Automatic Email Organization

2008 / 2009

Ludimila Luiza de Lima Gabriel

Automatic Email Organization

Ludimila Luiza de Lima Gabriel

Supervisor Nuno Escudeiro

Co-supervisor Walter Daelemans

MESTRADO EM ENGENHARIA INFORMÁTICA

Tecnologias do Conhecimento e Decisão

2008 / 2009

September 2009

«To those who selflessly and lovingly made this possible:
my beloved family »

Acknowledgements

I thank the Director of the Master program in Computer Science Engineering at School of Engineering (Porto Polytechnic Institute), **Dr. Fátima Rodrigues** for her prompt help whenever needed.

To the Director of the GECAD (Grupo de Investigação em Engenharia do Conhecimento e Apoio à Decisão) **Dr. Carlos Ramos** for all logistic, support and advice granted to me during the two years that I worked in this institution.

To the Portuguese Foundation for Science and Technology (MES-FCT) for funding me with a the two-year research grant under the projects EDGAR (POSI/EIA/61307/2004) and Coalesce (PTDC/EIA/74417/2006).

I would like to thank **Dr. Nuno Silva**, who kindly accepted me as a fellowship researcher almost three years ago, who introduced the researching environment to me and with whom I learned the meaning of words like work dedication and commitment.

To the director of the CNTS (Centrum voor Nederlandse Taal en Spraak), **Dr. Walter Daelemans**, my co-advisor, for welcoming me as a master student and for integrating me in his team, it was an extra motivation during my project.

I would like to thank the **Associação Portuguesa Para a Inteligência Artificial (APPIA)** that collaborated in this project providing the entire dataset source.

Many thanks to my supervisor **Nuno Escudeiro**, who gently accepted me as his master student and believed on my capabilities; I'm thankful for his guidance, help and support.

To **Anita Bernard** and **Wim De Bruyn** for making believe that life has always an open door and the world is full of new opportunities.

To the **Soares Vasco** family for the support during the very beginning of my academic life, I would like to thank Ana to show me that I should believe in myself, Fernando for his positive and careful attitude, Marco for his great sense of humor and João for teaching the interesting side of a good discussion.

I would like to thank my **friends** whom I was lucky to encounter during the last 7 years at **ISEP:** André Silva, Danielson Alves, Ivo Pereira, Fátima Ribeiro, Raquel Martins and Paulo Maio; each of them touched in a special way, I'm very thankful for their friendship. To the **employees** of **ISEP**: Alexandra Madureira for her incredible help with bureaucratic issues and Security Dias for his politeness and logistics support.

To my friends **David Roheim**, **Malka DiMartino**, **Myriam Levy** and **Yoel Engenstein** who laughed and shared with me great experiences and moments. To **Matina Volou** for her cheerfulness and friendliness during last summer.

To **Miriam Zeltzer**, **Baruch Zeltzer** and **Assaf Zeltzer** for making me feel home.

To **Jorge Santos** for his wise advice and his ability to show and look at life from a different perspective.

I am especially thankful to my dear friend **Hélio Martins** for pushing me through the difficult times. Working with him was professionally and personally essential to the development of this project; his help at many different moments over the last years were great. *"Bô e kool".*

Many thanks to **Nir Zeltzer** for being with me every day, making me happy and being my safe harbor, helping me during the hard moments and celebrating the good ones. His wonderful support and motivation were very important to keep me going forward with a great smile. To **Doron Zeltzer** who makes my weekends very special, his lovely laugh always inspired me.

I would like to thank my parents and siblings for their endless love. To my "little" brother **Felipe** for his great sense of humor and friendship. To my precious sister and friend **Daniela** for being always with me, her will to live is amazing and it always motivates me; I would like to thank her for listening, understanding and supporting me. I would like to thank my **mom** who taught me to fight for what I want, to keep positive and to believe that the future is bright. To my **dad** for everything he did for me during all these years, his efforts and unconditional support were essential to prepare me for the journey of life. I am incredible blessed to have their love not only during this project but also throughout my life.

Resumo Alargado

Introdução

Actualmente, as mensagens electrónicas são consideradas um importante meio de comunicação. As mensagens electrónicas – vulgarmente conhecidas como emails – são utilizadas fácil e frequentemente para enviar e receber o mais variado tipo de informação. O seu uso tem diversos fins gerando diariamente um grande número de mensagens e, consequentemente um enorme volume de informação.

Este grande volume de informação requer uma constante manipulação das mensagens de forma a manter o conjunto organizado. Tipicamente esta manipulação consiste em organizar as mensagens numa taxonomia. A taxonomia adoptada reflecte os interesses e as preferências particulares do utilizador.

Motivação

A organização manual de emails é uma actividade morosa e que consome tempo. A optimização deste processo através da implementação de um método automático, tende a melhorar a satisfação do utilizador.

Cada vez mais existe a necessidade de encontrar novas soluções para a manipulação de conteúdo digital poupando esforços e custos ao utilizador; esta necessidade, concretamente no âmbito da manipulação de emails, motivou a realização deste trabalho.

Hipótese

O objectivo principal deste projecto consiste em permitir a organização ad-hoc de emails com um esforço reduzido por parte do utilizador. A metodologia proposta visa organizar os emails num conjunto de categorias, disjuntas, que reflectem as preferências do utilizador. A principal finalidade deste processo é produzir uma organização onde as mensagens sejam classificadas em classes apropriadas requerendo o mínimo número esforço possível por parte do utilizador.

Para alcançar os objectivos estipulados, este projecto recorre a técnicas de mineração de texto, em especial categorização automática de texto, e aprendizagem activa. Para reduzir a necessidade de inquirir o utilizador – para etiquetar exemplos de acordo com as categorias desejadas – foi utilizado o algoritmo d-confidence.

Processo de organização automática de emails

O processo de organizar automaticamente emails é desenvolvido em três fases distintas: indexação, classificação e avaliação.

Na primeira fase, fase de indexação, os emails passam por um processo transformativo de limpeza que visa essencialmente gerar uma representação dos emails adequada ao processamento automático.

A segunda fase é a fase de classificação. Esta fase recorre ao conjunto de dados resultantes da fase anterior para produzir um modelo de classificação, aplicando-o posteriormente a novos emails. Partindo de uma matriz onde são representados emails, termos e os seus respectivos pesos, e um conjunto de exemplos classificados manualmente, um classificador é gerado a partir de um processo de aprendizagem. O classificador obtido é então aplicado ao conjunto de emails e a classificação de todos os emails é alcançada. O processo de classificação é feito com base num classificador de máquinas de vectores de suporte recorrendo ao algoritmo de aprendizagem activa d-confidence.

O algoritmo d-confidence tem como objectivo propor ao utilizador os exemplos mais significativos para etiquetagem. Ao identificar os emails com informação mais relevante para o processo de aprendizagem, diminui-se o número de iterações e consequentemente o esforço exigido por parte dos utilizadores.

A terceira e última fase é a fase de avaliação. Nesta fase a performance do processo de classificação e a eficiência do algoritmo d-confidence são avaliadas. O método de avaliação adoptado é o método de validação cruzada denominado *10-fold cross validation*.

Conclusões

O processo de organização automática de emails foi desenvolvido com sucesso, a performance do classificador gerado e do algoritmo d-confidence foi relativamente boa. Em média as categorias apresentam taxas de erro relativamente baixas, a não ser as classes mais genéricas.

O esforço exigido pelo utilizador foi reduzido, já que com a utilização do algoritmo d-confidence obteve-se uma taxa de erro próxima do valor final, mesmo com um número de casos etiquetados abaixo daquele que é requerido por um método supervisionado.

É importante salientar, que além do processo automático de organização de emails, este projecto foi uma excelente oportunidade para adquirir conhecimento consistente sobre mineração de texto e sobre os processos de classificação automática e recuperação de informação. O estudo de áreas tão interessantes despertou novos interesses que consistem em verdadeiros desafios futuros.

Palavras-chaves

Mineração de texto, recuperação de informação, categorização automática de textos, categorização automática de emails.

Abstract

Nowadays, emails are one of the most important information sources and ways to communicate. Emails are used for many and varied purposes generating a high amount of information. This high amount of information requires a manipulation of emails intended to organize emails on a specific taxonomy that makes it easier to find a particular message or a thread of messages on some subject.

Manual emails categorization is a time consuming task. Users' satisfaction when using an email client might increase if the organization of emails according to user specific interests is available at a reduced effort. The main goal of this project is to develop a solution to organize emails automatically requiring reduced efforts from the users. This organization reflects user's own interests aiming to satisfy each and every personal interest.

This document describes systematically and deeply the approach developed in this project for automatic email categorization which is based on text categorization using support vector machines and active learning.

The methodology adopted reached good results; generally each category presents a low error rate at a reduced workload, with few exceptions at categories representing general soften concepts. The users' effort was reduced using d-confidence algorithm.

Keywords: Text mining, information retrieval; active learning; text categorization; emails categorization.

Table of Contents

List of Figures

List of Tables

Notations

API	Application Programming Interface
APPIA	Associação Portuguesa Para a Inteligência Artificial
CNTS	Centrum voor Nederlandse Taal en Spraak
Dtm	Document-term matrix
HTML	Hypertext Markup Language
IR	Information Retrieval
ML	Machine Learning
SVM	Support Vector Machine
TFxIDF	Term Frequency - Inverse Document Frequency
VSM	Vector Support Model
WWW	World Wide Web

Chapter 1

Introduction

Emails are today one of the most important tools supporting interpersonal communication. Emails are used for private, institutional, professional and many other purposes generating a high volume of messages received every day. This poses new problems to users that have to put significant efforts on email organization. To manage emails, users frequently organize them in folders referring to subjects of particular personal relevance. This procedure intends to organize emails on a specific set of classes mapping user personal interests during a given time frame.

Manual emails categorization is a time consuming task; any improvements on the way users execute this task might increase users' satisfaction and improve efficiency. A potential solution for this problem is to use text categorization methods that automate the organization process.

Current email systems usually offer some functionality that intends to assist users in managing large quantities of email messages. However, these approaches are focused on the accuracy of the email categorization process. Accuracy is a fundamental characteristic of an email categorization system but it is not the only issue; our main focus is on the reduction

of user effort, another relevant issue in managing voluminous corpora (in the text mining field the term *corpus*, *corpora* in plural, refers to collections of text documents), while keeping acceptable accuracy.

The hypothesis we wish to evaluate is that it is possible to organize email messages accurately according to a set of classes describing user personal interests with a little effort from the user and without requiring any specific technical knowledge in order to describe user interests.

This proposal is based on text mining techniques and active learning. It applies d-confidence, an active learning algorithm that is particularly focused on the fast description of unknown classes, trying to learn a concept at low cost, i.e., while posing few queries to the user.

The experimental results we have achieved led us to conclude that is possible to organize emails automatically with a good accuracy rate at a reduced cost.

This remainder of this document is organized as follow:

Chapter 2 – below, presents an overview of text and email organization, specifying each phase of this process: indexing, classification and evaluation. It also introduces concepts of machine learning and text mining.

Chapter 3 – Indexing Text Documents, presents the indexing process of text documents and, mainly, emails indexing. It describes relevant steps as pre-processing and document representation.

Chapter 4 – Classification Process, presents the classification process as how a classifier is generated and then applied to a dataset. It also describes the d-confidence algorithm in detail.

Chapter 5 – Automatic Email Organization Prototype, describes the prototype developed based on the methodology adopted for email organization. It specifies the approach for the prototype and the respective architecture.

Chapter 6 – Automatic Email Organization Process Evaluation, presents and discusses the evaluation results of the methodology adopted, including the performance of the d-confidence algorithm and the classification process.

Chapter 7 – Conclusion, summarizes the work done, and refers to future work and final remarks.

This project was developed as part of the two years master degree program in Computer Science Engineering, area Knowledge Technologies and Decision at the School of Engineering – Polytechnic of Porto, Portugal under the supervision of Professor Nuno Escudeiro.

Chapter 2

Text Categorization

This chapter describes the state-of-art on automatic text categorization. It presents different concepts and processes used during automatic text classification including an overview about machine learning, specifically semi-supervised learning and active learning, and a description of how the classification process is accomplished and the relevant aspects that arise when the target corpus is a set of emails.

2.1 Machine Learning

Text documents categorization, in the context of emails categorization, is mainly a classification task [Sebastiani, 2002]. Text categorization defines one or more categories to text documents based on their content.

It is common to refer to text classification as text categorization, document classification, document organization or topic spotting [Sebastiani, 2002] and to categories as classes, labels or topics.

Nowadays, the principal approach for automatic categorization of text is based on machine learning [Sebastiani, 2002].

Machine learning (ML) is concerned with developing analytical models that explain data [Buitelaar, Paul; Cimiano, Phillipp, 2008], in other words, it is concerned with finding algorithms and techniques to allow computers to learn from examples that are fed as an input. These techniques can be categorized mainly as supervised, unsupervised and semi-supervised learning. Active learn is a particularly efficient approach of iterative supervised learning that we have used in this work.

Typically, machine learning is divided in three areas: clustering, classification and regression. *Clustering* is an unsupervised learning technique to generate homogeneous groups of input examples called *clusters* [Pantel, Patrick; Pennacchiotti, Marco;, 2006]. *Classification* is a learning technique, where pre-defined classes are assigned to instances bases on quantitative information. Classification may be operated on supervised, semi-supervised or active learning settings. *Regression* may be considered similar to classification but it predicts continuous dependent variables.

As referred before, this project uses classification techniques to solve a categorization problem. It is also relevant to refer that text categorization belongs to the text mining area. “Text mining is a burgeoning new field that attempts to glean meaningful information from natural language text” [Witten, 2004].

The following sub-sections present in more details the different phases of the learning process in the classification domain.

2.1.1 Supervised, Unsupervised and Semi-supervised learning

Text Organization is a learning process where training instances can be labeled or not. The learning process is classified as supervised, unsupervised or semi-supervised depending on the presence or absence of pre-labeled cases in the training set (*Figure 1*).

S**upervised learning** is based in a learning process supervised by known categories and trained instances [Sebastiani, 2002]. It defines classifiers using pre-labeled examples. Only labeled examples are used for training. This setting requires a set of examples of all classes to be labeled in advance.

The **unsupervised learning** setting, a.k.a. clustering, is a learning process based only on unlabeled examples. Classes are not known in advance. Does not require any labeling effort from the users.

Semi-supervised learning combines labeled and unlabeled data for training [Zhu, et al., 2009], providing algorithms that take advantage of this combination. Since it does not require all examples to be pre-labeled it reduces the time required to label data; on the other hand it leverages the information on unlabeled cases solving the problem of scarce labeled data.

