

U.ERGÜN

**EVALUATION OF TAXONOMY BASED
CONCEPT EXTRACTION SYSTEM COSMIX
CASE FOR TEXT CATEGORIZATION**

UMUT ERGÜN

ÇANKAYA UNIVERSITY

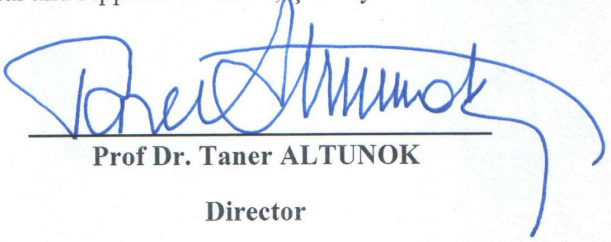
APRIL 2011

Title of the Thesis : **Evaluation of Taxonomy Based Concept Extraction System**

Cosmix: Case For Text Categorization


Submitted by **Umut ERGÜN**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya
University




Prof Dr. Taner ALTUNOK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of
Master of Science.



Prof Dr. Mehmet R. TOLUN
Head of Department


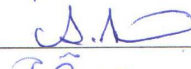

This is to certify that we have read this thesis and that in our opinion it is fully
adequate, in scope and quality, as a thesis for the degree of Master of Science.



Asst. Prof. Dr. Abdül Kadir GÖRÜR
Supervisor

Examination Date: 28.04.2011

Examining Committee Members

Asst. Prof. Dr. Abdül Kadir GÖRÜR (Çankaya Univ.) 
Dr. Ali Rıza AŞKUN (Çankaya Univ.) 
Asst. Prof. Dr. Tansel ÖZYER (TOBB ETU Univ.) 

STATEMENT OF NON- PLAGIARISM

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Umut ERGÜN

Signature : 

Date : 16.09.2011

ABSTRACT

Evaluation of Taxonomy Based Concept Extraction System

Cosmix: Case for Text Categorization

Umut, ERGÜN

M.Sc., Department of Computer Engineering

Supervisor: Asst. Prof. Dr. Abdül Kadir GÖRÜR

April 2011, 36 Pages

The aim of this study is creating a Document Classification system using Vector Space Model as baseline classifier. Cosine similarity is used to calculate similarity between Training Set and Test Set. Finally similar files are used to suggest topics for test files. Same method is used to create Kosmix Training and Test Sets and suggest topics. Results are compared and comparison results shown that Cosine similarity method is more successful.

Keywords: Vector Space Model, Machine Learning, Kosmix, Cosine Similarity

ÖZ

Taksonomi Bazlı Konsept Çıkarım Sistemi Kosmix'in

Text Kategorizasyonu Alanında Değerlendirilmesi

Umut, ERGÜN

Yüksek Lisans , Bilgisayar Mühendisliği Bölümü

Danışman: Asst. Prof. Dr. Abdül Kadir GÖRÜR

Nisan 2011, 36 Sayfa

Bu çalışmanın amacı Vektör Uzay Modeli kullanılarak bir Doküman sınıflandırma sistemi ortaya koymaktır. Daha sonra kosinüs benzerliği kullanarak öğrenme dokümanları ve test dokümanları arasında benzerlikleri hesaplanmıştır. Son olarak yüksek benzerlikli dosyalar üzerinden kategori tahminlemede bulunulmuştur. Aynı tahminleme sistemi Kosmix dosyaları üzerinde de uygulanarak sonuçlar karşılaştırılmıştır. Kosinus benzerliği metodunun daha başarılı olduğu sonucuna varılmıştır.

Keywords: Vektör Uzay Modeli, Makine Öğrenimi, Kosmix, Kosinüs Benzerliği

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciation to my supervisor Assistant Professor Dr. Abdül Kadir GÖRÜR for wisdom, foresight and the guidance he provided throughout the thesis completion process.

TABLE OF CONTENTS

| | |
|------------------------------------|-----|
| STATEMENT OF NON-PLAGIARISM..... | iii |
| ABSTRACT | iv |
| ÖZ..... | v |
| ACKNOWLEDGEMENTS | vi |
| TABLE OF CONTENTS | vii |
| LIST OF TABLES..... | xi |
| LIST OF FIGURES..... | xii |
| CHAPTERS : | |
| 1. INTRODUCTION | 1 |
| 2. LITERATURE SURVEY | 4 |
| 2.1 Categorization techniques..... | 4 |
| 2.1.1 Decision trees | 4 |
| 2.1.2 Naive bayes approach | 5 |
| 2.1.3 Neural networks | 6 |
| 2.1.4 Decision rules..... | 7 |

| | |
|---|----|
| 2.1.5 K Nearest neighbors..... | 8 |
| 2.1.6 Regression based classification..... | 8 |
| 2.1.7 Vector based classification..... | 9 |
| 3. REUTERS & KOSMIX..... | 10 |
| 3.1 Reuters | 10 |
| 3.2 Kosmix | 11 |
| 4. EVALUATION OF TAXONOMY BASED CONCEPT EXTRACTION SYSTEM KOSMIX: CASE FOR TEXT CATEGORIZATION | 14 |
| 4.1 Overview | 14 |
| 4.2 Components | 15 |
| 4.2.1 Sample reuters file..... | 15 |
| 4.2.2 Sample kosmix file..... | 17 |
| 4.2.3 Stemmer | 20 |
| 4.2.4 Stop words..... | 20 |
| 4.3 Vector Space Model and Cosine Similarity..... | 21 |
| 4.4 Classification Steps..... | 22 |

| | |
|---|----|
| 4.4.1 Reuters corpus subset extraction..... | 22 |
| 4.4.2 Topic selection | 25 |
| 4.4.3 Training set extraction..... | 26 |
| 4.4.4 Test set extraction | 27 |
| 4.4.5 Kosmix retrieval..... | 27 |
| 4.4.6 Reuters training set..... | 28 |
| 4.4.7 Reuters test set..... | 29 |
| 4.4.8 Classification logic..... | 31 |
| 4.4.9 Kosmix classification | 31 |
| 5. EXPERIMENTS & EVALUATION..... | 32 |
| 5.1 Calculations..... | 32 |
| 5.2 Reuters Detailed Results | 32 |
| 5.3 Kosmix Detailed Results..... | 34 |
| 5.4 Comparison..... | 34 |
| 6. CONCLUSIONS AND FUTURE WORK..... | 36 |
| 6.1 Conclusions..... | 36 |

| | |
|-----------------------|----|
| 6.2 Future Work | 36 |
| REFERENCES..... | R1 |
| APPENDICES: | |
| A.Stop Words | A1 |
| B.CV..... | A3 |

LIST OF TABLES

TABLES

| | |
|-------------------------------|----|
| Table 1 Reuters Results | 33 |
| Table 2 Kosmix Results | 34 |

LIST OF FIGURES

FIGURES

| | |
|---|----|
| Figure 1 Kosmix Taxonomy..... | 11 |
| Figure 2 Topics Related to Pinot Noir..... | 13 |
| Figure 3 Sample Reuters XML File | 16 |
| Figure 4 Sample Kosmix Output..... | 19 |
| Figure 5 Reuters Subset Extraction | 25 |
| Figure 6 Actual Topics in Reuters News Corpus | 26 |
| Figure 7 Kosmix Retrieval Steps..... | 28 |
| Figure 8 Reuters Training Set | 29 |
| Figure 9 Reuters Test Set | 30 |

CHAPTER 1

INTRODUCTION

With the invention of Internet, life around the globe has changed drastically. Before the Internet, access to information was limited at best. Paper-back sources like books and newspapers, television and radio were the common information sources accessible. Yet the information flow was controlled by the origin of the information.

