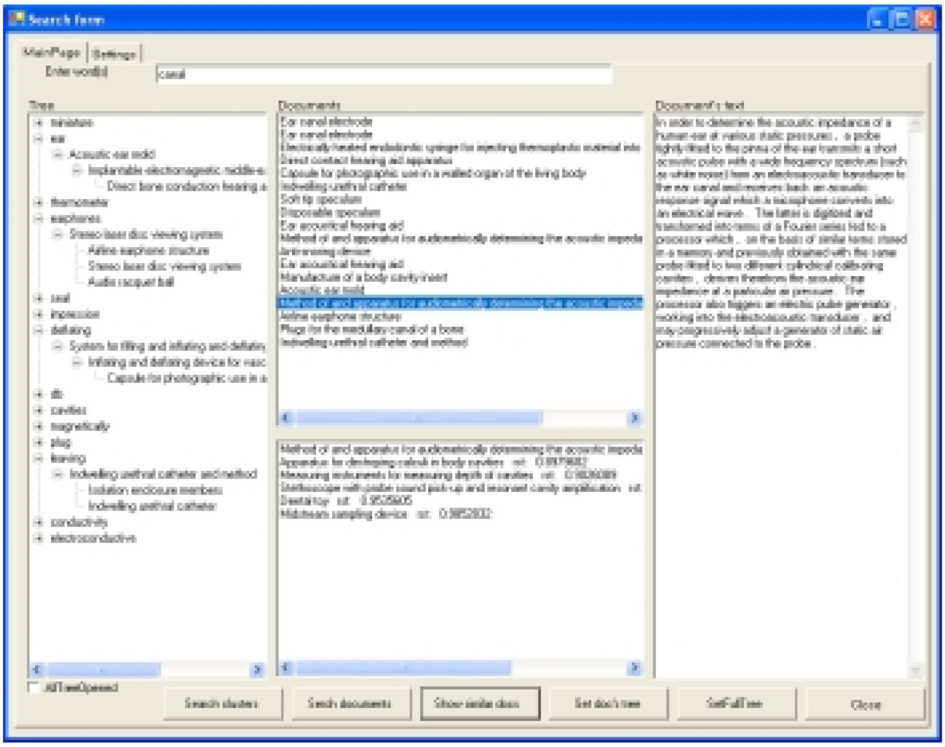
For decrease of the processing and cluster build time the documents and words indexing is applied. In all tables there are stored not words and documents, but their indexes. The accordance between documents, words and their indexes is stored in documents and dictionary tables.

3.3. IRS use.

System user has the ability to organize information search by entering words or a whole question, what is better for non-experienced user. User has the ability to choose keyword or cluster based search.

Keyword search result is a list of documents sorted according to their relevancy to the request.
Cluster based search result is a list of clusters sorted according to their relevancy to the request. In visual user interface every cluster name is a root of a tree, where every node is a document of this cluster. While looking through tree nodes the user has the ability to see text of the document in a separate window. Selecting one document there is the possibility to find relevant documents to the selected one.

System search interface is presented on the following picture.



Enter word(s): canal

Tree:

- miniature
- ear
  - Acoustic ear mold
    - Implantable electromagnetic middle-e
      - Direct bone conduction hearing a
- thermometer
- earphones
  - Stereo laser disc viewing system
    - Airline earphone structure
    - Stereo laser disc viewing system
    - Audio racquet ball
- seal
- impression
- deflating
  - System for filling and inflating and deflating
    - Inflating and deflating device for vasc
      - Capsule for photographic use in a
- db
- cavities
- magnetically
- plug
- leaving
  - Indwelling urethral catheter and method
    - Isolation enclosure members
    - Indwelling urethral catheter
- conductivity
- electroconductive

Documents:

Ear canal electrode
Ear canal electrode
Electrically heated endodontic syringe for injecting thermoplastic material into
Direct contact hearing aid apparatus
Capsule for photographic use in a walled organ of the living body
Indwelling urethral catheter
Soft tip speculum
Disposable speculum
Ear acoustical hearing aid
Method of and apparatus for automatically determining the acoustic impeda
Anti-snoring device
Ear acoustical hearing aid
Manufacture of a body cavity insert
Acoustic ear mold
Method of and apparatus for automatically determining the acoustic impeda
Airline earphone structure
Plugs for the medullary canal of a bone
Indwelling urethral catheter and method