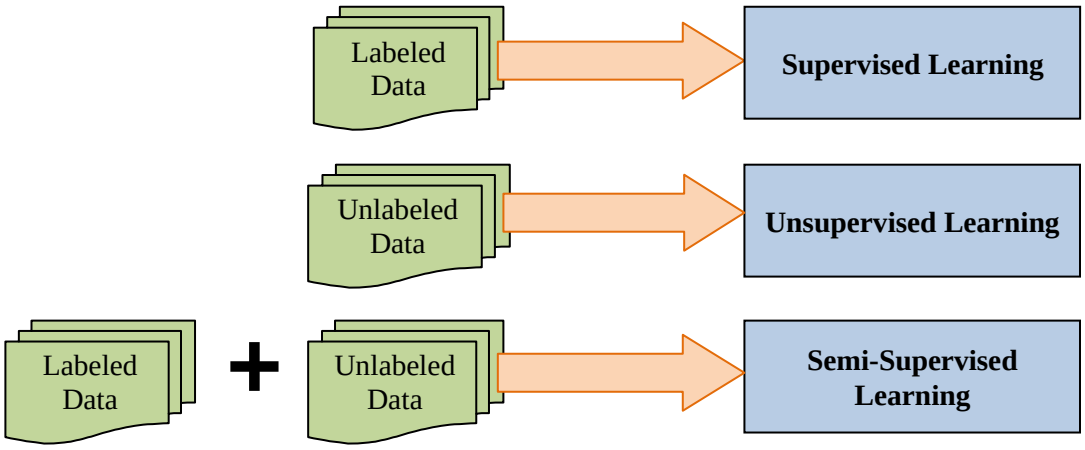

Figure 1 – Supervised, unsupervised and semi-supervised learning

There is no absolute perfect approach but, for each situation, the one that is the most appropriated according to specific goals and circumstances. In this project, labeled and unlabeled examples are used, an active learning solution is adopted where only a few pre-labeled emails are required. Then, with these labeled emails, a learner is generated and

applied to the unlabeled emails to select the most informative unlabeled example asking the user for its label. This process is repeated iteratively until an acceptable model is learned.

2.1.2 Active Learning

The active learning approach is based in a learning process where examples to label are criteriously selected by the active learning algorithm. It detects and asks the user to label the most informative examples in instance space. These requests made by active learners are called *queries*. Active learning allows generating personal classification, since it reflects users' preferences [Muslea, et al., 2006].

In general, active learning significantly reduces the numbers of queries improving the classification effectiveness; essentially because by filtering the queries it uses the most informative examples and not all randomly selected examples, thus minimizing the human labeling effort [David A., et al., 1996]. Since the learner chooses the examples and it can actively ask for labels, it is a good solution for large sets of data when labeling is an expensive and demanding process.

As referred above, this project is based on an active learning approach, to define which cases should be labeled in advance. The d-confidence algorithm [Escudeiro, et al., 2008] is used. D-confidence is described in 4.3.

2.2 Text Classification

Text classification is based mainly on machine learning techniques; usually a learner builds a classifier for different categories based on a set of labeled documents. It is formally [Sebastiani, 2002] defined as:

$$\langle d_j, c_i \rangle \in D \times C$$

- $D = \{d_1, d_2, \dots, d_{|D|}\}$ is a set of documents.
- $C = \{C_1, C_2, \dots, C_{|C|}\}$ is a set of predefined categories.

The task is to approximate the target function Φ':

$$\Phi' = D \times C \rightarrow \{T, F\}$$

To the classification function (or classifier) Φ:

$$\Phi = D \times C \rightarrow \{T, F\}$$

- T is a value assigned to $\langle d_j, c_i \rangle$ to indicate a decision that document d_j is under c_i.
- F is a value assigned to $\langle d_j, c_i \rangle$ to indicate a decision that document d_j is not under c_i.

The effectiveness of a classification process is defined based on these values; it is related to the classifier ability to predict the right class.

The text classification process (*Figure 2*) is based on three complex phases [Giorgetti, et al., 2003], [Sebastiani, 2002]: document indexing, classifier learning and evaluation.

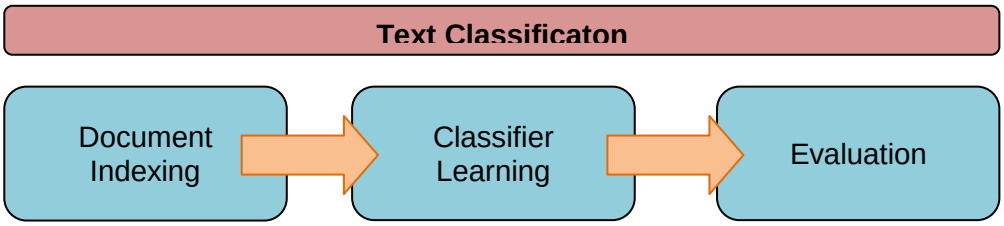

Figure 2 – Text Classification

Each of these phases is presented in a different chapter during this project.

2.3 Text Classification Terminology and Notation

Typically, text classification generates classifiers based on an inductive process (learner) and on a set of pre-classified documents. This classifier will then be used to categorize new documents. A formal definition of text classification is described on section 2.2 but a specific terminology [Muslea, 2002] is used in this project and described in this section.

Instance space is the set of all documents, represented as D. **Example** or **instance** is an email (or document) from the instance space, represented as d.

Target function refers to the classifier that always predicts the correct class of d. It is represented as Φ'. **Classification function** or **classifier** is an hypothesis function, represented as Φ.

Training example is an example used during the learning process. This example can be labeled or unlabeled according to the kind of learning applied (learning classification is described on sections2.1.22.1.1 and 2.1.4). If it considers labeled examples, these examples are represented as $\langle \mathrm{d}, \mathrm{c}(\mathrm{d}) \rangle$, where d is the classified (or labeled example) and c is the class to which d belongs.

Training set is a set of training examples.

2.4 Email Categorization

Nowadays, emails are part of our daily lives especially as a very useful communication tool. Emails are used for many different purposes and they manipulate huge amounts of data and information. To better understand and use large collections of emails, users organize them in folders. This project aims to develop a methodology for automatic email organization.

2.4.1 Email as a corpus

This section presents an overview about email organization with some known methods to organize them.

One of the common methods to organize emails is to manually archive messages into folders [Ayodele, et al., 2007], though this method does not reflect necessarily a true classification since folders names do not reflect always their content. This method is also time consuming and rely on the user.

Nevertheless, semi-automatic classification under specific folders is a frequent approach. Some examples of semi-automatic email organization into folders are:

- Email clients as Mozilla Thunderbird and Microsoft Outlook which work according to rule sets defined by a user;

- IBM's MailCat [Segal, et al., 2002] that it is based on users' mail-filling habits providing a list with the most appropriate folders (classification) for new email messages and
- Magi [Payne, et al., 1997] that it is based also on users' interaction and it uses a learning algorithm to classify new messages.

Another approach to classify emails is using social network [Stolfo, et al., 2004] analysis, where it tracks and analyzes users' hits and behavior.

Email categorization is useful also to define a prioritization for emails based on which content is important for the user. Another approach to assess incoming emails is making recommendations before emails reach the user's inbox, classifying them according to their priority, i.e. high or low importance.

[Ayodele, et al., 2007] propose a solution where emails are grouped according to different activities, in this approach emails are grouped by extracting most frequent words in the content of emails and comparing these words with frequent words from emails under a specific class.

A rule-based system can classify emails semi-automatically to their respective folders but it can be complicated when it requires users expertise to develop the right rules. One solution for this approach is to develop graphical user interfaces to allow and to facilitate users' interaction [Helfman, et al., 1995].

Email categorization can be useful not only to assign messages to user-created folders but also to filter SPAM. [Klimt, et al., 2004]. SPAM (also known as junk emails) are emails unsolicited by the recipient, sent to many recipients and often of a commercial nature. Actually, in most of the cases, SPAM classification uses heuristics [Meyer, et al., 2004] and in few cases it is based on statistical methods from test classification [Kolcz, et al., 2004].

This project uses an active learning algorithm to classify emails. The methodology proposed in this work is described in the next section.

2.4.2 Email Organization

The methodology to automatically organize emails is represented on *Figure 3*. Starting from a set of emails and a set of user-defined categories – and using text mining

techniques for classification in an active learning setting – it generates an automatic email organization process. This process is explained in details throughout the next chapters.

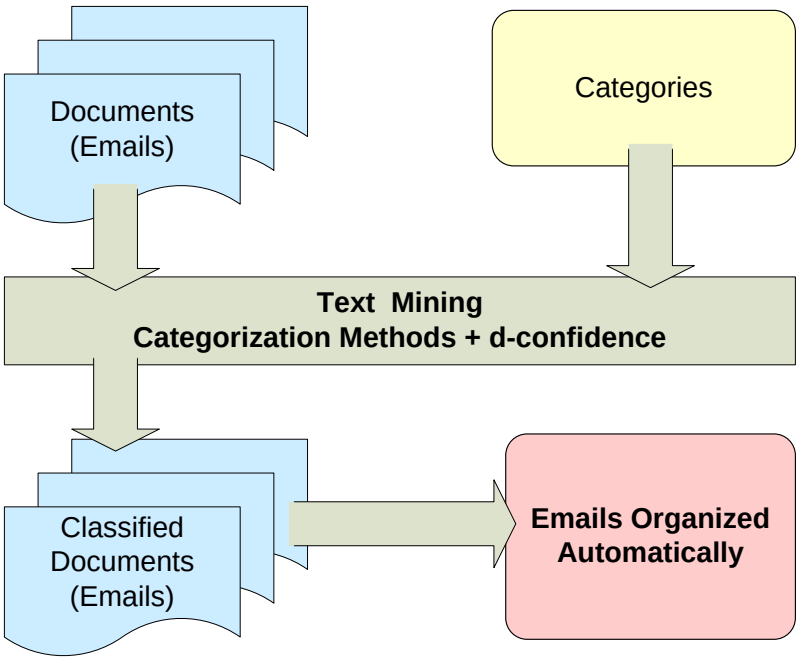

Figure 3 – Automatic Email Categorization

The methodology to automatically organize emails is based on the three phases referred above: document indexing, classifier learning and evaluation. *Figure 4* represents these phases.

The first phase, *pre-processing*, transforms the source dataset, cleaning and extracting only relevant information from emails. The second phase, called *learning*, generates a classification model, or classifier, based on categories describing user interests and a set of exemplary emails; in this phase the classifier is also applied to non-labeled emails to predict their categories. The last phase, *evaluation*, evaluates the whole classification process using, in this work,a cross-validation method.

Since users are required to label exemplary cases, the learning process is explicitly conducted towards user personal interests, being developed under an active learning approach. To minimize the number of required labels, i.e. queries to the user, the d-confidence algorithm is applied.

The classifier is a crucial part for the classification process but the support given by the d-confidence improves significantly the time-consuming labeling task. It increases the user satisfaction and effectiveness of the organization process.

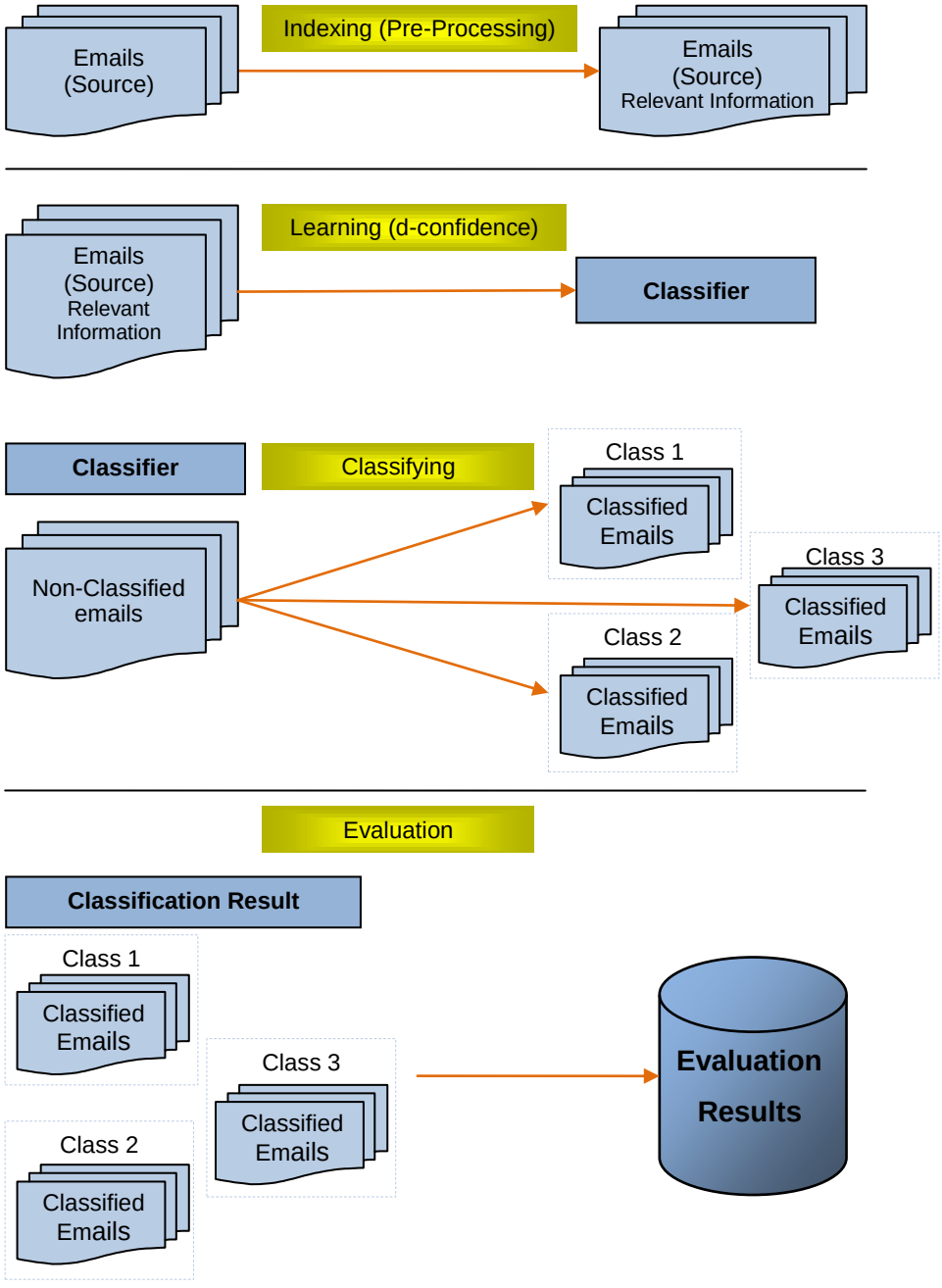

Figure 4 – Email Categorization

After an automatic classifying process, emails are classified into a specific taxonomy. This taxonomy is composed of different categories describing user specific interests. Further on, these categories can be presented to users in several ways; for example: different categories can be presented as folders.

2.5 Conclusion

This chapter presented an overview on text and, specifically, emails organization.

Typically an organization problem, in this context, is mainly a text categorization problem, which requires the specification of a set of categories in advance. These categories are defined by users or else they are proposed by the classification system.

In this project the categories are defined by the user who has the responsibility of identifying examples describing them. Hence, an initial set of exemplary emails is classified (or labeled) manually defining an input for an initial learner that is iteratively refined.

Chapter 3

Indexing Text Documents

Indexing documents is the first phase of a text classification process; this phase is extremely relevant since it prepares the dataset to the next phases. It transforms documents through pre-processing tasks extracting relevant information and representing them in a more appropriated format; however at the same time this process should not be too aggressive to avoid loss of data. It is important to remind that, in the context of this project, we refer to emails as documents.

This chapter describes the indexing process, clarifying pre-processing and document representation tasks. The last sub-section explains the specific case of emails indexing.

3.1 Indexing

In this phase different text pre-processing techniques are used as: tokenization, terms normalization, extraction of stop words, stemming and reduction of the number of terms. The main goal is to transform documents, to a format which is suitable to machine learning processing. It reduces the number of terms, selects the most relevant information and organizes that data according to a model that is adequate for automatic computer processing.