During the early Internet era, everyone that can afford a personal computer and an internet connection gained access to incredible amount of information. Websites were developed by companies; newspapers became digital and the online document access flourished. Still slow connection speeds, low capacity storage devices and relatively low computer speeds limited the information flow to a degree. Today we have fiber optic cables, large capacity reliable hard-disks and advanced speed computers that lets the users access any information they wish.

With the web 2.0 development, information flow is not only limited to the website origin, and users can influence and inform the rest of the web about any desired

topic. Technological changes and developments mentioned above require new approach and methodologies unless the advance becomes obsolete. Without proper classification and categorization Petabytes of information remain as an unprocessed and quite meaningless bulk of data. While the origin of the information could tag and categorize the actual data, bulk data received from unknown origin remains useless unless properly identified. The need for a mechanism to auto classify data is obvious. According to Sebastiani document classification can be defined as: “Text categorization (TC – also known as text classification, or topic spotting) is the task of automatically sorting a set of documents into categories (or classes, or topics) from a predefined set.”[1]

In order to be able to automatically categorize a set of documents, a set of already categorized documents should be presented to the system. System learns from the training set of documents and creates a relation between newly presented document and the training set to successfully suggest a category for the new document. This process is defined as Unsupervised Document Classification. If human interaction is requested before the decision process the flow becomes supervised and therefore named as Supervised Document Classification. Both methods are based on Machine Learning [1].

Most of the classification methods treat documents as bag of words [2] which usually causes loss of sentence structure, therefore word groups or nouns and verbs have no special weighting. Also it is imperative to choose a set of training documents that

covers most of the words that are related to that category, so when a new document is added all its words are already contained and weighted in the system. Representing all available categories equally in the training set ensures the best possible learning achieved in the system.

Document classification benefits span from email filtering to category based search engines, mail routing to news monitoring and content classification [3].

CHAPTER 2

LITERATURE SURVEY

Document classification efforts date back to 1960's starting with Harold Borko, Myrna Bernick and Lauren B. Doyle's works. Since then, with the application of several mathematical methods, Information Retrieval Society developed different techniques. Main categorization techniques are decision trees, Naive-Bayes approach, neural networks, decision rules, k-nearest neighbors, regression-based and vector-based methods.

2.1 Categorization Techniques

2.1.1 Decision Trees

Decision tree methods rebuild the manual categorization of the training documents by constructing well-defined true/false-queries in the form of a tree structure where the nodes represent questions and the leafs the corresponding category of documents [4]. After building the decision tree new document can be run through the tree and

classified properly according to predefined rules. Since decision tree contains the classification logic, it is easy to apply to a new document by any automated system available.

A risk of the application of tree methods is known as "over fitting": A tree over fits the training data if there exists an alternative tree that categorizes the training data worse but would categorize the documents to be categorized later better [4]. This circumstance is the result of the algorithm's intention to construct a tree that categorizes every training document correctly; however, this tree may not be necessarily well suited for other documents. This problem is typically moderated by using a validation data set for which the tree has to perform in a similar way as on the set of training data [5].

Other techniques to prevent the algorithm from building huge trees (that anyway only map the training data correctly) are to set parameters like the maximum depth of the tree or the minimum number of observations in a leaf. If this is done, Decision Trees show very good performance even for categorization problems with a very large number of entries in the dictionary [4].

2.1.2 Naive-Bayes Approach

There are two groups of Bayesian approaches in document categorization: Naive and non-Naive Bayesian approaches. The naive part of the former is the assumption of word (i.e. feature) independence, meaning that the word order is irrelevant and

consequently that the presence of one word does not affect the presence or absence of another one. This assumption makes the computation of Bayesian approaches more efficient. But although the assumption is obviously severely violated in every language, it has been shown that the classification accuracy is not seriously affected by this kind of violations [6]. Nevertheless, several non-naive Bayesian approaches eliminate this assumption [7].

Naive Bayesian approaches have been developed comparatively early and have been studied frequently in data mining before the topic of document categorization gained importance. They perform as well as newer, more sophisticated methods [8] and also show a very good runtime-behavior during the categorization of new documents [9]. A disadvantage of Bayesian approaches in general is that they can only process binary feature vectors [7] and, thus, have to abandon possibly relevant information.

2.1.3 Neural Networks

Different neural network approaches have been applied to document categorization problems. While some of them use the simplest form of neural networks, known as perceptrons, which consist only of an input and an output layer [10], others build more sophisticated neural networks with a hidden layer between the two others [11]. In general, these feed-forward-nets consist of at least three layers (one input, one output, and at least one hidden layer) and use back propagation as learning mechanism [12]. However, the comparatively old perceptron approaches perform surprisingly well [10].

The advantage of neural networks is that they can handle noisy or contradictory data very well [12]. Furthermore some types of neural networks are able to comprehend fuzzy logic [13], but one has to change from back propagation as learning mechanism to counter propagation (for which worse categorization results are reported [11]). The advantage of the high flexibility of neural networks entails the disadvantage of very high computing costs. Another disadvantage is that neural networks are extremely difficult to understand for an average user; this may negatively influence the acceptance of these methods. [5]

2.1.4 Decision Rules

Decision rule algorithms construct for every category a rule set that describes the profile of this category. In general, a single rule consists of a category name and a feature of the dictionary which is typical for the training documents belonging to the considered category.

Then the rule set is created by combining the separate rules with the logical operator "or". Usually not all of the rules are required to categorize the documents adequately. Therefore, heuristics are applied to reduce the size of the rule sets. The goal is to achieve a reduced rule set per category which, however, does not affect the categorization of the training documents [14].

2.1.5 K-Nearest Neighbors

The k-Nearest Neighbor algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space. K-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors is a positive integer, typically small. If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. In binary (two class) classification problems, it is helpful to choose k to be an odd number as this avoids tied votes. [15]

2.1.6 Regression Based Classification

For this method the training data are represented as a pair of input/output matrices where the input matrix is identical to our feature matrix A and the output matrix B consists of flags indicating the category membership of the corresponding document in matrix A. Thus B has the same number of rows like A (namely m) and c columns where c represents the total number of categories defined. The goal of the method is to find a matrix F that transforms A into B' (by simply computing $B'=A * F$) so that B' matches B as well as possible. The matrix F is determined by applying multivariate regression techniques [16].

2.1.7 Vector Based Classification

One of the simplest categorization methods is the centroid algorithm. During the learning stage only the average feature vector for each category is calculated and set as centroid-vector for the category [17].

Unless the document clusters overlap each other, this method does not need many training documents. If, however, the document clusters overlap each other or the category consists of two or more different topics (clusters), the algorithm performs often poor. The method is also inappropriate if the number of categories is very large [4].

CHAPTER 3

REUTERS & KOSMIX

3.1 Reuters

Reuters is a well-known news agency based on United Kingdom. In 2000, Reuters Ltd made available a large collection of Reuters News stories for use in research and development of natural language processing, information retrieval, and machine learning systems. This corpus, known as "Reuters Corpus, Volume 1" or RCV1 is significantly larger than the older, well-known Reuters-21578 collection heavily used in the text classification community. Corpus is a collection of XML files that include necessary information such as Title, Headline, Dateline, Text and Codes. Typical XML file has more than one Code concerning topics that file text is belonging to. In our studies we have processed Title, Headline and Text information of files as a base for categorization. Corpus includes 109993 files.