Method of and apparatus for automatically determining the acoustic impeda
Apparatus for destroying calculi in body cavities rel: 0.8979682
Measuring instruments for measuring depth of cavities rel: 0.9836089
Stethoscope with probe sound pick-up and resonant cavity amplification rel:
Dental toy rel: 0.9526605
Midstream sampling device rel: 0.9852832

Document's text:

In order to determine the acoustic impedance of a human ear at various static pressures , a probe tightly fitted to the pinna of the ear transmits a short acoustic pulse with a wide frequency spectrum (such as white noise) from an electroacoustic transducer to the ear canal and receives back an acoustic response signal which a microphone converts into an electrical wave . The latter is digitized and transformed into terms of a Fourier series fed to a processor which , on the basis of similar terms stored in a memory and previously obtained with the same probe fitted to two different cylindrical calibrating cavities , derives therefrom the acoustic ear impedance at a particular air pressure . The processor also triggers an electric pulse generator , working into the electroacoustic transducer , and may progressively adjust a generator of static air pressure connected to the probe .

4. Experiments

4.1 Testing methods

Document classification is one of the well known measures of clustering process quality.
IRS was tested in classification environment. A collection of law documents was used in this test.

Test goals were:

- algorithm accuracy test
- different classification methods comparison
- algorithm efficiency evaluation

In the collection there were documents devoted to different topics. There were 60,000 documents in the collection. There were given 100 topics and for every topic there was a training set, a set of documents devoted to this topic. The volume of a training set was near 5% of the collection size. Classification was provided using three versions of the algorithm. There were used two clustering versions and two classification versions.

For the first clustering method (A1) the dictionary was build. For dictionary build words with $df(y) \le 2$, $df(y) \ge 1000$, and $tf(y) \le 5$, $tf(y) \ge 1000$ from the main dictionary were deleted.

For the second clustering method (A2) documents which were larger then most collection documents were not taken into account when contexts were build. Such large documents increase contexts build time very much. As these documents number was only 15% of all collection size, then their removal while words contexts were build did not influence on the results much. When clusters were build these documents were taken into account.

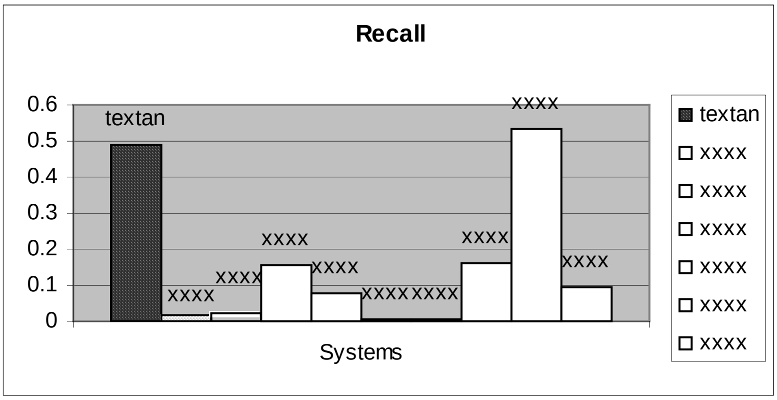
The classification methods are fully described in 2.6 section. In testing there were used both the first (CM1) and the second (CM2) classification methods.
For both classification methods every test document was classified in 5 of 100 topics maximum

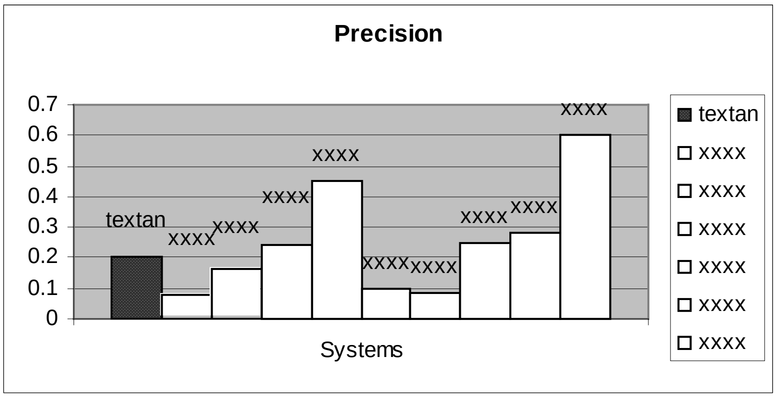
Taking into account different approaches to dictionary generation we used three versions of classification algorithm in this experiment.