3.2 Pre-Processing

3.2.1 Tokenization

Tokenization is directly related with lexical analysis. Lexical analysis, in computer science, is the process of transforming a sequence of text in a sequence of tokens. It aims the conversion of a given input, for example a sentence, to manageable components. “Tokenization splits text into simple tokens, such as numbers, punctuation, symbols and words of different types, adding a “Token” annotation to each” [Maynard, Diana; Li, Yaoyong; Peters, Wim;, 2008]. It allows the representation of text as a list of classified tokens.

This process is very important for text pre-processing since it reduces the complexity of the data set; however, it should be also a careful process because the complexity reduction can generate also loss of information if too aggressive.

Ian Witten e Frank Eibe [Witten, et al., 2005 p. 310] present one example that show how this process is not as simple as it looks: considering cases with words like *doesn’t* and *aren’t* a decision need to be taken or they will be considered as different terms when compared with *does not* or *are not;* and actually they have the same semantic meaning.

3.2.2 Normalization

Usually in text categorization, during the tokenization process, a normalization process is also applied to document’s content. This normalization (Table 1) aims to recognize terms written in different ways as equivalents.

Table 1 – Normalization

Terms	**Terms after normalization**
Vice-versa	viceversa
U.K.	uk
UK	uk
Portuguese	portuguese
Organization	organisation
Cup	mug

A normalization process removes irrelevant characters as white space, punctuation and numbers, and/or it transforms the entire content to lower case letters and/or also by using mapping rules it defines equivalents terms. English language is a good example where mapping rules are applied to normalize same terms with American spelling and British spelling, for example: “colour” and “color”, “labelled” and “labeled”.

3.2.3 Stop words

Typically, frequent terms occurring many times but without added value for classification purposes are removed. These terms are referred to as *stop words* (Table 2) and they can be articles, prepositions and any other semantically irrelevant words (at least at the majority of times).

Table 2 – Stop Words

Examples of Stop Words (English)		
of	only	such
the	there	them

The process of removing stop words is based on a list named *stop list*. This list contains all the words to be removed (stop words). Of course, stop lists are directly related to a specific language.

In this project because the source dataset is from a Portuguese association but the content is mainly expressed in English, hereby an English stop list is used.

3.2.4 Stemming

Another very common transformation is *stemming*; it normalizes terms reducing them to the same stem. For example (Table 3):

Table 3 – Stemming

Term	Stem
origin	origin
original	origin
originality	origin
originally	origin

This process reduces terms to the same stem removing suffixes or/and prefixes of a word; it generates words stems which can be or not the same linguistic stem [Spark-Jones, et al., 1997], [Chaves, 2003].

3.2.5 Dimensionality Reduction

Usually the number of terms in a text corpus is quite big; however many of these terms are not relevant for the classification process. A big dimensionality leads to computation problems and it increases the difficulty on detecting relations among them; besides, terms can be useless for distinguishing characteristics of documents.

The dimensionality reduction is frequently based on the term frequency value. The term frequency value is the number of times that a term occurs in a document; this value is calculated after removing stop words [Mladenic, et al., 1999]. After calculating term frequencies, all terms under a certain threshold are removed. A low frequency of some terms means also that these terms are not relevant to predict a class and they can affect the effectiveness of the classification process.

All processes where terms are removed can induce a significative loss of data if too aggressive. In the case of reducing the number of terms, it is crucial to find an adequate threshold that determines what information is to be removed.

3.3 Document Representation

Typically, text documents are represented by a vector space model (VSM) [Salton, et al., 1975].

In a VSM each vector element refers to a term and has a value associate to it. Using document indexing methods, documents are represented as a vector of term weights; in this vector each term has a weight according to a specific document. Document-terms matrices (dtm) are generated as a natural extension of the VSM document.

A dtm (Table 4) represents natural language documents as a matrix, where documents terms are represented in columns and the respective documents in rows.

Table 4 – dtm matrix

	Terms					
Documents	Term 1	Term 2	Term 3	Term 4	…	Term n
Document 1	0	1,2	0	6,2		4
Document 2	3,5	0	0	0		6,7
Document 3	0	9,5	2,1	5,0		0
Document 4	5,2	11,6	13,4	0		0
…						
Document n	7,5	0	10	11,3		8,9

All elements a_{ij} represent a document d_i and a term t_j with a respective appropriate weight.

The weight can be a binary value which indicates whether a term exists or not in a document, or a numeric measure indicating how many times a given term occurs or how relevant the term is.

A well-known term weighting function is the TFxIDF *Term Frequency - Inverse Document Frequency* [Salton, et al., 1988], [Singhal, et al., 1996]. This approach is implemented in this project and it is defined as:

$$TF \times IDF = TF(t_k, d_i) \times IDF(t_k) = TF(t_k, d_i) \times \log \frac{|D|}{DF(t_k)} \qquad \text{[Equation 1]}$$

Where:

- $\mathrm{TF}(\mathrm{t_k}, \mathrm{d_i})$ is the number of times that the term $\mathrm{t_k}$ occurs on the document $\mathrm{d_i}$,
- $|\mathrm{D}|$ is the total number of documents in the corpus,
- $\mathrm{DF}(\mathrm{t_k})$ is the number of documents where the term $\mathrm{t_k}$ occurs.

The result of this implementation is a weight that is a statistical measure that determines how important a term is to a document in the context of a given corpus.

The entire corpus goes through an indexing process where the output is a document-term matrix with its respective TFxIDF weights.

3.4 Indexing Emails

In the context of this project, the indexing process aims to prepare the email corpus for the next two phases: classification and evaluation.

As referred above, the main goal of a indexing process is to generate an output in a more appropriated format. The information represented in the documents is converted into a different representation that is easier to be manipulated by computer programs.

Figure 5 represents the indexing procedure where the input is a corpus of emails and the result is a document-term matrix representing each email as a vector of term weights.

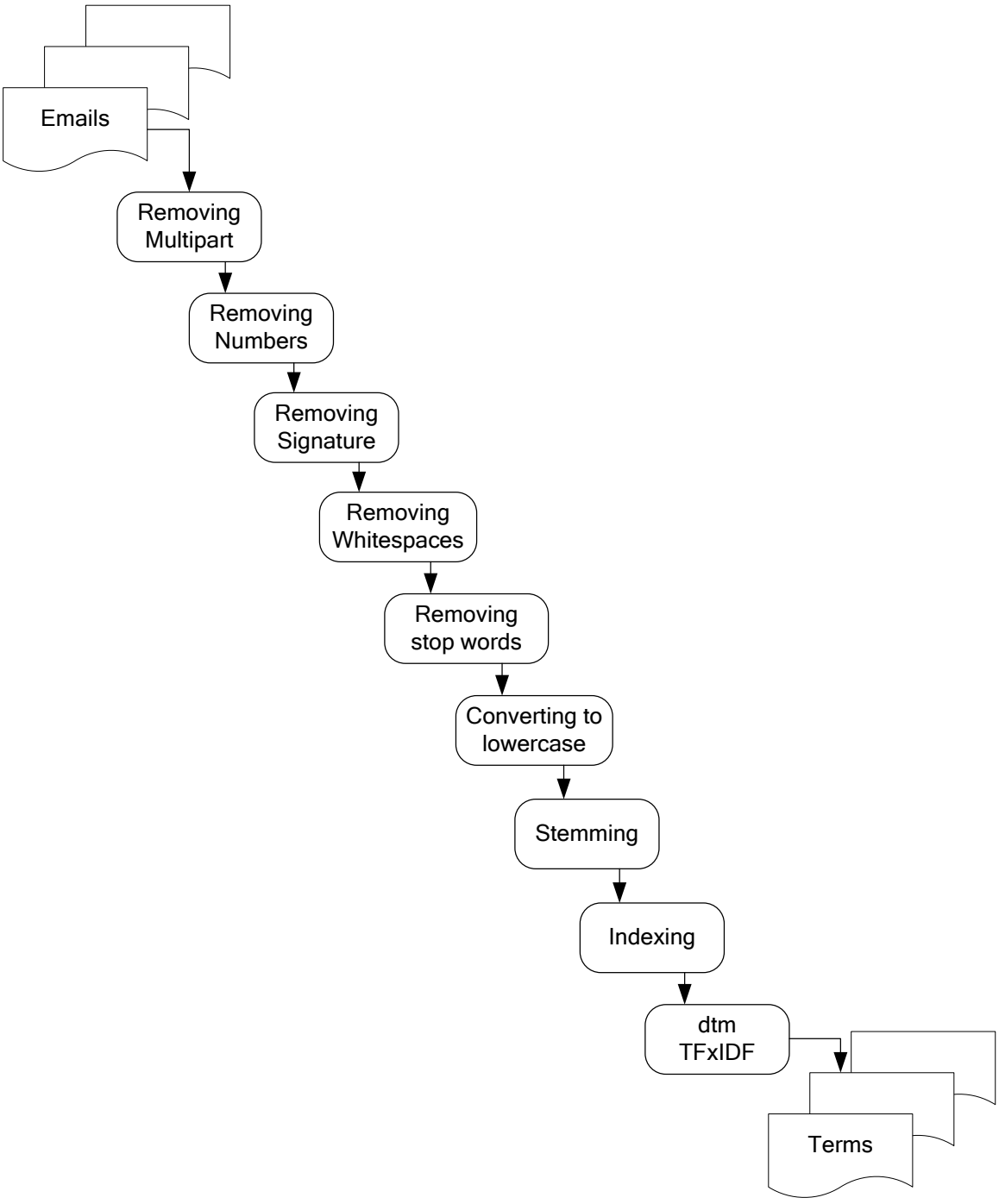

Figure 5 – Indexing Emails

During this process (*Figure 5*) the following transformations are applied to emails:

1. Removing multipart – it removes non-text from multipart emails; it is considered a cleaning task since the result is emails represented only with plain text. Multipart emails are represented by two parts: one plain text (text/plain) and one HTML (text/html).

2. Removing numbers – It removes any number from emails; since numbers expressed on emails are usually irrelevant, all numbers are deleted from their content.
3. Removing signature – It removes signature lines; the information represented on emails signatures is not relevant for classification then signature lines are also deleted.
4. Removing whitespace – It strips unnecessary whitespaces.
5. Removing punctuation – It removes all punctuation marks; as only terms are important for classification tasks, punctuations marks are perfectly unnecessary thus their removal produces cleaner and smaller content, easily to treat on the next phases.
6. Removing stop words – It removes all stop words according to a specific language; as explained in 3.2.3 stop words are frequently words/terms that in most cases have no influence on the classification result.
7. Converting content to lowercase – It converts all characters to lowercase; it is crucial to reduce the computacional effort of the stemming task that all letters are represented in the same case (usually lowercase is preferred) otherwise it will not able to recognize which terms should be transformed to their respective stem.
8. Stemming – It reduces terms to their respective stem; it is explained and exemplified in 3.2.4.
9. Finally, the last task is to generate a document term matrix (dtm) with all terms with a frequency higher than a given threshold. After some tests with the threshold set to two, three, four and five, it was verified that the threshold set to four is the most appropriated to generate a set of relevant terms. So, for all tests the threshold is set to four. As referred above weights are defined by the algorithm TFxIDF (Eq. 1).

Indexing emails is an essential and complex process for email organization; it includes different tasks with numerous methods. The approach described in this section is reimplemented and represented in Chapter 5.

3.5 Conclusion

The main achievement of the work described in this chapter respects to the comprehension and systematization of the indexing process. It described: information retrieval methods used in text processing tasks, how text documents are usually represented and how the specific case of indexing emails is implemented.

The next chapter explains in details how the classification process is executed. The output of this phase is used as the input of the next phase, i.e. the result of the indexing process is used as the input to the classification phase.

Chapter 4

Classification Process

This chapter presents the second phase of a categorization process: learning a classifier and applying the generated classification model.

The algorithms used during the learning process: support vector machine and d-confidence, are also explained.

4.1 Classification

This section presents a description of how a classification process is carried out. The classification process is divided in two phases: classifier learning and classifying. The first one produces a classifier and the second one applies this classifier to predict labels for new instances.

A text classifier, $\Phi(d_j, c_i)$, for classes $\mathrm{C} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_{|\mathrm{C}|}\}$ is generated by an inductive process. This process recognizes characteristics that a document should have to belong to a specific class. To build classifiers for C, it is necessary to label a set of documents, referred as Ω, and to know the value of $\Phi(d_j, c_i)$ for every $\langle d_j, c_i \rangle \in \Omega \; X \; C$.

There are several methods to learn a classifier from training data [Sebastiani, 2002], such as: probabilistic methods, decision tree and decision rule learners, neural networks, example-based methods, support vector machines (SVM), classifier committees and others.

An active learning (2.1.2) approach is adopted to categorize emails during the learning process. Our approach uses a SVM classifier and the d-confidence active learning algorithm; both algorithms are described in the next sections.

4.2 Learning

This phase is called learning since a classifier is learned from a set of pre-labeled examples; as referred in the previous section it is an inductive process where common characteristics are recognized to a specific class. In practice it learns by labeled examples and it is capable to classify new unlabeled examples.

In this project the algorithm adopted to generate a classifier is the well-known algorithm for classification named Support Vector Machine [Cortes, et al., 1995].

SVM [Vapnik, 1995] uses a vector space model to represent a dataset; the main goal is to find a separating hyperplane between any two classes which maximizes the margin that separates the positive and negative examples from each class (*Figure 6*).

The margin is the distance from the hyperplane to the nearest element/point from each class. The nearest points define the support vectors. A good separation is defined by the largest distance between classes.

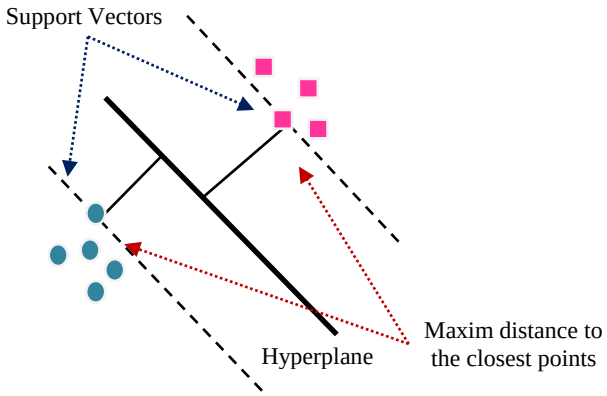

Figure 6 – Support Vector Machine

As indicated before, in this work, the learning phase is based on SVM and on the d-confidence algorithm. Starting from a set of unlabeled emails the main goal is to be able to classify these emails, asking the user as few labels as possible.

The classifier generates a set of labels along with their respective confidence values. The d-confidence algorithm aims to select the most informative unlabeled emails to ask the user to label them. The main goal is to reduce the number of labels that are requested to the user when compared to supervised approaches and common active learning algorithms. The d-confidence algorithm is explained in details on the next section.

This active learning phase is the most important phase of the email organization process as it determines the classes and their respective instances. *Figure 7* represents the learning process.