3.2 Kosmix

Kosmix is a categorization engine which organizes the Internet into topic pages allowing users to explore the Web by topic, "presenting a dashboard of relevant videos, photos, news, commentary, opinion, communities and links to related topics"[18]The cornerstone of the Kosmix explore engine is its taxonomy and categorization technology. The Kosmix taxonomy consists of millions of topics organized hierarchically, reflecting is-a relationships. For example, San Francisco is-a city. The resulting hierarchical structure is a directed acyclic graph (DAG).

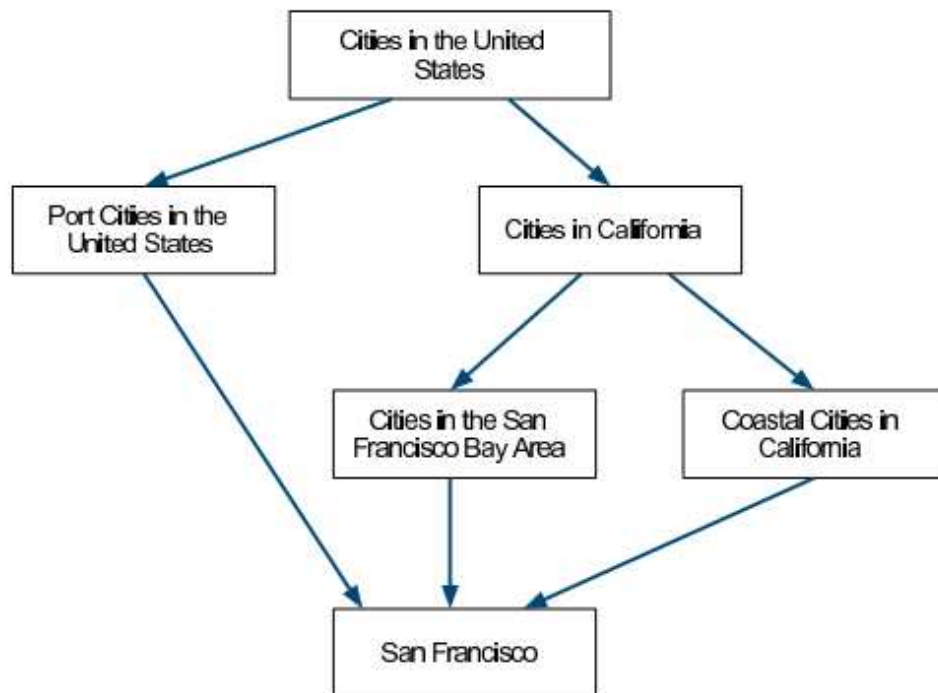


Figure 1 A small fragment of the Kosmix taxonomy

Figure 1 shows a small piece of the Kosmix taxonomy. The taxonomy also encodes many relationships beyond is-a. For example, there is a member-of relationship connecting a music group with its members, and a capital-of relationship connecting a country with its capital city. There are many thousands of such relationship types captured in the taxonomy. The taxonomy has been built over three years using a combination of human curation and algorithmic methods. The raw materials include several publicly available taxonomies, such as DMOZ and Wikipedia, as well as hundreds of special purpose taxonomies in specific fields, such as health, automobiles, and music. The details of how the taxonomy is created and maintained need not concern us here, but the technical challenges we had to surmount include: Merging overlapping taxonomies, taking into account that the same concept might be named differently in the two taxonomies.

Keeping the taxonomy up to date by identifying new topics on an ongoing basis. At Kosmix we gather and analyze millions of RSS feeds every day to identify new topics, such as people, music groups, and so on. The second key to the Kosmix explore engine is the Kosmix Categorization Service (KCS).

Given a user query, KCS determines the nodes in the taxonomy that are most closely connected with the query. The details of the algorithms involved are proprietary to Kosmix and are not relevant to the discussion here. We will content ourselves with an example illustrating the functionality provided by KCS. Let us say the query is “Pinot Noir.” KCS determines that Pinot Noir is a kind of wine, which is a related to

foods and beverages. It also determines that Pinot Noir is a kind of wine grape, and is related to viticulture and vineyards. Figure 2 shows a small selection of the full list of topics KCS determines are related to Pinot Noir. These topics are displayed on the topic page in the section titled Related in the Kosmos[19].



Figure 2 Topics related to Pinot Noir

CHAPTER 4

EVALUATION OF TAXONOMY BASED CONCEPT EXTRACTION SYSTEM KOSMIX: CASE FOR TEXT CATEGORIZATION

4.1 Overview

Our study consists of two main components: Native classifier and Kosmix classifier. Native classifier is based on solely Reuters News Files that we have selected as training and test sets. Kosmix classifier uses the Kosmix output we have received after submitting the Reuters News File into Kosmix API. Forming training set out of Reuters News File set requires that we homogenously select an equal number of files that represent each topic. If some topics are not represented in the training set we may not use the test set to achieve accurate results. Test set is also based on same logic and contains a number of files that represent each topic.

While Reuters test and training sets are enough for native classifier, Kosmix classifier requires that we represent news files in terms of Kosmix categories and

entities. Kosmix API only accepts documents that have more than 500 characters submitted.

4.2 Components

4.2.1 Sample Reuters File

```

<?xml version="1.0" encoding="iso-8859-1" ?>
<newsitem itemid="31787" id="root" date="1996-09-04" xml:lang="en">
<title>BRAZIL: Rainstorm kills one in Rio de Janeiro.</title>
<headline>Rainstorm kills one in Rio de Janeiro.</headline>
<dateline>RIO DE JANEIRO 1996-09-04</dateline>
<text> <p>A rainstorm with hail and winds of 40 miles per hour (65 kph) killed a woman
and wreaked havoc in Rio de Janeiro, disaster relief officials said on
Wednesday.</p> <p>The storm, which left many Rio neighbourhoods temporarily without
electricity, raged for about an hour on Tuesday evening, knocking down trees, flooding
streets and blowing tiles off roofs, a spokeswoman for Rio's disaster relief agency
said.</p> <p>One woman was killed when a newspaper stand blown over by the wind fell on
top of her, the spokeswoman said.</p> <p>Santos Dumont domestic airport was shut down
because of the storm and traffic jams formed everywhere. Models and visitors at Rio de
Janeiro's spring and summer fashion show had to run for cover in high heels and
miniskirts when the wind blew down the tent over the catwalk.</p> <p>A spokesman for the
National Meteorological Institute said a cold front approaching from the south had
clashed with the city's hot air masses, causing the sudden tropical
rainstorm.</p> </text>
<copyright>(c) Reuters Limited 1996</copyright>
<metadata>
<codes class="bip:countries:1.0">
  <code code="BRAZ">
    <editdetail attribution="Reuters BIP Coding Group" action="confirmed"
date="1996-09-04"/>
  </code>
</codes>
<codes class="bip:topics:1.0">
  <code code="GCAT">
    <editdetail attribution="Reuters BIP Coding Group" action="confirmed"
date="1996-09-04"/>
  </code>
  <code code="GDIS">
    <editdetail attribution="Reuters BIP Coding Group" action="confirmed"
date="1996-09-04"/>
  </code>
  <code code="GWEA">
    <editdetail attribution="Reuters BIP Coding Group" action="confirmed"
date="1996-09-04"/>
  </code>
</codes>
<dc element="dc.date.created" value="1996-09-04"/>
<dc element="dc.publisher" value="Reuters Holdings Plc"/>
<dc element="dc.date.published" value="1996-09-04"/>
<dc element="dc.source" value="Reuters"/>
<dc element="dc.creator.location" value="RIO DE JANEIRO"/>
<dc element="dc.creator.location.country.name" value="BRAZIL"/>
<dc element="dc.source" value="Reuters"/>
</metadata>
</newsitem>

```

Figure 3 Sample Reuters XML File

Typical Reuters News File (Figure 2) consists of several sections to include

necessary information. In our study we have used title, headline and text sections combined as text that is represented by the news file. As a classification medium we have used codes section in the bottom. Only Bip Topics are used in classification and Country Code is discarded.