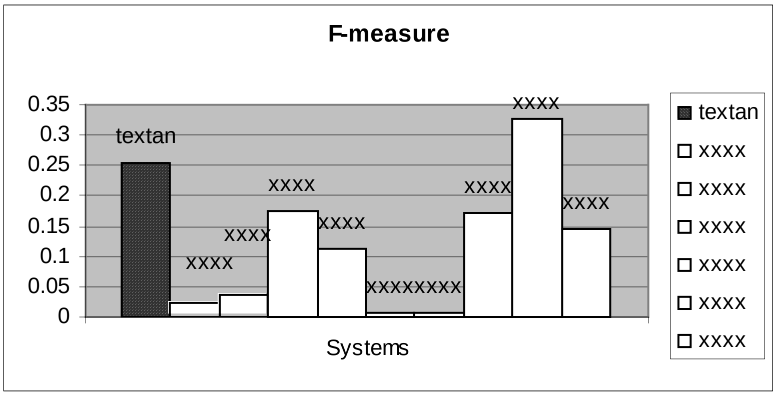
The first one (textan) depends on clustering method A1 and classification method CM2.
The second one (docsan) depends on clustering method A1 and classification method CM1.
The third one (dicsan) depends on clustering method A2 and classification method CM1.

4.2 Results analysis

Results were evaluated by Russian Information Retrieval Evaluation Seminar (http://romip.narod.ru/) using their specific methods and software. Such measures as macro-average recall, precision and F-measure were calculated. Precision is how many relevant documents are retrieved as a result of a query. Recall is how many relevant documents in the corpus are actually returned. F-measure joins the recall and precision measures. The results are shown on the next figures. Other systems which took part in the seminar pointed as xxxx. All the seminar results may be found at http://romip.narod.ru/romip2004/index.html .







Next table shows list of some topics test documents were classified in.

№	Category
1	Family law
2	Inheritance law
3	Water industry
4	Catering
5	Inhabitants' consumer services
6	Rent truck

7	International law of the space
8	Territory in international law
9	Off-economic relations fellows
10	Off-economic dealerships
11	Economy free trade zones. Customs unions.

Recall results for every category are put in the following table. Results which were the best for the category are selected with bold type. All results are set in percents.

C V	1	2	3	4	5	6	7	8	9	10	11
textan	33	34	35	**60**	46	26	27	**98**	**75**	25	**100**
xxxx	1	0	0.2	3	4	0	0.9	0	3	0	2
xxxx	0	0	4.3	2.3	0	5	0.9	8	3	0	0.8
xxxx	55	86	75	19	59	51	80	0	41	82	0
xxxx	21	39	2	22	15	6	0	1.4	0	5	0
xxxx	40	43	16	11	25	23	10	1.4	1.2	5	0
xxxx	23	4	2.5	1.1	18	7	0.9	0	1.2	10	0
xxxx	2.7	0	0	0	1.5	0	0	0	0	0	0
xxxx	2.2	0	0	0	1.5	0	0	0	0	0	0
xxxx	37	21	12	22	18	27	51	0	0	0	0

First method (textan) recall is near 50% and second and third method (docsan, dicsan) recalls are worse. As the second and third method used the same classification methods and the first and the second methods used the same clustering methods, then it is clear that the most influence is provided by the classification algorithm. The second classification algorithm was then more effective.

The results show that current version of classification algorithm returns plentiful set of documents which contains almost all necessary. As IRS has the ability to sort documents according to the given topic then it is possible to present results so that the more relevant documents are situated in the beginning of the list. This can be used to increase classification precision.

5. Literature.

[1] Dobrynin V., Patterson D, Rooney N., Contextual Document Clustering, Lecture Notes in Computer Science № 2997, Advances in Information Retrieval, 2004 year, p.167-180.

[2] Dobrynin V., Patterson D, Rooney N. Galushka. N., UK Patent Application № 0322600.8.UK Patent Office 25.09.2003

[3] N. Tishby and F. C. Periera and W. Bialek., The Information Bottleneck Method.

[4] http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/index.html

[5] Sebastiani F., Machine Learning in Automated Text Categorization.