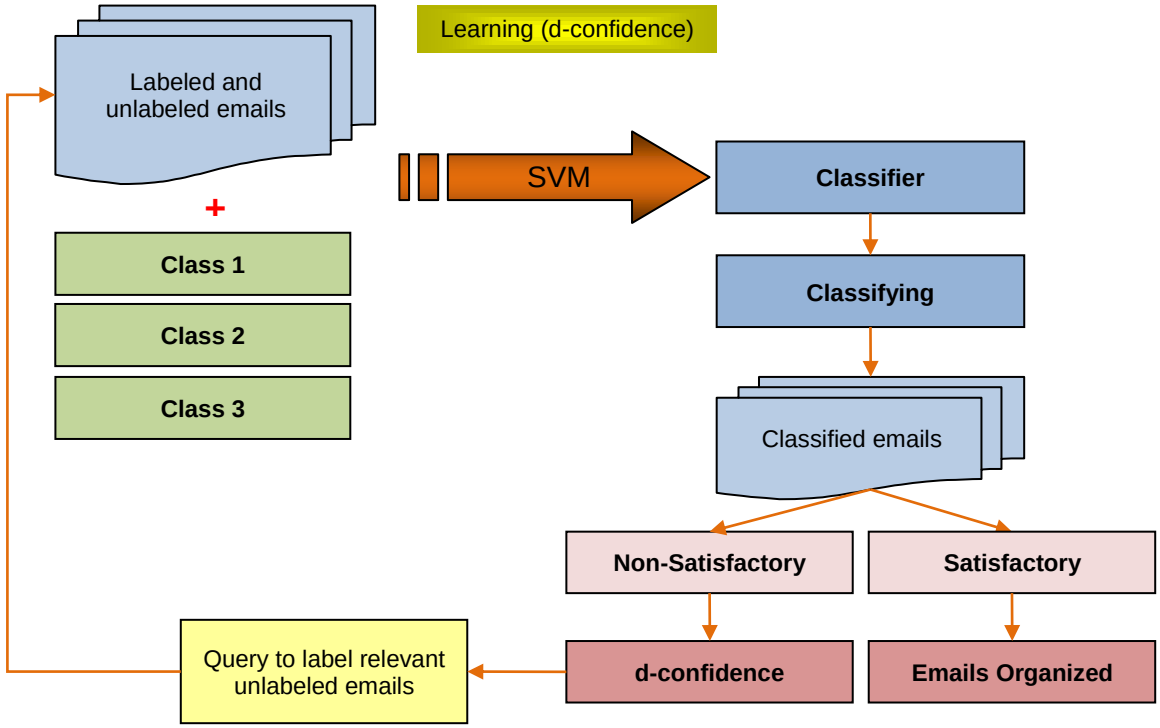

Figure 7 – Classifier Learning with d-confidence

The process represent in *Figure 7* consists of:

1) At the beginning, define a set of target classes and a set of few labeled emails;
2) Learn a classifier using the sets defined previously and the SVM algorithm. This classifier is able to label/classify the non-labeled emails however the effectiveness can be satisfactory or not;
3) If the effectiveness of the classification is satisfactory, the classification process ends and the final result is presented;
4) If the effectiveness is not satisfactory the d-confidence is used to identify informative emails whose labels are requested to the user. These newly labeled examples are then added to the training set that is used to train a new classifier; each iteration is supposed to improve the classifier ability to organize the corpus according to user interests;
5) A new index order of the unlabeled emails is produced; this new index contains a relevant order as a result of the d-confidence algorithm; it sorts unlabeled emails by decreasing order of their informative potencial. Automatically presenting and querying the user to label relevant emails

increases the classification effectiveness and it reduces the user effort when compared to fully supervised settings.

6) At this point there are again a set of target classes and a set of few labeled and many unlabeled emails; the process is restarted.

After generating a good classifier the next step is classifying all the corpus, this process is cited in this section and it is described in this chapter. However, before describing the classifying process it is extremely relevant to explain the d-confidence algorithm.

4.3 D-confidence algorithm

D-confidence is an active learning algorithm proposed by [Escudeiro, et al., 2008] and in this project it is applied to a set of emails with a specific goal: to classify them while requiring from users a reduced effort and no need to dominate any technical issues.

D-confidence is an active learning algorithm that selects queries based on “a criterion that aggregates the posterior classifier confidence and the distance between unlabeled cases and known classes”. This criterion is based in cases that have a low confidence and high distance to known classes. D-confidence is applied in cases where few instances of the target space are labeled and many are unlabeled. It focuses on identifying representative instances from all classes with as few queries as possible.

The queries are selected based on the ratio between classifier confidence and the distance among cases and known classes, this ratio is called d-confidence (or d-conf value). It is represented by the following equation:

$$dConf_i(u) = mean_k\left(\frac{pc_i(c_k|u)}{median_j(d(u,xlab_{j,k}))}\right) \qquad \text{[Equation 2]}$$

Where:

- $(\mathbf{u})$ is an unlabeled example,
- $\mathbf{mean_k}$ is the mean of d-confidence on known classes
- $\mathbf{pc_i}(\mathbf{c_k}|\mathbf{u})$ is the posterior confidence of known classes $\mathbf{c_k}$
- $\mathbf{median_j}(\mathbf{d}(\mathbf{u}, \mathbf{xlab_{j,k}}))$ is the median of the distance between (u) and all know instances belonging to class $\boldsymbol{c_k}$ $(\mathbf{u}, \mathbf{xlab_{j,k}}))$
- $\boldsymbol{d}$ is the distance indicator between $(\mathbf{u})$ and the known cases $(\mathbf{xlab_{j,k}})$

For an unlabeled example$(\mathbf{u})$, the classifier generates the posterior confidence to known classes. Then confidence is divided by the median distance between case $(\mathbf{u})$ and all known cases $(\mathbf{xlab_{j,k}})$ bellowing to the class $\boldsymbol{c_k}$.

In other words, the d-confidence determines the ratio between the classifier confidence and the distance to known classes. It allows querying the user to label only relevant emails, given our current classifier, thereby reducing the user effort.

4.4 Classifying

The last step is to classify the non-labeled emails. This process (*Figure 8*) is done using the classifier generated by the SVM algorithm and the algorithm d-confidence on the previous phase. It applies the classifier to the email corpus aiming to generate a set of classified emails.

An evaluation process is also very important to determine the performance of the methodology proposed, 6.2.3. Chapter 5 describes the prototype used to test this methodology and Chapter 6 the experimental results we have obtained.

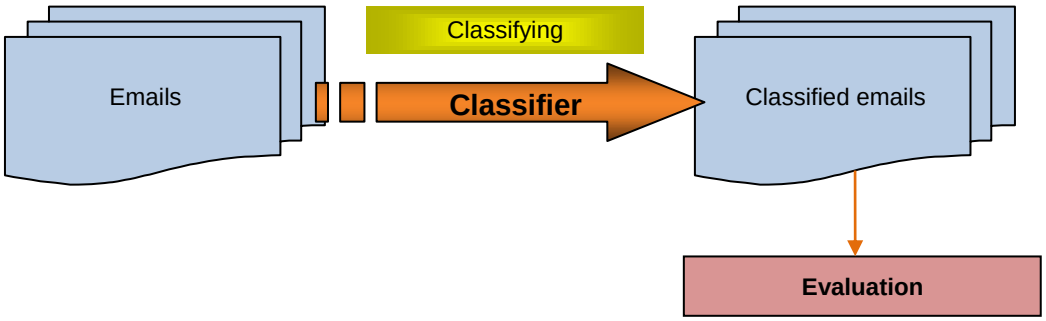

Figure 8 – Classifying Emails

The classifier generates confidence values for all the classes for all emails. If this classification is not satisfactory, as described on the previous sub-section, a new classifier can be generated; but to improve the performance, new queries to classify non-labeled emails are required to the user.

4.5 Conclusion

This chapter presented the classification process of emails categorization; it consists mainly in the learning method to learn a classifier and the application of this classifier to a set of non-classified emails. This chapter also described the algorithms used in this process and how it is a difficult process since each phase has several different tasks and subtasks.

A prototype was implemented to test the proposed methodology. The next chapter addresses this prototype with its characteristics.

Chapter 5

Automatic Email Organization Prototype

This chapter aims to describe the prototype developed for email organization. The main goal of this prototype is to automatically organize emails on a taxonomy reflecting user's own interests while requiring little effort form the user. Once the classification model for user interests is learned it will be applied to fresh incoming emails; thus, newly received emails are placed on the appropriate category without any additional user effort.

The following sections present the specification and architecture of this prototype, and the technology choices for the development phase.

5.1 Specification

The prototype developed reflects the methodology presented on Chapter 3 and Chapter 4. It is a prototype to the specific case of Associação Portuguesa Para a Inteligência Artificial (APPIA) mailing list. The prototype is implemented based on the dataset source provided by APPIA, whom probably will evaluate the final result.

Starting from this dataset, the phases presented on the methodology are implemented; a user interface is developed to allow the categorization according to users' preference. The user has two different roles in the system: to classify their emails and to visualize the new classified emails.

During the manual classification part, users label some examples specifying their preferences; these labeled examples will be later used on the algorithms for learning a classification model and consequently to classify their email, i.e. users' preference are reflected on the automatic categorization process. The visualization is a result of the previous labeling step and the automatic classification process.

5.2 Architecture

The prototype architecture is represented in *Figure 9*. It shows how the different steps and features are connected.

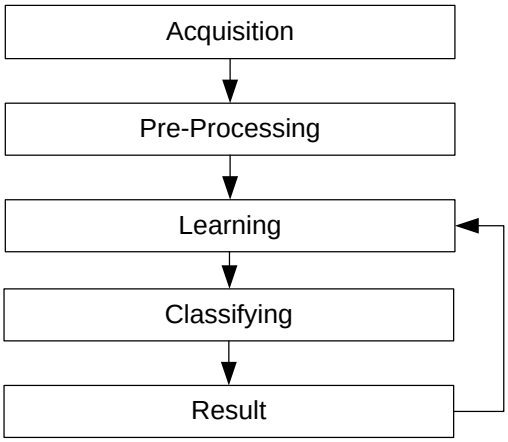

Figure 9 – Prototype architecture

The first phase is the acquisition of the dataset source; this dataset is a set of emails provided by the APPIA. The second phase is the pre-processing where the emails go

through a process of cleaning and indexing, the email content are also represented as document term matrix with weights defined by TFxIDF.

On the third phase the user is asked to label some emails according to a specific category; and then using the SVM learning algorithm a model is generated. On the fourth phase the learning model is applied to all email generating the classification of all emails.

The last phase is to show the results to the user, if the result is not satisfactory the algorithm d-confidence is applied and the user can define new labels and the whole process is done again; the d-confidence is applied to query the user to label relevant emails which have not been labeled before. Each of these phases is explained in detail on the next sub-sections.

The user interaction is represented on *Figure 10*.

Initially the user is asked to give examples from at least two different classes, with these labeled examples the classifier is generated, this classifier is applied to all non-labeled emails and the d-confidence values are calculated.

After, the non-labeled examples, i.e. the examples not classified manually by the user, are sorted by increasing order of d-confidence and are presented to the user. If the user decides to label more emails, these new labeled emails are added to the set of labeled examples and the classifier is generated again from the new labeled set.

The classification process is done as many times as the user wants. In each iteration, the newly labeled emails are added to the training set.

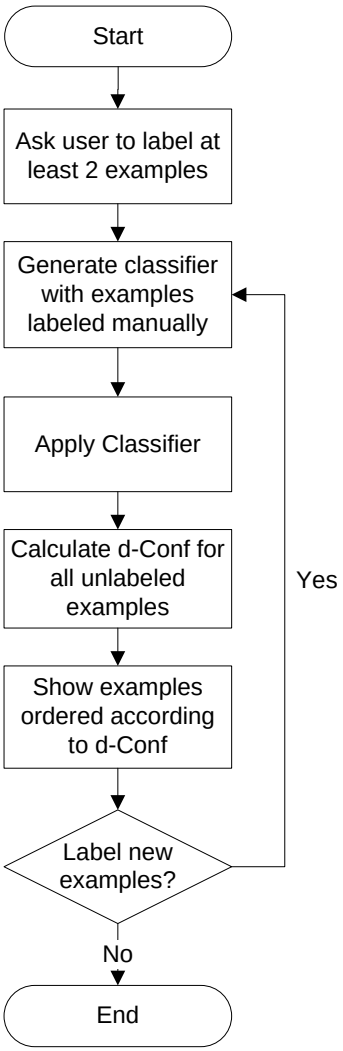

Figure 10 – User Interaction

5.2.1 Acquisition

The activity diagram of the acquisition process is represented as *Figure 11*.

The acquisition process obtains all emails. Emails are loaded from a folder and then the encoding language is checked. After the acquisition of all emails' content, to assure that their content reflects what were represented initially, an encoding process is ran. This process verifies all possible changes caused by different text encodings.

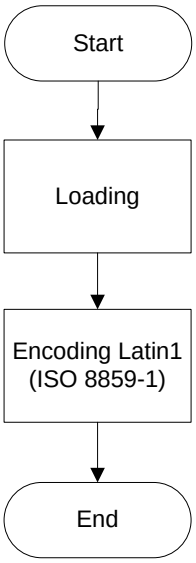

Figure 11 – Acquisition

5.2.2 Pre-processing

This phase is extremely important since pre-processing prepares the corpus for the next phases (classifying and learning). The methods applied during this phase are usually methods used in IR. Pre-processing is a transformative process that reduces the numbers of terms from the emails; it also represents these emails in a more appropriated format, i.e. dtm with TFxIDF weights.

The activity diagram of the pre-processing process is represented as *Figure 12*:

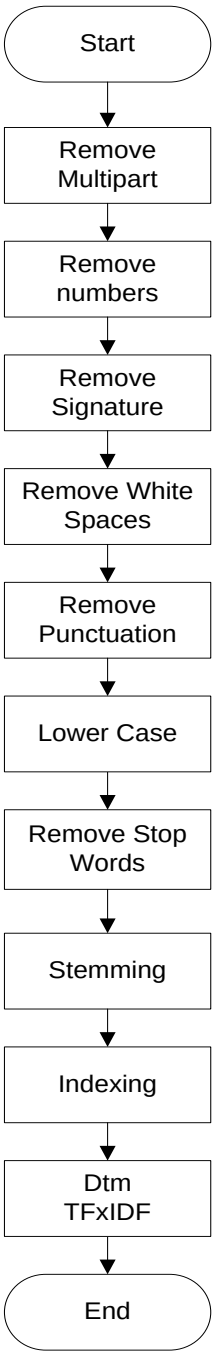

Figure 12 – Pre-processing

This process is defined by a set of different tasks; these tasks are described in section 3.4. The order of each task is very important to reduce the execution time, for example: removing stop word and stemming are much faster when the emails are already clean, without multipart, numbers, punctuation and all the terms in lowercase.

5.2.3 Learning

This phase aims to generate a classifier model able to classify all emails. The activity diagram of the learning process is represented as (*Figure 13*):

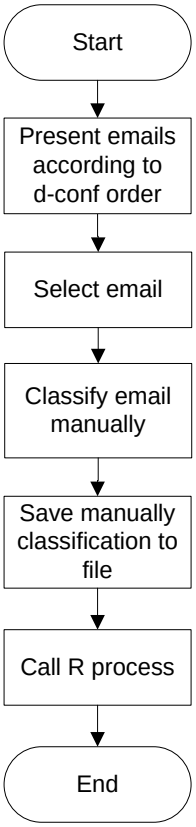

Figure 13 – Learning (Prototype)

Initially at least two different classes and their respective examples need to be defined, this manually classification is saved in a text file and a R process is called.