4.2.2 Sample Kosmix File

When we read a news file, we take title, headline and text sections and combine them into a single text object. The object is then submitted to Kosmix API to receive the Kosmix representation of the given text.

While we use text as classification base in Reuters Files, in Kosmix Files(Figure 2) Categories and Entity Names as single words. Since Kosmix File does not have a corresponding codes section as similar to Reuters News File, we use the original Reuters News File as code basis. Since Kosmix File is the result of Reuters News File text extraction it is an appropriate practice to use its codes for Kosmix File as well.

```

{
  "categories": [{
    "id": 6662985,
    "name": "ScienceAndNature",
    "score": 0.678818
  }],
  "langid": [{
    "l": "en",
    "score": 1
  }],
  "mentions": [
    {
      "EntityClassID": "7899871",
      "EntityClassName": "Financial services",
      "EntityID": 8197491,
      "EntityName": "Morgan Stanley Smith Barney",
      "EntityScore": 0.957621,
      "id": "mid25998349",
      "value": "smith barney"
    },
    {
      "EntityClassID": "7848713",
      "EntityClassName": "Companies",
      "EntityID": 1419010,
      "EntityName": "Freese-Notis",
      "EntityScore": 0.919813,
      "id": "mid25998350",
      "value": "freesenotis"
    },
    {
      "EntityClassID": "7840958",
      "EntityClassName": "Cities",
      "EntityID": 7836476,
      "EntityName": "Chicago, Illinois",
      "EntityScore": 0.768499,
      "id": "mid25998351",
      "value": "chicago"
    },
    {
      "EntityClassID": "",
      "EntityClassName": "",
      "EntityID": 8147073,
      "EntityName": "Tropical cyclones",
      "EntityScore": 0.766953,
      "id": "mid25998352",
      "value": "hurricane edouard"
    },
    {
      "EntityClassID": "",
      "EntityClassName": "",
      "EntityID": 7991486,
      "EntityName": "Midwestern United States",
      "EntityScore": 0.530417,
      "id": "mid25998353",
      "value": "us midwest"
    }
  ],

```

```

{
  "EntityClassID": "8028609",
  "EntityClassName": "Organisms",
  "EntityID": 7978698,
  "EntityName": "Maize",
  "EntityScore": 0.391085,
  "id": "mid25998354",
  "value": "corn"
},
{
  "EntityClassID": "8028609",
  "EntityClassName": "Organisms",
  "EntityID": 8173397,
  "EntityName": "Wheat",
  "EntityScore": 0.232198,
  "id": "mid25998355",
  "value": "wheat"
},
{
  "EntityClassID": "",
  "EntityClassName": "",
  "EntityID": 1430599,
  "EntityName": "Frost",
  "EntityScore": 0.105154,
  "id": "mid25998356",
  "value": "frost"
},
{
  "EntityClassID": "",
  "EntityClassName": "",
  "EntityID": 2607197,
  "EntityName": "Moisture",
  "EntityScore": 0.0718463,
  "id": "mid25998357",
  "value": "moisture"
},
{
  "EntityClassID": "",
  "EntityClassName": "",
  "EntityID": 3898652,
  "EntityName": "Threat",
  "EntityScore": 0.015085,
  "id": "mid25998358",
  "value": "threat"
}
],
"message": {
  "id": "25998359",
  "timestamp": "2011-04-03 15:09:17.686"
},
"urlcats": []

```


Figure 4 Sample Kosmix Output

4.2.3 Stemmer

In linguistic morphology, **stemming** is the process for reducing inflected (or sometimes derived) words to their stem, base or root form – generally a written word form. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root. Algorithms for stemming have been studied in computer science since 1968. Many search engines treat words with the same stem as synonyms as a kind of query broadening, a process called **conflation**. [20]

For further explanation if we take a Sample Reuters News File that has “Fishing” word as text and use it as a training set for the system; and take a News File that has “Fish” as a test file we would find no similarities in between. Therefore it is a necessary step to process text content of the files through stemmer to find a common ground in between.

4.2.4 Stop Words

Stopwords are set of characters that almost occur in every file and therefore irrelevant for topic classification. If we can find a word that only occurs in a single file, and if we submit a test file that has that word, we would definitely classify the test file as training file. So the less a word occurs in whole training set, the more

chance we have to classify another file based on that word. As an example, day and month names are present mostly in every file and removed as stop words. Appendix C includes stopwords used in our system.

4.3 Vector Space Model and Cosine Similarity

Vector Space Model [21] indicates that weight of a term can be calculated with the formula

$$w_i = tf_i * \log\left(\frac{D}{df_i}\right) \quad (4.1)$$

tf_i = term frequency (term counts) or number of times a term i occurs in a given document.

df_i = document frequency or number of documents containing term i

D = number of documents in a selected set.

Vector Space Model forms a base for calculating weights of every term in given set of file. Since Model does not distinguish between common used and rarely used terms, it must be applied together with stop word removal and stemming to improve classification performance. Model also ignores where the term occurs therefore relations with other terms are disregarded.

A document is represented by all its terms such as we first calculate term weight for every single term then we take squares of all weights, sum them, and take square root of the result. Final value is the vector that represents our document in the space model. [21]

$$\cos \theta = \frac{\mathbf{d}_2 \cdot \mathbf{q}}{\|\mathbf{d}_2\| \|\mathbf{q}\|} \quad (4.2)$$

In order to calculate similarity between two documents angle between them, dot products of the two documents is divided into the multiplication of norms.

Vector space model has some disadvantages in providing the accurate results in classification. Comparing two documents representing the same category with different words would not have a positive result. On the other hand, two files with different categories but similar words May have a false positive result. Also documents with too many terms may not be accurately represented.[23]

4.4 Classification Steps

4.4.1 Reuters Corpus Subset Extraction

Kosmix api does not provide meaningful results for files that have less than 500 characters. Since we need to represent our files in terms of Kosmix results, first step was to extract the files that have more than 500 characters and form another subset. Resulting set of files that have more than 500 characters, Title, Headline and Text

combined are at total of 44565. On subset extraction we did not use stemmer since number of words remains the same but we used stop word removal before counting the actual number of characters.

Reuters Subset Extraction

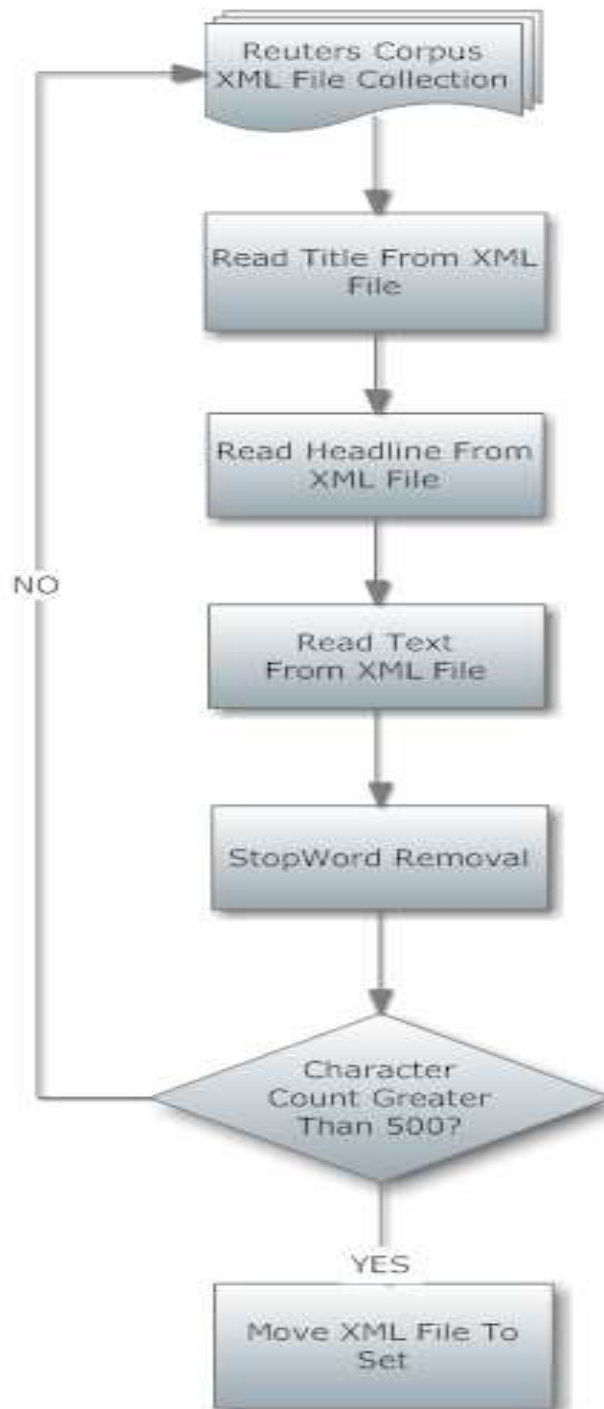


Figure 5 Reuters Subset Extraction

4.4.2 Topic Selection

In order to form homogenous Training and Test sets, first we have analyzed the whole set of Reuters Corpus. Results indicated some topics are represented in a very small subset of files. Topics that are contained in less than 200 files are excluded from the set. We have also ruled out the topics that are always containing other topics. For example files that are having topics as GCAT also always had GDEF, GREL and GJOB. Selected topics for categorization process are : GDEF, GDIP , GDIS , GENT , GENV ,GHEA , GJOB , GPRO , GREL , GSCI , GSPO , GVIO , GVOTE , GWEA , ECAT.

- 1POL CURRENT NEWS - POLITICS
- 2ECO CURRENT NEWS - ECONOMICS
- 3SPO CURRENT NEWS - SPORT
- 4GEN CURRENT NEWS - GENERAL
- 6INS CURRENT NEWS - INSURANCE
- 7RSK CURRENT NEWS - RISK NEWS
- CCAT CORPORATE/INDUSTRIAL
- ECAT ECONOMICS
- ENT12 CURRENT NEWS - ENTERTAINMENT
- GCAT GOVERNMENT/SOCIAL
- GCRIM CRIME, LAW ENFORCEMENT
- GDEF DEFENCE
- GDIP INTERNATIONAL RELATIONS
- GDIS DISASTERS AND ACCIDENTS
- GEDU EDUCATION
- GENT ARTS, CULTURE, ENTERTAINMENT
- GENV ENVIRONMENT AND NATURAL WORLD
- GFAS FASHION
- GHEA HEALTH
- GJOB LABOUR ISSUES
- GOBIT OBITUARIES
- GODD HUMAN INTEREST
- GPOL DOMESTIC POLITICS
- GPRO BIOGRAPHIES, PERSONALITIES, PEOPLE
- GREL RELIGION GREL RELIGION
- GSCI SCIENCE AND TECHNOLOGY
- GSPO SPORTS
- GTOUR TRAVEL AND TOURISM
- GVIO WAR, CIVIL WAR
- GVOTE ELECTIONS
- GWEA WEATHER
- GWELF WELFARE, SOCIAL SERVICES
- GSCI SCIENCE AND TECHNOLOGY
- GSPO SPORTS
- GTOUR TRAVEL AND TOURISM
- GVIO WAR, CIVIL WAR
- GVOTE ELECTIONS
- GWEA WEATHER
- GWELF WELFARE, SOCIAL SERVICES
- GMIL MILLENNIUM ISSUES

Figure 6 Actual Topics in Reuters News Corpus

4.4.3 Training Set Extraction

Since we have some topics that have little as 200 files in the set we have decided to take 100 files for every topic as training set. Some selected files also included other topics therefore the resulting set has 100-150 files for every topic to ensure every topic is represented in the training set. Resulting training set consists of 1556 files that has the following topics with the corresponding counts: GDEF 100, GDIP 113, GDIS 113, GENT 100, GENV 100, GHEA 100, GJOB 100, GPRO 100, GREL 100, GSCI 100, GSPO 100, GVIO 100, GVOTE 100, GWEA 100, ECAT 130

4.4.4 Test Set Extraction

We have decided to represent every topic with 50 files in our test set. Extraction process is duplicate of Training Set Extraction. As in Training Set, Test Set also has some topics represented in between 50 – 79 files. Resulting test set consists of 776 files that has the following topics with the corresponding counts: GDEF 50, GDIP 78, GDIS 69, GENT 50, GENV 50, GHEA 50, GJOB 50, GPRO 50, GREL 50, GSCI 50, GSPO 50, GVIO 50, GVOTE 50, GWEA 50, ECAT 79

4.4.5 Kosmix Retrieval

In order to be able to represent our Training and Test files in terms of Kosmix results, we needed to submit our files content to Kosmix API and saved the returning document.

Reuters Kosmix Retrieval

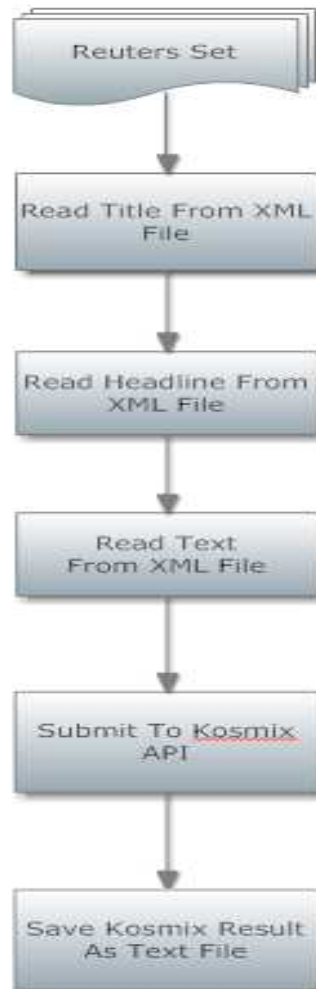


Figure 7 Kosmix Retrieval Steps

4.4.6 Reuters Training Set

Reuter's classification starts with reading the XML files text headline and title.

Combined text is stemmed and has its stop words removed. Then xml file name is inserted to Files table, topics file has is inserted to a cross table named File-Topics, if

file words are not already present in the Words table they are inserted one by one, and finally File-Words cross table keeps what file has what words. Now that we have files and words presented in the database, we first calculate word frequencies and file magnitudes based on the data.