R (http://www.r-project.org/) is a language and a free software environment for statistical computing and graphics; many features can be easily added via packages. The classifier is generated using R and different packages but mainly the *tm* and *e1071* packages, which support text mining and support vector machines.

The R process is represented in *Figure 14*. When the R process is called it loads the file with the manual classification by the user and also the termxdocument matrix, dtm. In this specific case, where the whole dataset is know, the same dtm with weights from all

documents is always considered; of course in an application where the user gets new emails and wants to classify them, the first phase is redone and a new dtm is generated.

To create the classification model, the SVM algorithm from the e1071 package is called. It generates the classifier to be used on the next phase.

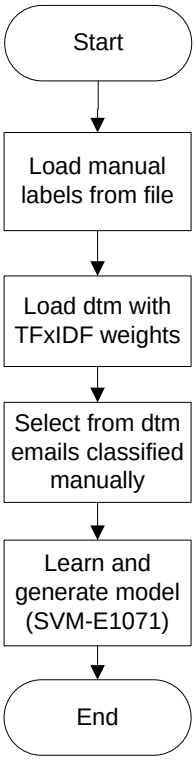

Figure 14 – Learning (R)

By the end of the learning process a classification model is ready to be applied to the corpus to produce a set of classified emails. This process is described in the next sub-section.

5.2.4 Classifying

This phase aims to classify all non-classified emails. The activity diagram of the classifying process is represented according to *Figure 15* and *Figure 16*. This process is implemented also by the user interface and an R process.

In R environment (*Figure 15*) the classifier generated previously is applied to all non-classified emails, before closing the R session two text files are created, one with classification results and another one with d-confidence values to each email classified.

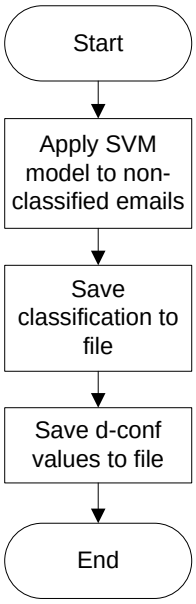

Figure 15 – Classifying (R)

Then, at the application stage, the classification results' file is read and the result is presented to the user. If the result is not satisfactory the user can opt to reclassify all the emails, however the user is asked to label few new examples. These new examples are presented according to the d-confidence value, that list all examples according to their relevance as perceived by the current classification model (*Figure 10*).

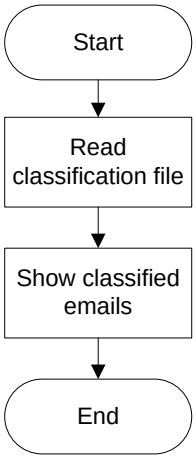

Figure 16 – Classifying (Prototype)

The classifying phase shows the final result of the full email classification process.

5.3 Used Technologies

This project was developed using different technologies; besides the Microsoft operating system all tools and programming languages are open source.

Operating System

- Microsoft Windows Vista Business

Development applications

- NetBeans IDE 6.5.1.
- R 2.9.0.
- Notepad ++ 5.3.1.

Programming Languages

- JAVA Platform, Standard Edition 6 (JDK Version 1.6.0)
- R

Application Programming Interfaces (APIs)

- Lucene Java 2.2.0
- tm 0.3.-4.1. (R)
- e1071 1.5.19 (R)

5.4 Conclusion

The main goal of this chapter was to provide a better understanding of the prototype implemented for the adopted methodology. It described the specification, the architecture and the technologies used.

The next chapter presents the evaluation made to test the effectiveness of the whole classification process.

Chapter 6

Automatic Email Organization Process

Evaluation

The previous chapters described deeply and systematically the methodology used for the problem of automatical email organization. This chapter aims to evaluate this methodology. The main goal is to check if the email classification was efficient and if it reached the proposed objectives. It evaluates the d-confidence algorithm and measures the performance of the classification task.

An introduction of evaluation concepts and methods is presented in sections 6.1 and 6.2. The last section presents in detail the results obtained during evaluation experiments.

6.1 Evaluating

After classifying a set of documents it is important to evaluate classification effectiveness. Different evaluation measurements can be used but mainly it evaluates the ability to generate accurate classification. In the next sub-sections an overview of different performance measures is presented.

6.1.1 Training Set, Test Set and Validation Set

Usually, for an evaluation process the dataset is split into two or three different subsets, these subsets are described as training, test and validation set [Sebastiani, 2002]. Many cases use an approach where the initial data set is split in just two subsets: training set and test set (*Figure 17*).

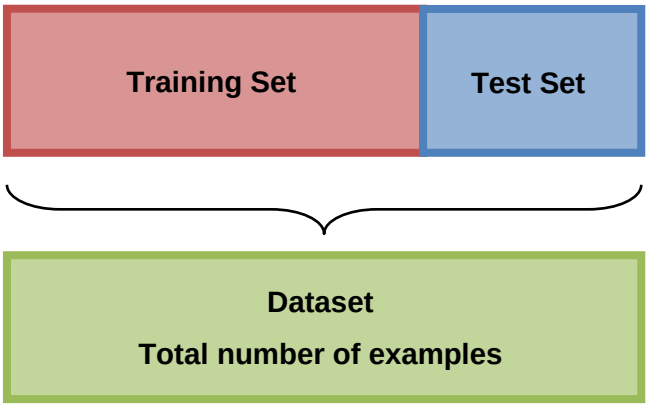

Figure 17 – Training Set and Test Set

The validation set is used to optimize the model and when it is adopted it makes part of the training set (*Figure 18*).

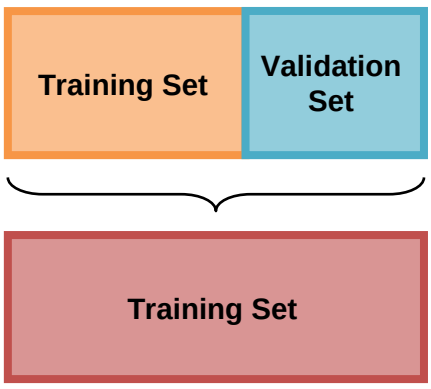

Figure 18 – Validation Set

Training Set

Training sets are used by one or more learning methods to generate classifiers. They are used to build a classification model.

Test Set

Test sets are used to measure the classifier performance. Typically they are used to calculate the error and evaluate the classifier effectiveness.

Validation Set

The validation set is used to optimize classifier parameters or to select a specific classifier; generally it is used to tune the classification model.

In the evaluation process each document d_j is classified automatically by a classifier and then the result is compared with a manually classification made by experts, if both labels match it means the classifier classified the document correctly.

6.1.2 Confusion matrix

Many measures for evaluating the performance of classifiers are based on confusion matrixes, also called contingency tables [Bush, et al., 2008].These matrixes contain information about the number of instances correctly and incorrectly classified.

A confusion matrix is represented as Table 5. After a classification process an instance is classified under a specific class (predicted), considering c and $\bar{c}$ as possible classes the following information is described:

Table 5 – Confusion Matrix

Class c_i		**Correct Classification**	
		c (true)	$\bar{c}$ (false)
Predicted Classification	c (true)	**TP**	**FP**
	$\bar{c}$ (false)	**FN**	**TN**

Where one class is predicted as positive c and another one negative $\bar{c}$ and:

- **TP** – **True Positives** – is the number of instances correctly predicted as class c,
- **TN** – **True Negatives** – is the number of instances correctly predicted as class $\bar{c}$,
- **FP** – **False Positives** – is the number of instances incorrectly predicted as class c,
- **FN** – **False Negatives** – is the number of instance incorrectly predicted as class $\bar{c}$.

Using the confusion matrix it is possible to compute measures for evaluating the performance, such as: precision, recall, accuracy and error. Other types of measures and evaluation methods are presented in the following section.

6.1.3 Precision and recall

Precision and recall measures are calculated according to the following specifications.

Precision

Precision is the proportion of right predictions for the class c.

$$p = \frac{TP}{TP + FP}$$

Recall

Recall is the proportion of documents rightly predicted for the class c.

$$r = \frac{TP}{TP + FN}$$

However, precision and recall values are both required to evaluate performance [Yang, et al., 1999], since improving precision implies degrading recall and the other way round. In 1979, *Rijsbergen* [Rijsbergen, 1979] proposed the F-Measure that combines precision and recall.

F-Measure

$$f_\beta = \frac{(\beta^2 + 1) \cdot p \cdot \mathrm{r}}{(\beta^2 \cdot p + r)}, \beta \in [0, \infty[$$

Where β is a parameter to define the precision and recall weights. Usually β is defined as 1, granting precision and recall the same weight; in this case the previous function, known as the F1 measure, is represented as:

$$f_1 = \frac{2 \cdot p \cdot r}{p + r}$$

6.1.4 Accuracy and error rates

Accuracy and error are more general measures, since they consider the total of classified documents. They are calculated according to the following specifications.

Accuracy

Accuracy is the proportion of right documents classified and it is represented as:

$$a = \frac{TP + TN}{TP + TN + FP + FN}$$

Error

Error is the proportion of wrong classified documents and it is represented as:

$$E = 1 - a = \frac{FP + FN}{TP + TN + FP + FN}$$

6.2 Evaluation Methods

Evaluation methods are directly related with the size of the dataset; in large datasets a lot of information can be easily extracted for validation, in this case it is quite easy to produce a training and test set; however when the dataset is relatively small it is necessary to decide how the initial dataset is going to be separated, defining both training and test sets. The repetitive use of cross validation is a common solution.

This section describes how a cross validation method is applied and some kinds of implementation.

6.2.1 Cross Validation

Cross validation [Kohavi, 1995] is a model evaluation method that uses the initial dataset as training and test sets. One part of the dataset is used to train a learner and the other part to test the learner previously generated. It consists in removing some data before the learning phase; then the data removed is used to test the performance of the classifier. It permits to test the learned model in different data.

Typically this statistical method crosses the training and the testing sets in successive rounds or folds; in each round the data used to learn and validate the classifier is different because sets were crossed [Refaeilzadeh, et al.]. This process is referred as *k-fold* cross validation, where data is partitioned in *k* folds (subsets) with equally or almost equally size and consequently *k* iterations of training and testing are executed. An example of a 3-fold cross validation is exemplified on *Figure 19* :

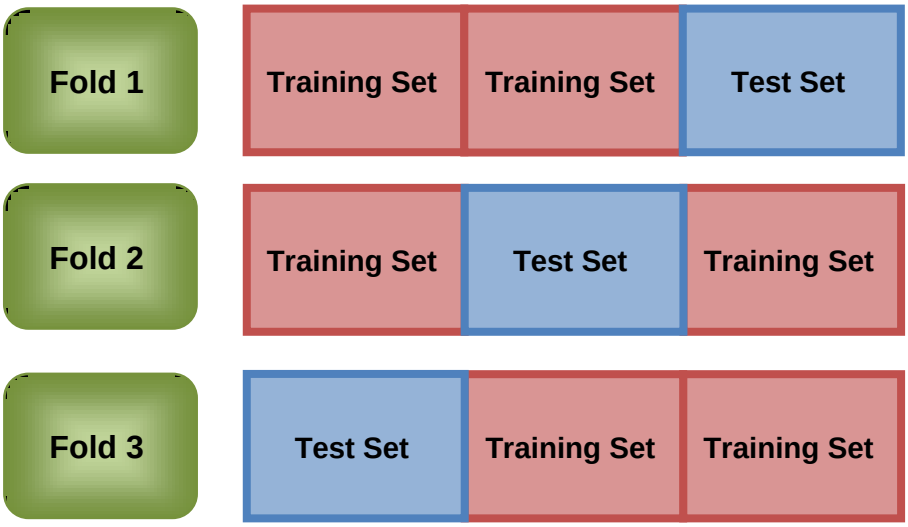

Figure 19 – 3-fold Cross Validation

The initial dataset is divided in three equal folds, two of them are used for training and the remaining one is used for test; on each iteration these folds are crossed.

The next two sub-sections describe the holdout validation and the 10-fold cross validation methods.

6.2.2 Holdout

The holdout method is a basic implementation of cross validation. As explained on the previous sub-section the dataset is separated into two sets [Witten, et al., 2005].; it is common to assign about 2/3 of the dataset as training set and about 1/3 as testing set [Freitas, 2002].

This method is easily comprehensible and applied to big datasets, however the result can have a significantly variation. This disadvantage is connected to the way by which examples are chosen for the training set and for the test set. Since both sets are generated based on a single random partition of the data, it is possible that examples for some classes are present only on the train set or only on the test set, of course that this kind of situation distorts the results.

In most cases to avoid this kind of limitation and to use the available data in a better way, the k-fold cross validation is used.

6.2.3 K-fold Cross Validation

The k-fold cross validation is an appropriate way to improve the effectiveness of a validation process. As referred the data is partitioned in k folds with equally or almost equally size and k iterations are executed. It is important to stress that in each interaction a different fold of data is used for testing though k-1 folds are used for training.

The disadvantage of this method is that it can be a computational and time consuming task, since the classification process is executed k times; despite of this disadvantage it is a very efficient method where all dataset examples are used one time for test and k-1 times for training.

Generally the value adopted is k=10, this is an attractive value especially because it uses 90% of the dataset for training, in other words: it makes predictions using 90% of the data.

The 10-fold cross validation is the method used in this project to evaluate the performance of emails classification.

6.3 Experimental Evaluation

This section describes the evaluation process and the results obtained on the evaluation of the proposed methodology. It aims to:

- Analyze the performance of the d-confidence algorithm in identifying significant examples of each class querying the user to label them as little as possible,
- Evaluate the error rate of the methodology used for classification.

The method adopted for this evaluation is the 10-fold cross validation (6.2.3).

6.3.1 Experimental Setting

The corpus used was provided by APPIA and contains 565 real email messages. These emails come from the APPIA mailing list. It is important to refer that these emails are real emails and not a controlled corpus, specially prepared for evaluation; subsequently the results obtained refer to realistic experiments.

The corpus has messages written in English and in Portuguese (*Figure 20*), with a distribution of 90% and 10% respectively. Since the rate of emails in English is quite high when compared to Portuguese messages, we have decided to opt for English as the main language. This choice is fundamental and has a great impact on the pre-processing stage.

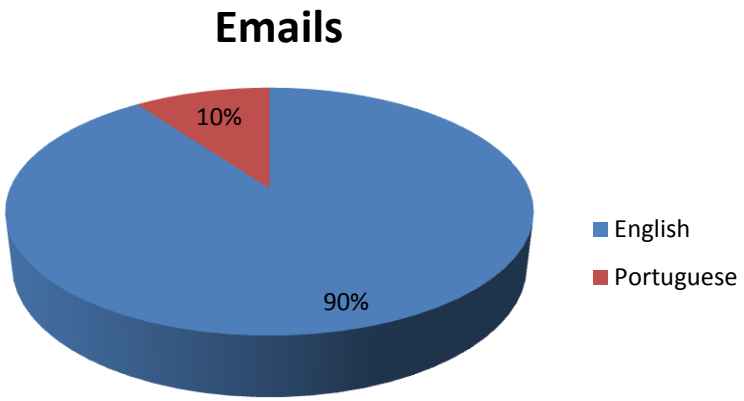

Figure 20 – Emails → Languages

The number of terms in each email varies during the pre-processing phase. Originally, emails have between 39 and 1978 terms each and in the end of this process the number of terms per email varies between 165 and 495.

6.3.2 Evaluation Method

6.3.2.1 Indexing

During the indexing phase (Chapter 3) the following set of pre-processing tasks were applied:

- All emails were loaded from a specific folder,
- Encoding to Latin, especially important for emails represented in Portuguese,
- Removing of multiparts,
- Removing of numbers,
- Removing of signature,
- Removing of whitespaces,
- Removing of punctuation,
- Converting characters to lower case,
- Removing of stopwords in English,
- Stemming for English,
- Generate document term matrix with weights calculated with the algorithm TFxIDF.

As a result of this process, emails become represented in a clean way, easier to manipulate. These emails are represented in a document term matrix (dtm) and in text files documents. The dtm representation is important for the classification process and the text files documents are important to present information to users.

The dtm contains 565 emails and 1251 terms. From these 565 emails, (*Figure 21*) 449 have relevant information. The indexing process identifies 116 emails with no relevant information.

The indexing process recognizes 449 emails with relevant information based on the steps presented above and on *Chapter 3*. In this process, numbers, whitespaces, punctuation and stopwords are removed. It is also defined a threshold for the minimum

number of documents a term appears in, i.e., it considers all the terms that appear at least in 4 documents.

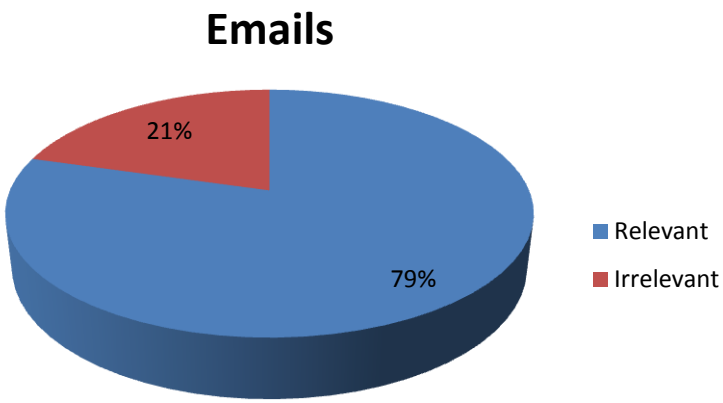

Figure 21 – Emails → Relevant terms

The 449 relevant emails, corresponding to approximately 80% of the initial corpus, have been used in evaluation. The other 20% are merely useless and they have been removed from the corpus.

The text files contain the representation of the emails after the pre-processing phase. Annex I shows how an email looks like before pre-processing and after this phase.

Another relevant characteristic is that original emails in our corpus contain between 39 and 1708 words. The graphic bellow (*Figure 22*) represents the distribution of words per email:

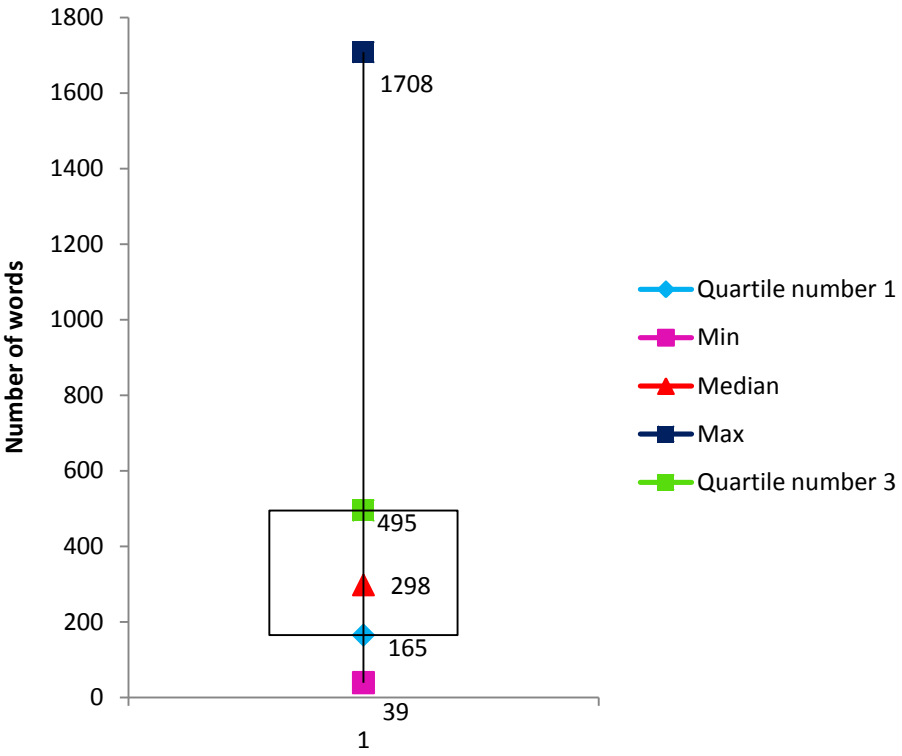

Figure 22 – Emails → Number of words per email (Box Plot)

It is interesting to observe (Figure 22) that although the number of words per email varies between 39 and 1708, most of them have between 165 and 495 words, i.e. 50% of the emails have between 165 and 495 words, 25% have less than 165 words and 25% have more than 495 words.

Nowadays users write their emails in very different formats and structures, consequently in this area there is still a need to develop more studies to try to interpret these emails.

6.3.2.2 Classifying and Evaluating

After obtaining the document term matrix with all emails and terms, the dataset is ready for the classification and evaluation phases.

Initially all emails were labeled manually, hence an analysis of results could be done. Six different categories have been considered:

1. APPIA – Emails related to APPIA,
2. Call of papers – Emails related with call for papers for any kind of conference,

3. AI (Artificial Intelligence) – Emails related with artificial intelligence,
4. Job – Emails related with announcements of jobs and fellowships,
5. Call of Participation – Emails related with participation in any kind of event,
6. TLEIA (Trabalhos de Licenciatura em Inteligência Artificial – Bachelor projects in artificial intelligence) – Emails related with the contest of bachelor projects in AI.

The number of emails per categories is represented on *Figure 23*. The most representative categories are: Call of papers and Call of participation.

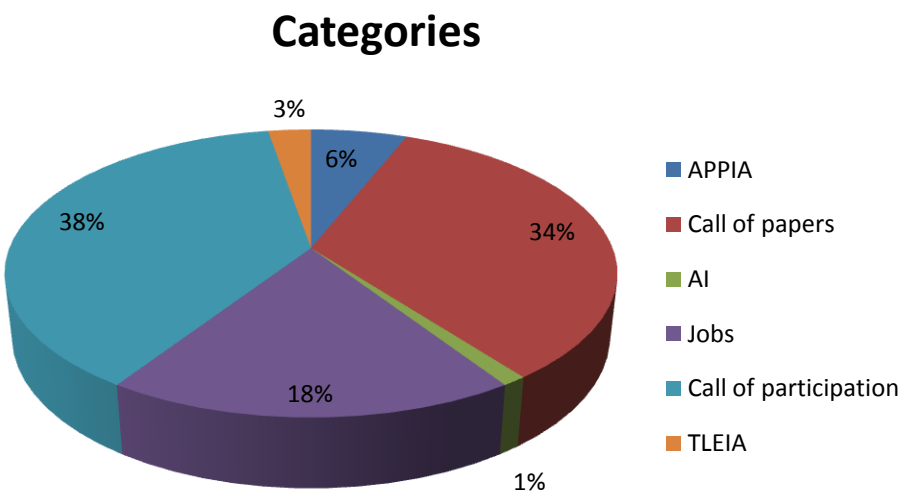

Figure 23 – Emails' categories

After labeling all emails, the 10-fold cross validation method was implemented. To implement this validation method the following steps were adopted:

- The 449 emails set has been partitioned in 10 subsets by random sampling: 4 subsets have 44 emails; 3 other subsets have 45 emails and the remaining 3 subsets have 46 emails;
- For each fold one of these subsets is set apart to test and the remaining 9 subsets are used to train. Each training set has 402, 403 or 404 instances
- In each fold there are 402, 403 or 404 iterations according to the size of the training set,

- In each fold, 2 labeled examples are initially set on the training set; then, in each iteration, a query is made to the user and one labeled example is added to the labeled set,
- The SVM algorithm is applied to the labeled set,
- The d-confidence values are generated for unlabeled examples in the training set,
- The query for the next interaction is the unlabeled case with the lowest d-confidence value,
- The error and accuracy rates are calculated on the test set of the current fold.
- Performance statistics are estimated by the average of the 10 folds.

The 10-fold cross validation method was a time-consuming task, since the classification process is generated 4490 times.

6.3.3 Results

This subsection presents the experimental results we have obtained.

The following graphic presents the error rates achieved as new queries (labeled cases) are added to the training set (*Figure 24).* The graphic represents the micro-average of the error rate obtained at each iteration in the 10-folds. The error rate starts at approximately 80% reducing to approximately 29% as new queries are added. After approximately 150 queries, error seems to stabilize, with just a slight upward tendency. Annex II contains all the error rates obtained and Annex III the confusion matrixes for each fold.

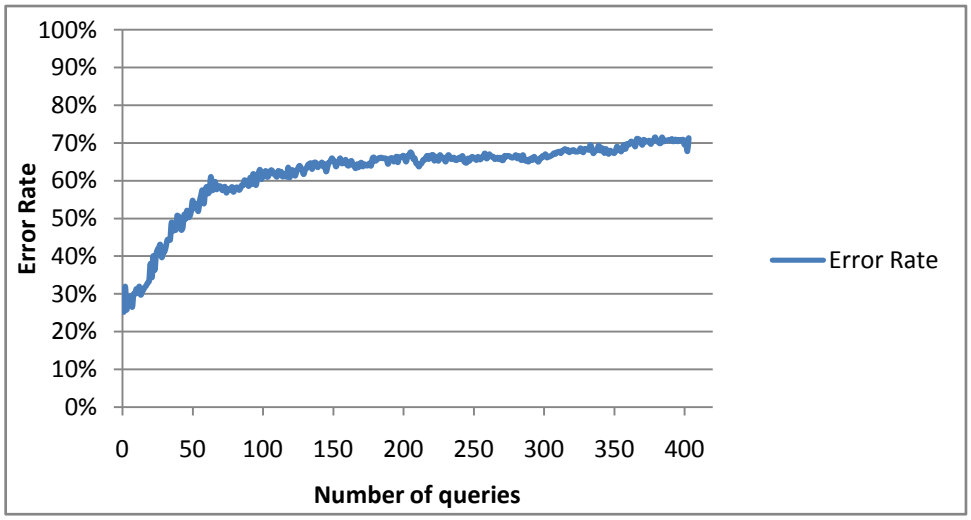

Figure 24 – Error

From the results obtained (*Figure 24*) it is possible to observe that after few queries the classifier improves significantly; it shows that even with a few number of labeled examples it is possible to generate a good classification. The test confirms that d-confidence is a good active learning algorithm and it is able to generate an email categorization without the need to require an expensive fully supervised setting.

The average of the error rate per category is represented on *Figure 25*; this error rate was generated on the last iteration of each fold and using micro-average.

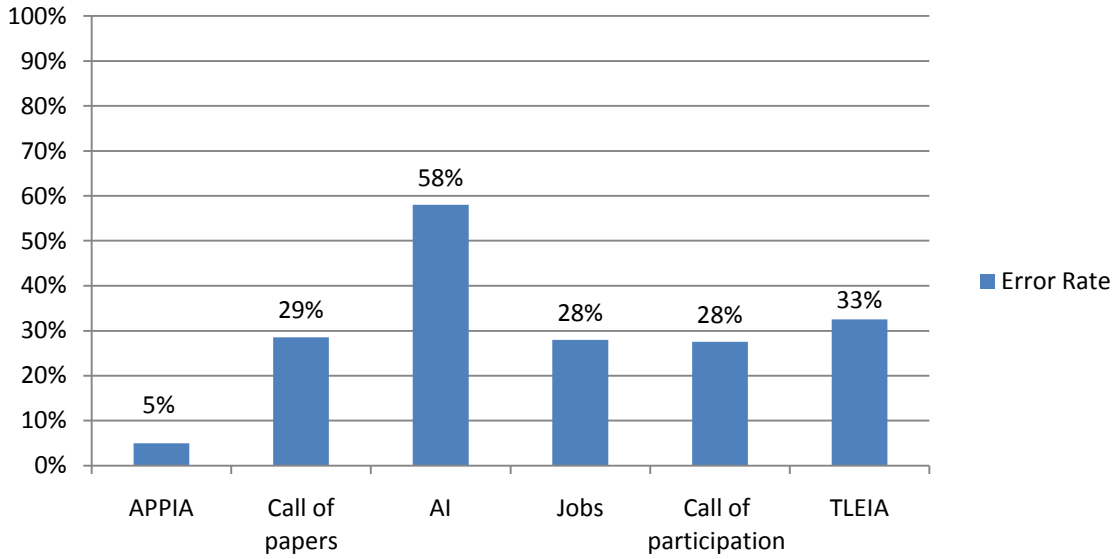

Figure 25 – Error rate per category

The classes with higher error rate are *AI* and *TLEIA*, these 2 classes are also the classes with a very low (*Figure 26*) numbers of examples, representing 1% and 3% of the dataset. Based on these results we can deduce a correlation between the number of examples of a given class on the corpus and the error rate on that class: a low rate of examples corresponds to a high error rate. However there is one case: class *APPIA* where the number of examples is low, representing 6% of the dataset, and the error rate is also low. In this situation the error rate is low because emails content is clear and specific, i.e., the classifier recognizes easily the common characteristics of email from this class.

The error rate is directly related to the kind of information the examples contain; if they contain general information they are hard to be characterized; that is what happens with *AI* and *TLEIA* classes. This fact influences directly the error rate.

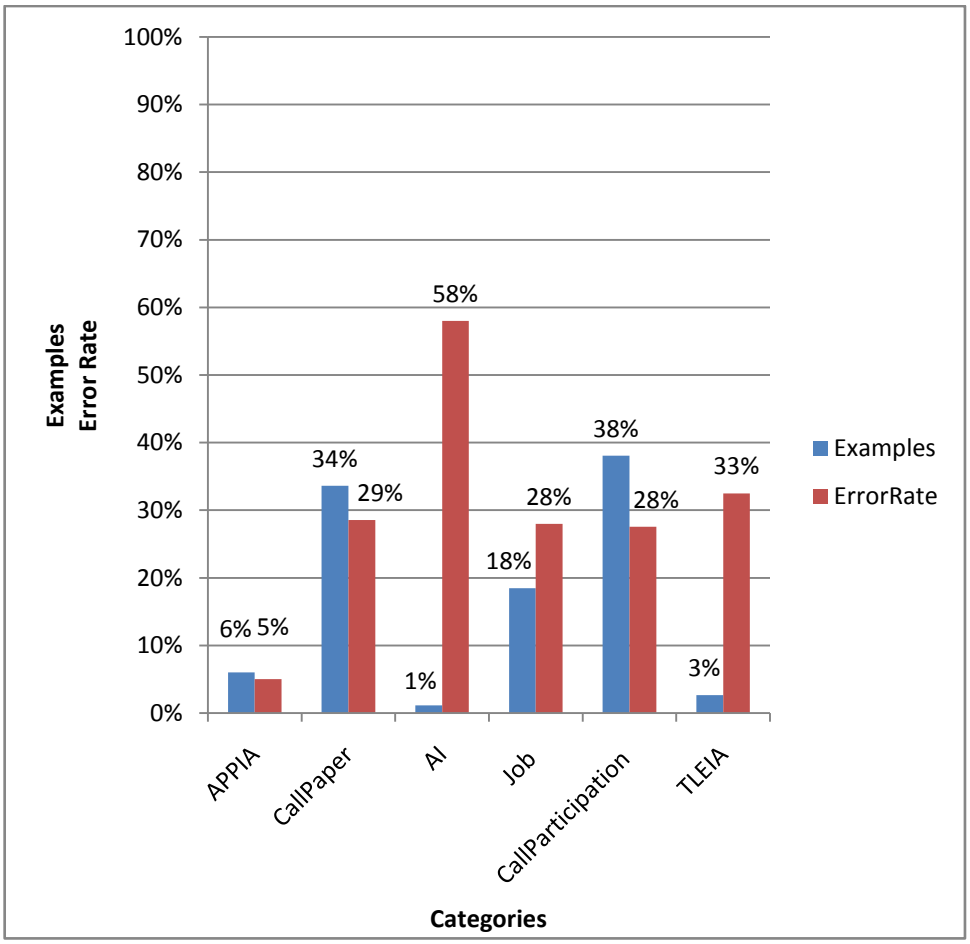

Figure 26 – Error Rate versus Number of examples

The classes *Call of Papers*, *Call of Participation* and *Job* present almost the same error rate around 30%; they are also the 3 classes with higher number of examples. Although these 3 classes contains a high number of examples, their content is general, it influences directly the error rate. For example: a call of paper could be also considered a call of participation; the content can be also ambiguous. Another point is that emails represent text in natural language which might be highly ambiguous.