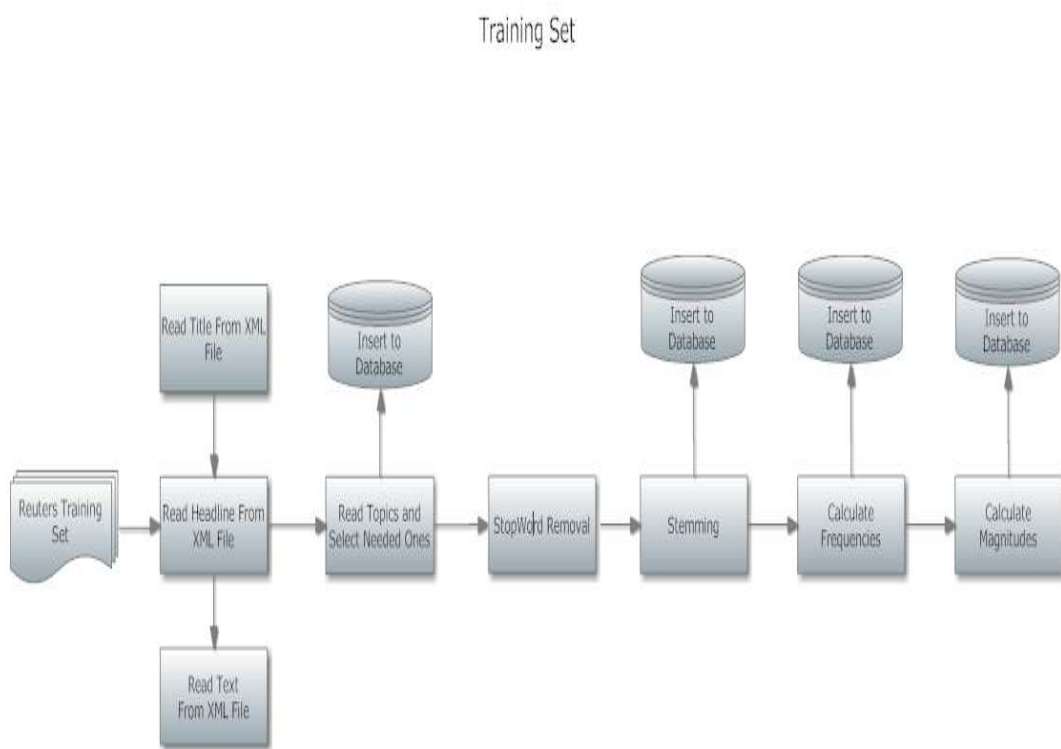


Figure 8 Reuters Training Set

4.4.7 Reuters Test Set

Now that we have formed the database that will form the base for our categorization we have to test it .Testing starts with reading an XML file from test set. We calculate magnitude of the test file as we did with training files. Then we use cosine similarity to calculate similarity between test file and the other files in the training set. We take top 5 similar documents from results, multiply the similarity values with topics, and take the topic with largest value as suggested topic. Finally we insert the actual topics and selected topic to results table.

Testing

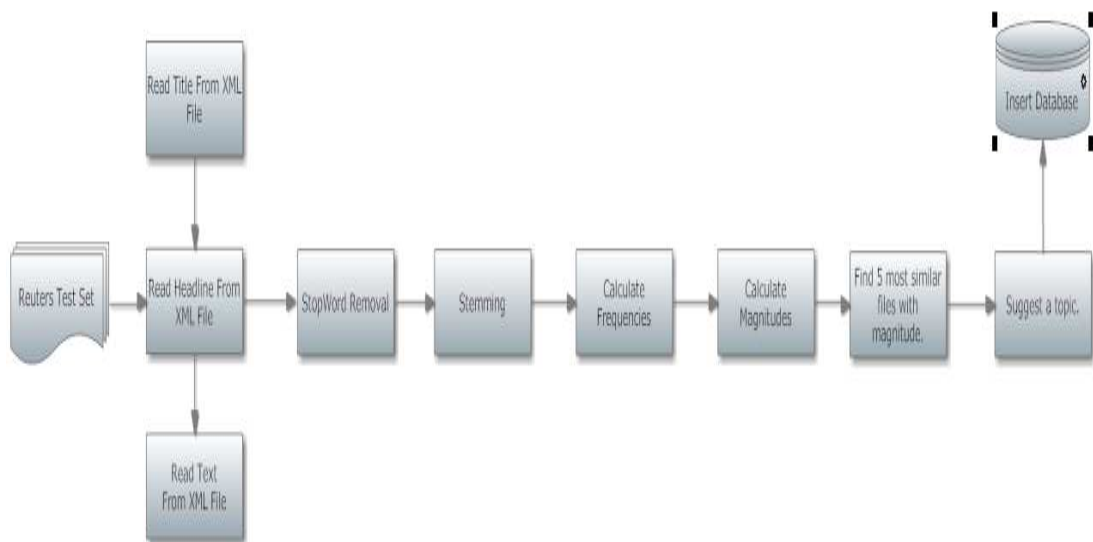


Figure 9 Reuters Test Set

4.4.8 Classification Logic

Our way of classification a test file is first calculating the cosine similarity scores with given file and whole training set then sorting them in descending order. Top 5 files are selected inside the collection and their topics are multiplied with their cosine similarity scores. If a topic occurs in 2 files, cosine similarity scores are summed up and at the end the topic with highest score is selected and suggest by the classifier as

4.4.9 Kosmix Classification

Training set creation process is very similar to Reuters Training Set creation process with only difference being instead of words taken from text part of the Reuters xml file, we take the corresponding Kosmix files categories. After the completion of database we test the data with Kosmix test set like in Reuters Process.

CHAPTER 5

EXPERIMENTS & EVALUATION

5.1 Calculations

To understand the experiment results better, precision recall metrics are used in our study. For each topic we select:

If file actual topic contains the topic and also suggested topic contains the topic it is called true positive (tp). If file actual topic does not contain the topic but suggested topic contains the topic it is called false positive (fp). If file actual topic contains the topic but suggested topic does not contain the topic it is called false negative (fn). If neither file actual topic nor suggested topic contains the topic it is called true negative (tn). Precision is $tp/(tp+fp)$. Recall is $tp/(tp+fn)$. True Negative Rate $tn/(tn+fp)$. Accuracy $(tp+tn)/(tp+tn+fp+fn)$.

F Measure = $2 \cdot (\text{Precision} \cdot \text{Recall}) / (\text{Precision} + \text{Recall})$. [22]