The classification generated an error rate of 29%; however, in different categories this value might be very different. The error is directly related to the distribution of classes (categories) in the training set – for example: the category that has fewer examples has also a higher error rate – and also to the concepts that are represented by classes – classes representing clear-cut concepts have higher accuracy. Also the fact that some classes represent ambiguous concepts affects directly the error rate since the classifier does not classify properly the emails according to their content.

6.4 Conclusion

This chapter aimed to describe the evaluation process of the proposed methodology. In general the results obtained seem good, since it reaches an error rate of 29% on a real corpus. The algorithm d-confidence generated an error rate closer to the final error value, with less labeled examples than the number of examples used on supervised learning. It is important since it reduces the users' effort. During the tests was possible to notice that two characteristics are important for the classification effectiveness: the number of examples per class and how the information is represented. High number of examples and emails with specific information are easier to be characterized.

Chapter 7

Conclusion

This work was a great opportunity to acquire a high level of knowledge about text mining and methods of text classification and information retrieval.

The conclusion of the work done is present in this chapter. Next sections describe the achievements of this work and possible work to be done in the near future; finally the last section presents general remarks about the entire process of development of this project.

7.1 Achievements

The main goal of this project was to develop a methodology to organize automatically emails on a taxonomy reflecting user's own interests at a low cost. Main concerns were accuracy and low user effort, i.e. classify mail messages on the appropriate category and reduce the workload required to the user when specifying self interests; to reach this goal different phases were defined: indexing emails, classification and evaluation.

It is important to refer that a period of study and analysis of text mining, machine learning, information retrieval and extraction were extremely relevant and crucial for the development of this project. During all phases new knowledge was acquired, new concepts and methods were learned.

In all phases during the development of this project different achievements were reached:

Indexing emails

- Knowledge acquisition of the global indexing process;
- Knowledge acquisition of natural language processing methods, such as tokenization, normalization, stemming and others;
- Knowledge acquisition of machine learning, specially information retrieval;
- Analysis of Lucene Java 2.2.0.;
- Analysis of R 2.9.0.;
- Indexing routines implementation, in R, using the text mining package tm 0.3.-4.1.;
- The adequate input for the classification phase was generated based on the dataset source.

Classification

- Knowledge acquisition of text classification;
- Analysis of algorithms for learning and classification;
- Analysis of the support vector machine approach;
- Analysis of the d-confidence algorithm;
- Learning implementation to generate a classification model. It was also implemented in R and it used the package e1071 1.5.19;

- d-confidence implementation;
- Application of the classification model to the initial dataset;
- Classification of all emails.

Evaluation

- Knowledge acquisition of evaluation measures;
- Knowledge acquisition of evaluation methods;
- Analysis of the k-fold cross validation method;
- Error rates implementation, also in R;
- Implementation of the 10-fold cross validation to evaluate the classification performance;
- Analysis of the result obtained with 10-fold cross validation;
- Satisfactory classification on a real corpus;
- The users' effort on labeling examples was reduced.

Prototype

Besides all the achievements reached during the categorization process; there are other points that should be focused about the prototype that has been implemented.

It is important to stress that the goals proposed by this project were achieved. Using the automatic email classification process that we propose, it is possible to generate an email organization according to user specific interests. The task of organizing emails was mainly the use of text classification techniques divided in 3 different phases: indexing, classification and evaluation.

The results obtained using the 10-fold cross validation method were quite good. The methodology proposed reached an accuracy rate of 70%. The error rate could be considered high but since the source dataset is represented in natural language the result obtained was somehow expected.

7.2 Future Work

Despite of the fact that the automatic email categorization is satisfactory, there are some limitations and optimizations that can be done. This section identifies these limitations and proposes optimization approaches for them.

- Users are not recognized; the prototype is not multiuser; it is able to learn just a single user's preferences. There is no method to identify a user, the reorganization of users would be a very useful functionality since the system would be able to recognize the users' specific preferences and needs.
- The prototype and methodology were developed based in a real case but it was not yet tested in a real situation. It would be very interesting to develop and apply this proposal in a realistic environment.

7.3 Final Remarks

The main benefits obtained with this work are related to the classification of emails as corpus and particularly to the knowledge that has been acquired.

This project was an excellent opportunity to acquire solid knowledge about text mining and methods of text classification and information retrieval. Besides the achieved results, this unique and captivating area contains much more knowledge to be learned what many more challenges to explore in the future.

References

Ayodele, Taiwo, Khusainov, Rinat and Ndzi, David. 2007. Email classification and summarization: a machine learning approach. 2007.

Buitelaar, Paul; Cimiano, Phillipp. 2008. Ontology Learning. [book auth.] Paul Buitelaar and Phillipp Cimiano. *Ontology learning and population: bridging the gap between text and knowledge.* 2008, pp. 5-9.

Bush, William S., Edwards, Todd L. and Dudek, Scott M. 2008. Alternative contingency table measures improve the power and detection of multifactor dimensionality reduction. 2008.

Chaves, Marcirio Silveira. 2003. Um estudo e apreciação sobre algoritmos de stemming para a língua portuguesa. Cartagena de Indias, Colômbia : s.n., 2003.

Cortes, C. and Vapnik, Vladimir. 1995. Support-vector networks. 1995, pp. 273-297.

David A., Cohn, Zoubin, Ghahramani and Michael I., Jordan. 1996. Active learning with statistical models. s.l. : Journal of Artificial Intelligence Research, 1996.

Escudeiro, Nuno and Jorge, Alipio. 2008. Learning partially specified concepts with D-Confidence. 2008.

Freitas, Alex A. 2002. Classification. *Data mining and knowledge discovery with evolutionary algorithms.* s.l. : Springer, 2002.

Giorgetti, Daniela and Sebastiani, Fabrizio. 2003. Automating survey coding by multiclass text categorization techniques. 2003.

Helfman, J. and Isbell, C. 1995. Ishmail: immediate identification of important information. 1995.

Klimt, Bryan and Yang, Yiming. 2004. The Enron Corpus: a new dataset for email classication research. s.l. : ECML, 2004.

Kohavi, R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. 1995, pp. 1137-1145.

Kolcz, A., Chowdhury, A. and Alspector, J. 2004. The impact of feature selection on signature-driven spam detection. 2004.

Maynard, Diana; Li, Yaoyong; Peters, Wim;. 2008. NLP Techniques for Term Extraction and Ontology Population. [book auth.] Paul Builtelaar and Philipp Cimiano. *Ontology learning and population: bridging the gap between text and knowledge.* 2008.

Meyer, T. A. and Whateley, B. 2004. SpamBayes: effective open-source, bayesian based, email classification system. 2004.

Mladenic, Dunja and Grobelnik, Marko. 1999. Feature selection for unbalanced class distribution and Naive Bayes. San Francisco, CA : Morgan Kaufmann Publishers, 1999, pp. 258-267.

Muslea, Ion A. 2002. Active learning with multiple views. 2002.

Muslea, Ion, Minton, Steven and Knoblock, Craig A. 2006. Active learning with multiple views. s.l. : Journal of Artificial Intelligence Research, 2006, Vol. 27, pp. 203-233.

Pantel, Patrick; Pennacchiotti, Marco;. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. Sydney, Australia : s.n., 2006.

Payne, T. and Edwards, P. 1997. Interface agents that learn: an investigation of learning issues in a mail interface. *Applied Artificial Intelligence.* 1997.

Refaeilzadeh, Payam, Tang, Lei and Liu, Huan. Cross Validation. *Encyclopedia of Database Systems.* 2009 : Springer.

Rijsbergen, C. J. Van. 1979. Information Retrieval. s.l. : Butterworth, 1979.

Salton, G. and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. 1988.

Salton, G., Wong, A. and Yang, C.S. 1975. A vector space model for automatic indexing. s.l. : ACM, 1975.

Sebastiani, Fabrizio. 2002. Machine learning in automated text categorization. 2002, Vol. 34(1), pp. 1-47.

Segal, R.B. and Kephart, J.O. 2002. MailCat: an intelligent assistant for organizing email. s.l. : Proceedings of 3rd Annual Conference on Autonomous Agent, 2002.

Singhal, A., et al. 1996. Document length normalization. 1996.

Spark-Jones, K. and Willett, P. 1997. Readings in information retrieval. San Francisco: Morgan Kaufmann : s.n., 1997.

Stolfo, S. J., et al. 2004. Detecting viral propagations using email behavior profiles. 2004.

Vapnik, Vladimir. 1995. The nature of statistical learning theory. New York : Springer, 1995.

Witten, Ian H. and Frank, Eibe. 2005. Data Mining: practical machine learning tools and techniques. s.l. : Morgan Kaufmann, 2005.

Witten, Ian H. 2004. Text mining. 2004.

Yang, Yiming and Liu, X. 1999. A re-examination of text categorization methods. Berkley : s.n., 1999, pp. 42–49.

Zhu, Xiaojin and Goldberg, Andrew B. 2009. Introduction to semi-supervised learning. 2009.

Annex I – Email before and after indexing process

Before:

Return-Path: <rede-return-36-analide=di.uminho.pt@appia.pt>
From: "Joao Gama" <jgama@liacc.up.pt>
To: "APPIA - Rede" <rede@appia.pt>
Subject: [rede.APPIA] CFP- Learning 06
Date: Fri, 12 May 2006 18:01:07 +0100
Organization: LIACC
Message-ID: <1147453267.2423.2.camel@lira.niaad.liacc.up.pt>
MIME-Version: 1.0
Content-Type: multipart/alternative;
boundary="----=_NextPart_000_1FC8_01C7CE22.AB763850"
X-Mailer: Evolution 2.6.1 (2.6.1-1.fc5.2)
Thread-Index: AcZ15sLuUCbF6dTWQh6/Wf0bxfOs8A==
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.3138
X-No-Archive: yes
List-Help: <mailto:rede-help@appia.pt>
List-Unsubscribe: <mailto:rede-unsubscribe@appia.pt>
List-Subscribe: <mailto:rede-subscribe@appia.pt>
X-OlkEid: 1E44663AA5070033F1E5144AAE5D388A2E01F9BB

This is a multi-part message in MIME format.

------=_NextPart_000_1FC8_01C7CE22.AB763850
Content-Type: text/plain;
charset="iso-8859-1"
Content-Transfer-Encoding: 7bit

(please excuse duplications - please share this announcement)

L E A R N I N G ' 0 6

CALL FOR PAPERS

Vilanova i la Geltru (near Barcelona, Spain), October 2-5, 2006

Organized by the LARCA research group of UPC,
with help from the GREC and TALP research groups

DEADLINE: June 21, 2006 (longest day of the year)

Learning is an essential process, whether for living beings,
for physical machines, or for computational programs, if they
are to behave efficiently in front of complex tasks or evolving
environments. Consequently, intensive research is being dedicated
to the many aspects of this rich phenomenon.

Most of this research adopts a particular point of view: biological,
psychological, educational, mathematical, algorithmic, or
computational,
for instance. Nevertheless, many researchers consider that perspectives
coming from other scientific fields are important, sometimes critical,
to produce significant advances and useful results.

LEARNING'06 is one more along a series of biennial events, held since
1998,
intended to provide a forum for interdisciplinary study and discussion
of
the different aspects of learning. It is hoped that the interactions
taking
place at these events will contribute to a fluid interchange of
different
kinds of know-how to accelerate the progress along different research
conceptions and lines.

This event is located within proximity, in time and space,
of the two International Conferences on Discovery Science and
Algorithmic Learning Theory, to take place in Barcelona, october 7-10.

Program Committee of Learning'06:
Jose L Balcazar (Chair, Universitat Politecnica de Catalunya, Barcelona,

Spain);
Jess Cid-Sueiro (Universidad Carlos III, Madrid, Spain),
Peter Dayan (University College, London, United Kingdom),
Jose Dorronsoro (Universidad Autonoma de Madrid, Spain),
Pascal Frossard (Ecole Polytechnique Federale, Lausanne, Switzerland),
Joao Gama (Universidade do Porto, Portugal),
Huma Lodhi (University of Sheffield, United Kingdom),
Rafael Morales (Universidad de Malaga, Spain),
Fernando Perez-Cruz (Universidad Carlos III, Madrid, Spain),
Nestor Schmajuk (Duke University, Durham, NC, United States),
Tinne Tuytelaars (University of Leuven, Belgium).