5.2 Reuters Detailed Results

Table 1 Reuters Results

| Topic | A(1) S(1) | A(1) S(0) | A(0) S(1) | A(0) S(0) | Precision | Recall | True Negative | Accuracy | F Measure |
|--------------|----------------------|----------------------|----------------------|----------------------|------------------|---------------|--------------------------|-----------------|----------------------|
| GVOTE | 35 | 15 | 11 | 483 | 0.7608696 | 0.7 | 0.9777328 | 0.9522059 | 0.7291667 |
| GWEA | 48 | 2 | 10 | 484 | 0.8275862 | 0.96 | 0.9797571 | 0.9779412 | 0.8888889 |
| GVIO | 19 | 32 | 24 | 469 | 0.4418605 | 0.372549 | 0.9513184 | 0.8970588 | 0.4042553 |
| GDIP | 26 | 50 | 14 | 454 | 0.65 | 0.3421053 | 0.9700854 | 0.8823529 | 0.4482759 |
| GDIS | 13 | 58 | 4 | 469 | 0.7647059 | 0.1830986 | 0.9915434 | 0.8860294 | 0.2954545 |
| ECAT | 26 | 48 | 14 | 456 | 0.65 | 0.3513514 | 0.9702128 | 0.8860294 | 0.4561403 |
| GREL | 9 | 41 | 9 | 485 | 0.5 | 0.18 | 0.9817814 | 0.9080882 | 0.2647059 |
| GJOB | 21 | 29 | 17 | 477 | 0.5526316 | 0.42 | 0.965587 | 0.9154412 | 0.4772727 |
| GPRO | 13 | 37 | 17 | 477 | 0.4333333 | 0.26 | 0.965587 | 0.9007353 | 0.325 |
| GENV | 17 | 33 | 7 | 487 | 0.7083333 | 0.34 | 0.9858299 | 0.9264706 | 0.4594595 |
| GHEA | 21 | 30 | 8 | 485 | 0.7241379 | 0.4117647 | 0.9837728 | 0.9301471 | 0.525 |
| GSCI | 30 | 20 | 20 | 474 | 0.6 | 0.6 | 0.9595141 | 0.9264706 | 0.6 |
| GSPO | 46 | 4 | 3 | 491 | 0.9387755 | 0.92 | 0.9939271 | 0.9871324 | 0.929293 |
| GDEF | 15 | 35 | 2 | 492 | 0.8823529 | 0.3 | 0.9959514 | 0.9319853 | 0.4477612 |
| GENT | 26 | 24 | 19 | 475 | 0.5777778 | 0.52 | 0.9615384 | 0.9209559 | 0.5473684 |

While accuracy results are usually quite high, F-Measure results are varying for different topics. Topic with a high F-Measure means our selected training set and test sets are sampling the topic well.

5.3 Kosmix Detailed Results

Table 2 Kosmix Results.

| Topic | A(1) S(1) | A(1) S(0) | A(0) S(1) | A(0) S(0) | Precision | Recall | True Negative | Accuracy | F Measure |
|--------------|--------------|--------------|--------------|--------------|-----------|----------|------------------|-----------|--------------|
| GVOTE | 37 | 13 | 25 | 468 | 0.596774 | 0.74 | 0.94929 | 0.9300184 | 0.6607143 |
| GWEA | 23 | 27 | 17 | 476 | 0.575 | 0.46 | 0.9655172 | 0.9189687 | 0.5111111 |
| GVIO | 23 | 28 | 15 | 477 | 0.605263 | 0.45089 | 0.9695122 | 0.9208103 | 0.5168539 |
| GDIP | 24 | 52 | 12 | 455 | 0.666667 | 0.31579 | 0.9743041 | 0.8821363 | 0.4285714 |
| GDIS | 26 | 45 | 10 | 462 | 0.722222 | 0.366197 | 0.9788136 | 0.8987108 | 0.4859813 |
| ECAT | 21 | 53 | 21 | 448 | 0.5 | 0.283784 | 0.9552239 | 0.8637201 | 0.362069 |
| GREL | 15 | 35 | 17 | 476 | 0.46875 | 0.3 | 0.9655172 | 0.9042357 | 0.3658537 |
| GJOB | 24 | 26 | 20 | 473 | 0.545455 | 0.48 | 0.9817444 | 0.9152855 | 0.5106383 |
| GPRO | 8 | 42 | 17 | 476 | 0.32 | 0.16 | 0.9756593 | 0.8913444 | 0.2133333 |
| GENV | 9 | 41 | 9 | 484 | 0.5 | 0.18 | 0.9777328 | 0.907919 | 0.2647059 |
| GHEA | 21 | 29 | 12 | 481 | 0.636364 | 0.42 | 0.9918864 | 0.9244936 | 0.5060241 |
| GSCI | 18 | 31 | 11 | 483 | 0.62069 | 0.367347 | 0.9858012 | 0.9226519 | 0.4615385 |
| GSPO | 46 | 4 | 4 | 489 | 0.92 | 0.92 | 0.9756593 | 0.985267 | 0.92 |
| GDEF | 8 | 42 | 7 | 486 | 0.533333 | 0.16 | 0.9959514 | 0.9097606 | 0.2461538 |

5.4 Comparison

While Kosmix and Reuters results are very similar, our native classifier has achieved slightly better scores concerning F-Measure results. Based on Precision results, Kosmix Classification scored better on 4 topics and Reuters scored better on 11 topics. Based on Recall results, Kosmix Classification and Reuters Classification are at a draw with 7 each. Based on True Negative results Kosmix Classification scored better on 4 topics and Reuters scored better on 11 topics. Based on Accuracy results, Kosmix Classification scored better on 3 topics and Reuters scored better on 12 topics. Finally based on F-Measure results, Kosmix Classification scored better on 5 topics and Reuters scored better on 10 topics.

CHAPTER 6

CONCLUSIONS & FUTURE WORK

6.1 Conclusions

Our aim was to compare taxonomy based categorization search engine Kosmix with a very basic classifier such as Vector Space Model and Cosine Similarity. At the end accuracy and true negative scores were really high meaning both systems are capable of determining what is not the topic of given file. On the other hand with mediocre F-Measure scores both systems struggle to make an accurate positive suggestion.

6.2 Future Work

Suggestion algorithm may be switched into number of actual topics. After stemming Google search can be used to correct typo problems (i.e: mediterranean, mediterreanean). Instead of selecting Top 5 documents Top n documents can be used to experiment with the results. Instead of calculating the total cosine similarity score for topics, number of occurrences can be used. Training set can be expanded for better topic representation

REFERENCES

- [1] **Fabrizio Sebastiani.** (2005), Text categorization. In Alessandro Zanasi (ed.), *Text Mining and its Applications*, WIT Press, Southampton, UK, 2005, pp. 109-129.
- [2] **KAREL FUKA, RUDOLF HANKA** (2007) *Feature Set Reduction for Document Classification Problems* pp.1-7.
- [3] **C. H. A. Koster** (2003), University of Nijmegen, The Netherland, <http://www.cs.ru.nl/~kees/ir2/> [2003]
- [4] **GERSTL, P., HERTWECK, M., KUHN, B.** (2004) Text Mining: *Grundlagen, Verfahren und Anwendungen*, in: *Praxis der Wirtschaftsinformatik-Business Intelligence*, Vol. 39, No. 222, pp. 38-48.
- [5] **HEIDE BRUCHER, GERHARD KNOLMAYER, MARC-ANDRE MITTERMAYER** (2002) *Document Classification Methods for Organizing Explicit Knowledge*
- [6] **DOMINGOS, P., PAZZANI, M.** (1997) *On the Optimality of the Simple Bayesian Classifier under Zero-One Loss*, in: *Machine Learning*, Vol. 29, No. 2-3, pp. 103-130.
- [7] **Lam, W., Low, K. F., Ho, C.Y** (1997) Using a Bayesian Network Induction Approach for Text Categorization, in: *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pp. 745-750

[8] **WITTEN, I. H., FRANK, E.** (2011) *Data Mining - Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann Publishers: San Francisco.

[9] **HERMANN, K. Rakesh Agrawal:** (2002), Athena: *Mining-based Interactive Management of Text Databases*, <http://www3.informatik.tu-muenchen.de/lehre/WS2001/HSEM-bayer/textmining.pdf> [as of 02-03-2002]

[10] **NG, H. T., GOH, W. B., LOW, K. L.** (2010) *Feature Selection, Perceptron Learning, and a Usability Case Study for Text Categorization*, in: Proceedings of the 20th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp. 67-73.

[11] **RUIZ, M. E., SRINAVASAN, P** (1998) *Automatic Text Categorization Using Neural Network*, in: Proceedings of the 8th ASIS SIG/CR Workshop on Classification Research, pp. 59-72.

[12] **KRAHL, D., WINDHEUSER, U., Zick, F.-K** (2004) *Data Mining - Einsatz in der Praxis*, Addison Wesley Longman: Bonn.

[13] **NIE, J. H.** (1995) *Constructing Fuzzy Model by Self-Organizing Counterpropagation Network*, in: IEEE Transactions on Systems, Man and Cybernetics, Volume 25, No. 6, pp. 963-970.

[14] **APTE, C., DAMAREAU, F., WEISS, S. M** (2004) *Towards Language Independent Automated Learning of Text Categorization Models*, in: Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp. 23-30.

[15] **MUHAMMED MIAH** (2009) *Improved k-NN Algorithm for Text Classification* pp. 1-7

[16] **YANG, Y., CHUTE, C** (2008) *An Example-Based Mapping Method for Text Categorization and Retrieval*, in: ACM Transactions on Information Systems, Vol. 12, No. 3, pp. 253-277.

[17] **MADANI, O.** (2001), *ABCs of Text Categorization*, http://classes.seattleu.edu/computer_science/csse470/Madani/ABCs.html [04-24-2001].

[18] <http://www.crunchbase.com/company/kosmix>

[19] **ANAND RAJARAMAN** (2009) *Kosmix: Exploring the Deep Web using Taxonomies and Categorization*, Kosmix Corporation, Mountain View, CA, USA

[20] <http://en.wikipedia.org/wiki/Stemming> [as of 01-02-2011]

[21] **Salton, Gerard.** (1983) , *Introduction to Modern Information Retrieval*. McGraw-Hill.

[22] **Makhoul, John; Francis Kubala; Richard Schwartz; Ralph Weischedel,** (1999), *Performance measures for information extraction*. In: Proceedings of DARPA Broadcast News Workshop, Herndon, VA.

[23] <http://www.miislita.com/term-vector/term-vector-3.html> [01-02-2011]

APPENDIX A

Stop Words

| | | | | | | | |
|-------------|---------|-----------|----------|------------|----------|---------|----------|
| a | ani | begin | diifer | everyon | gotten | id | know |
| abl | announc | behind | do | everyth | h | ie | known |
| about | anoth | believ | doe | everywher | ha | if | l |
| abov | anybodi | below | doesn | ex | had | I'll | larg |
| abst | anyhow | besid | doesn't | except | happen | im | last |
| accord | anymor | between | don | f | hardli | immedi | late |
| accordingli | anyon | beyond | done | far | hasn | import | later |
| across | anyth | biol | don't | few | hasn't | in | latter |
| act | anywai | both | down | ff | have | inc | latterli |
| actual | anywher | brief | downward | fifth | haven | inde | least |
| ad | appear | briefli | due | first | haven't | index | less |
| adj | ar | but | dure | five | he | inform | lest |
| adopt | aren | by | e | fix | hed | instead | let |
| affect | arent | c | each | follow | henc | into | like |
| after | aris | ca | ed | for | her | invent | line |
| afterward | around | came | edu | former | here | inward | littl |
| again | as | can | effect | formerli | hereaft | is | ll |
| against | asid | cannot | eg | forth | herebi | isn | look |
| ah | ask | can't | eight | found | herein | isn't | ltd |
| al | at | caus | eighti | four | hereupon | it | m |
| all | auth | certain | either | from | herself | itd | made |
| almost | avail | certainli | els | further | hi | it'll | mai |
| alon | awai | co | elswher | furthermor | hid | itself | mainli |
| along | awfulli | com | end | g | him | I've | make |
| alreadi | b | come | enough | gave | himself | j | mani |
| also | back | contain | especi | get | hither | just | mayb |
| although | be | could | et | give | home | k | me |
| alwai | becam | couldn't | et-al | given | how | keep | mean |
| am | becaus | d | etc | go | howbeit | kei | meantim |
| among | becom | date | even | goe | howev | kept | meanwhil |
| amongst | been | did | ever | gone | hundr | kg | mere |
| an | befor | didn | everi | got | i | km | mg |

| | | | | | | | |
|-------------|-------------|---------------|------------|---------------|------------|------------|----------|
| might | nobodi | our | q | self | specif | thenc | togeth |
| million | non | ourselv | que | sent | state | there | too |
| miss | none | out | quickli | seven | still | thereaft | took |
| ml | nonetheless | outsid | quit | sever | stop | therebi | toward |
| more | noon | over | qv | shall | stopword | therefor | tri |
| moreov | nor | overal | r | she | strongli | therein | truli |
| most | normal | ow | ran | shed | sub | there'l | try |
| mostli | not | own | rather | she'll | substanti | thereof | ts |
| mr | note | p | rd | should | succesfuli | therer | twice |
| much | noth | page | re | shouldn | such | thereto | two |
| mug | now | part | readili | shouldn't | suffici | thereupon | u |
| must | nowher | particular | realli | show | suggest | there'v | un |
| my | o | particularli | recent | shown | sup | these | under |
| myself | obtain | past | ref | signific | sure | theyd | unfortun |
| n | obvious | per | regard | significantli | t | they'll | unless |
| na | of | perhap | regardless | similar | take | theyr | unlik |
| nai | off | place | rel | similarli | taken | they'v | until |
| name | often | pleas | relat | sinc | tell | thi | unto |
| nd | oh | plu | research | six | tend | think | up |
| near | ok | poorli | respect | slightli | th | those | upon |
| nearli | okai | possibli | result | so | than | thou | us |
| necessari | old | potenti | right | some | thank | though | usefulli |
| necessarili | omit | pp | run | somebodi | thanx | thoughh | usual |
| need | on | predominantli | s | somehow | that | thousand | v |
| neither | onc | present | sai | someon | that'll | throug | valu |
| never | onli | previous | said | someth | that'v | through | variou |
| nvertheless | onto | primarili | same | somethan | the | throughout | ve |
| new | or | probabl | saw | sometim | thei | thru | veri |
| next | ord | promptli | sec | somewhat | their | thu | via |
| nine | other | proud | section | somewher | them | til | viz |
| nineti | otherwis | provid | see | soon | themselv | tip | vol |
| no | ought | put | seem | sorri | then | to | vs |
| w | wa | wai | want | wasn | we | wed | welcom |
| went | where | weren | whenev | who | whod | whole | which |
| x | y | ye | yet | you | youd | you'll | your |
| yourself | yourself | you'v | z | | | | |

APPENDIX B

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Umut, Ergün

Nationality: Turkish (TC)

Date and Place of Birth: 23 October 1979 , İstanbul

Marital Status: Single

Phone: +90 532 240 76 26

mail: celefin@gmail.com

EDUCATION

| Degree | Institution | Year of Graduation |
|-------------|--|--------------------|
| MS | Çankaya Univ. Computer Engineering | 2011 |
| BS | Eastern Mediterranean Univ. Information Technology | 2006 |
| High School | Yüce Fen Lisesi Ankara | 1996 |

WORK EXPERIENCE

| Year | Place | Enrollment |
|--------------|----------------|-------------------|
| 2010-Present | Softtech | Software Analyst |
| 2009-2010 | C.E Technology | Software Engineer |
| 2009 | Ecalibra | Software Engineer |
| 2008 | Sentim | Software Engineer |
| 2006-2008 | PDI-Erkom | Software Engineer |

FOREIGN LANGUAGES

Advanced English, Basic German