URL: http://lsi.epsevg.upc.es/~learn06/learn2006.html

------=_NextPart_000_1FC8_01C7CE22.AB763850
Content-Type: text/html;
charset="iso-8859-1"
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>
<HEAD>
<META HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; =
charset=3Diso-8859-1">
<META NAME=3D"Generator" CONTENT=3D"MS Exchange Server version =
6.5.7036.0">
<TITLE>[rede.APPIA] CFP- Learning 06</TITLE>
</HEAD>
<BODY>
<!-- Converted from text/plain format -->

<P><FONT SIZE=3D2>(please excuse duplications - please share this =
announcement)</FONT>
</P>

<P><FONT SIZE=3D2>L E A R N I N G ' 0 6</FONT>
</P>

<P><FONT SIZE=3D2>CALL FOR PAPERS</FONT>
</P>

<P><FONT SIZE=3D2>Vilanova i la Geltru (near Barcelona, Spain), October =
2-5, 2006</FONT>
</P>

<P><FONT SIZE=3D2>Organized by the LARCA research group of UPC, </FONT>

<FONT SIZE=3D2>with help from the GREC and TALP research =
groups</FONT>
</P>

<P><FONT SIZE=3D2>DEADLINE: June 21, 2006 (longest day of the =
year)</FONT>
</P>

<P><FONT SIZE=3D2>Learning is an essential process, whether for living =
beings, </FONT>

<FONT SIZE=3D2>for physical machines, or for computational programs, =
if they </FONT>

<FONT SIZE=3D2>are to behave efficiently in front of complex tasks =
or evolving </FONT>

<FONT SIZE=3D2>environments. Consequently, intensive research is =
being dedicated </FONT>

<FONT SIZE=3D2>to the many aspects of this rich phenomenon.</FONT>
</P>

<P><FONT SIZE=3D2>Most of this research adopts a particular point of =
view: biological, </FONT>

<FONT SIZE=3D2>psychological, educational, mathematical, =
algorithmic, or</FONT>

<FONT SIZE=3D2>computational, </FONT>

<FONT SIZE=3D2>for instance. Nevertheless, many researchers consider =
that perspectives </FONT>

<FONT SIZE=3D2>coming from other scientific fields are important, =
sometimes critical, </FONT>

<FONT SIZE=3D2>to produce significant advances and useful =
results.</FONT>
</P>

<P><FONT SIZE=3D2>LEARNING'06 is one more along a series of biennial =
events, held since</FONT>

<FONT SIZE=3D2>1998,</FONT>

<FONT SIZE=3D2>intended to provide a forum for interdisciplinary =
study and discussion</FONT>

<FONT SIZE=3D2>of </FONT>

<FONT SIZE=3D2>the different aspects of learning. It is hoped that =
the interactions</FONT>

<FONT SIZE=3D2>taking </FONT>

<FONT SIZE=3D2>place at these events will contribute to a fluid =
interchange of</FONT>

<FONT SIZE=3D2>different </FONT>

<FONT SIZE=3D2>kinds of know-how to accelerate the progress along =
different research </FONT>

<FONT SIZE=3D2>conceptions and lines. </FONT>
</P>

<P><FONT SIZE=3D2>This event is located within proximity, in time and =
space, </FONT>

<FONT SIZE=3D2>of the two International Conferences on Discovery =
Science and </FONT>

<FONT SIZE=3D2>Algorithmic Learning Theory, to take place in =
Barcelona, october 7-10.</FONT>
</P>

<P><FONT SIZE=3D2>Program Committee of Learning'06: </FONT>

<FONT SIZE=3D2>Jose L Balcazar (Chair, Universitat Politecnica de =
Catalunya, Barcelona,</FONT>

<FONT SIZE=3D2>Spain);</FONT>

<FONT SIZE=3D2>Jess Cid-Sueiro (Universidad Carlos III, Madrid, =
Spain),</FONT>

<FONT SIZE=3D2>Peter Dayan (University College, London, United =
Kingdom),</FONT>

<FONT SIZE=3D2>Jose Dorronsoro (Universidad Autonoma de Madrid, =
Spain),</FONT>

<FONT SIZE=3D2>Pascal Frossard (Ecole Polytechnique Federale, =
Lausanne, Switzerland),</FONT>

<FONT SIZE=3D2>Joao Gama (Universidade do Porto, Portugal),</FONT>

<FONT SIZE=3D2>Huma Lodhi (University of Sheffield, United =
Kingdom),</FONT>

<FONT SIZE=3D2>Rafael Morales (Universidad de Malaga, Spain),</FONT>

<FONT SIZE=3D2>Fernando Perez-Cruz (Universidad Carlos III, Madrid, =
Spain),</FONT>

<FONT SIZE=3D2>Nestor Schmajuk (Duke University, Durham, NC, United =
States),</FONT>

<FONT SIZE=3D2>Tinne Tuytelaars (University of Leuven, =
Belgium).</FONT>
</P>

<P><FONT SIZE=3D2>URL: <A =
HREF=3D"http://lsi.epsevg.upc.es/~learn06/learn2006.html">http://lsi.epse=
vg.upc.es/~learn06/learn2006.html</A></FONT>
</P>

</BODY>
</HTML>
------=_NextPart_000_1FC8_01C7CE22.AB763850--

After

(please excuse duplications - please share this announcement)

L E A R N I N G ' 0 6

CALL FOR PAPERS

Vilanova i la Geltru (near Barcelona, Spain), October 2-5, 2006

Organized by the LARCA research group of UPC,
with help from the GREC and TALP research groups

DEADLINE: June 21, 2006 (longest day of the year)

Learning is an essential process, whether for living beings,
for physical machines, or for computational programs, if they
are to behave efficiently in front of complex tasks or evolving
environments. Consequently, intensive research is being dedicated
to the many aspects of this rich phenomenon.

Most of this research adopts a particular point of view: biological,
psychological, educational, mathematical, algorithmic, or
computational,
for instance. Nevertheless, many researchers consider that perspectives
coming from other scientific fields are important, sometimes critical,
to produce significant advances and useful results.

LEARNING'06 is one more along a series of biennial events, held since 1998,
intended to provide a forum for interdisciplinary study and discussion of

the different aspects of learning. It is hoped that the interactions taking place at these events will contribute to a fluid interchange of different kinds of know-how to accelerate the progress along different research conceptions and lines.

This event is located within proximity, in time and space, of the two International Conferences on Discovery Science and Algorithmic Learning Theory, to take place in Barcelona, october 7-10.

Program Committee of Learning'06:
Jose L Balcazar (Chair, Universitat Politecnica de Catalunya, Barcelona, Spain);
Jess Cid-Sueiro (Universidad Carlos III, Madrid, Spain),
Peter Dayan (University College, London, United Kingdom),
Jose Dorronsoro (Universidad Autonoma de Madrid, Spain),
Pascal Frossard (Ecole Polytechnique Federale, Lausanne, Switzerland),
Joao Gama (Universidade do Porto, Portugal),
Huma Lodhi (University of Sheffield, United Kingdom),
Rafael Morales (Universidad de Malaga, Spain),
Fernando Perez-Cruz (Universidad Carlos III, Madrid, Spain),
Nestor Schmajuk (Duke University, Durham, NC, United States),
Tinne Tuytelaars (University of Leuven, Belgium)
URL: http://lsi.epsevg.upc.es/~learn06/learn2006.html

Annex II – Error Rates

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
1	60%	82%	82%	89%	61%	89%	75%	66%	98%	66%	77%
2	69%	67%	62%	63%	61%	61%	75%	73%	70%	80%	68%
3	69%	82%	82%	89%	61%	61%	75%	73%	70%	80%	74%
4	69%	82%	82%	63%	61%	70%	82%	73%	66%	64%	71%
5	69%	67%	64%	63%	61%	91%	82%	73%	70%	84%	72%
6	69%	67%	62%	63%	61%	91%	73%	73%	70%	77%	71%
7	69%	67%	84%	65%	61%	91%	73%	75%	70%	80%	74%
8	69%	67%	82%	63%	61%	61%	75%	73%	70%	80%	70%

9	69%	67%	82%	63%	61%	61%	75%	73%	70%	80%	70%
10	69%	67%	82%	63%	61%	61%	75%	73%	70%	66%	69%
11	60%	82%	78%	65%	61%	61%	75%	73%	70%	73%	70%
12	60%	67%	82%	63%	61%	61%	75%	75%	70%	66%	68%
13	67%	82%	82%	63%	61%	61%	75%	75%	70%	66%	70%
14	69%	67%	82%	63%	61%	61%	75%	73%	70%	75%	70%
15	69%	62%	76%	74%	61%	61%	75%	75%	70%	66%	69%
16	67%	64%	78%	63%	61%	61%	75%	77%	70%	66%	68%
17	69%	67%	76%	59%	59%	61%	75%	75%	73%	66%	68%
18	69%	87%	60%	61%	65%	61%	75%	75%	64%	55%	67%
19	76%	71%	62%	63%	61%	61%	75%	75%	59%	64%	67%
20	58%	67%	64%	48%	61%	59%	70%	75%	55%	64%	62%
21	60%	64%	67%	74%	61%	70%	75%	73%	48%	66%	66%
22	60%	67%	64%	57%	52%	61%	68%	52%	52%	66%	60%
23	60%	78%	76%	61%	57%	59%	61%	73%	50%	64%	64%
24	60%	56%	56%	63%	59%	63%	68%	55%	55%	64%	60%
25	60%	69%	53%	63%	46%	70%	55%	52%	52%	64%	58%
26	58%	67%	58%	63%	41%	59%	57%	57%	55%	64%	58%
27	58%	62%	51%	63%	54%	59%	57%	52%	52%	61%	57%
28	56%	60%	58%	63%	63%	70%	59%	59%	52%	64%	60%
29	60%	60%	67%	63%	59%	50%	55%	59%	55%	64%	59%
30	60%	58%	53%	61%	52%	63%	59%	61%	57%	64%	59%
31	53%	58%	49%	72%	57%	52%	61%	52%	55%	66%	57%
32	53%	56%	56%	61%	57%	48%	59%	52%	52%	64%	56%
33	53%	49%	49%	61%	57%	65%	57%	52%	55%	59%	56%
34	53%	51%	51%	59%	54%	59%	59%	57%	55%	59%	56%
35	53%	51%	53%	48%	39%	43%	59%	61%	52%	50%	51%

36	53%	47%	56%	70%	50%	37%	59%	57%	50%	55%	53%
37	53%	60%	51%	54%	37%	59%	59%	55%	50%	52%	53%
38	56%	60%	44%	54%	43%	52%	57%	55%	52%	57%	53%
39	53%	44%	42%	48%	43%	48%	57%	55%	52%	50%	49%
40	53%	60%	49%	48%	37%	50%	52%	50%	50%	45%	49%
41	53%	60%	56%	43%	54%	52%	43%	48%	50%	43%	50%
42	53%	60%	47%	52%	46%	50%	50%	57%	50%	66%	53%
43	53%	64%	51%	50%	50%	50%	48%	61%	48%	48%	52%
44	53%	53%	40%	46%	46%	57%	48%	57%	45%	45%	49%
45	56%	60%	53%	41%	43%	50%	48%	57%	41%	52%	50%
46	51%	53%	47%	35%	43%	59%	50%	52%	45%	43%	48%
47	58%	51%	42%	41%	43%	61%	45%	59%	55%	41%	50%
48	56%	60%	44%	46%	41%	67%	45%	57%	39%	39%	49%
49	60%	49%	47%	54%	39%	41%	52%	57%	43%	36%	48%
50	56%	51%	40%	48%	33%	43%	55%	48%	45%	34%	45%
51	47%	51%	40%	43%	37%	50%	50%	48%	48%	59%	47%
52	53%	47%	42%	43%	39%	48%	50%	50%	43%	52%	47%
53	53%	56%	42%	43%	39%	39%	41%	45%	48%	55%	46%
54	47%	56%	40%	50%	43%	48%	45%	50%	48%	55%	48%
55	51%	56%	44%	50%	35%	43%	45%	48%	48%	34%	45%
56	47%	53%	42%	48%	37%	43%	43%	50%	41%	34%	44%
57	44%	49%	44%	43%	33%	41%	39%	50%	43%	39%	43%
58	49%	56%	44%	50%	33%	50%	41%	45%	43%	50%	46%
59	49%	51%	36%	46%	39%	46%	30%	43%	45%	43%	43%
60	51%	53%	33%	46%	30%	48%	32%	43%	34%	45%	42%
61	53%	58%	40%	48%	37%	43%	27%	39%	39%	50%	43%
62	53%	53%	44%	41%	30%	48%	25%	43%	39%	50%	43%

63	44%	49%	31%	41%	30%	43%	34%	39%	41%	36%	39%
64	53%	51%	42%	50%	37%	41%	34%	34%	41%	41%	42%
65	47%	51%	42%	48%	37%	37%	34%	34%	41%	43%	41%
66	44%	49%	40%	46%	37%	39%	34%	32%	43%	39%	40%
67	49%	47%	44%	50%	37%	41%	39%	39%	39%	39%	42%
68	49%	47%	33%	50%	37%	39%	36%	39%	45%	41%	42%
69	44%	49%	40%	46%	35%	41%	36%	36%	45%	41%	41%
70	47%	51%	42%	48%	39%	39%	30%	34%	45%	41%	42%
71	47%	49%	33%	52%	35%	46%	36%	43%	43%	41%	43%
72	49%	51%	47%	48%	35%	37%	30%	43%	43%	41%	42%
73	44%	47%	44%	54%	39%	39%	34%	32%	43%	39%	42%
74	47%	47%	40%	52%	35%	43%	36%	41%	43%	48%	43%
75	44%	49%	44%	52%	30%	41%	36%	39%	45%	41%	42%
76	49%	51%	42%	50%	37%	43%	34%	30%	45%	41%	42%
77	49%	49%	40%	52%	39%	46%	32%	32%	45%	41%	42%
78	47%	53%	40%	52%	35%	33%	41%	34%	43%	39%	42%
79	44%	56%	44%	52%	33%	37%	39%	39%	48%	39%	43%
80	47%	53%	42%	54%	37%	39%	34%	34%	43%	39%	42%
81	47%	58%	40%	54%	37%	41%	34%	32%	36%	39%	42%
82	44%	58%	42%	54%	33%	41%	39%	27%	41%	39%	42%
83	44%	56%	42%	54%	37%	43%	36%	30%	41%	41%	42%
84	49%	53%	40%	52%	33%	43%	36%	30%	43%	41%	42%
85	40%	49%	44%	52%	30%	46%	32%	30%	43%	45%	41%
86	40%	51%	44%	48%	28%	41%	39%	30%	45%	45%	41%
87	40%	51%	44%	48%	30%	41%	34%	27%	41%	41%	40%
88	40%	51%	49%	46%	30%	43%	34%	30%	41%	43%	41%
89	33%	53%	47%	43%	35%	48%	34%	27%	43%	45%	41%

90	40%	53%	49%	46%	35%	43%	39%	30%	39%	41%	41%
91	31%	51%	44%	46%	33%	43%	39%	27%	41%	36%	39%
92	36%	53%	42%	46%	33%	46%	41%	34%	41%	39%	41%
93	33%	53%	44%	43%	28%	46%	39%	23%	34%	39%	38%
94	36%	53%	42%	43%	30%	39%	41%	27%	34%	39%	39%
95	38%	53%	42%	43%	30%	46%	39%	39%	45%	36%	41%
96	40%	51%	42%	46%	30%	41%	41%	27%	39%	36%	39%
97	33%	51%	38%	43%	28%	39%	43%	25%	41%	36%	38%
98	33%	51%	38%	46%	28%	41%	34%	27%	36%	36%	37%
99	40%	51%	42%	46%	28%	41%	45%	25%	43%	34%	40%
100	36%	47%	40%	46%	28%	41%	45%	25%	36%	36%	38%
101	31%	47%	44%	46%	28%	35%	41%	27%	39%	45%	38%
102	31%	47%	47%	43%	28%	39%	34%	27%	39%	39%	37%
103	36%	47%	47%	43%	30%	39%	39%	27%	36%	45%	39%
104	33%	47%	53%	48%	28%	35%	41%	23%	39%	41%	39%
105	33%	47%	49%	41%	28%	37%	34%	27%	39%	41%	38%
106	33%	47%	49%	41%	30%	35%	39%	27%	32%	39%	37%
107	33%	47%	47%	41%	35%	33%	43%	25%	34%	39%	38%
108	33%	47%	49%	41%	33%	35%	41%	30%	39%	36%	38%
109	33%	42%	47%	43%	37%	37%	45%	25%	36%	39%	39%
110	38%	44%	47%	43%	35%	37%	45%	25%	36%	39%	39%
111	31%	47%	49%	43%	33%	37%	39%	20%	36%	39%	37%
112	36%	42%	49%	43%	37%	39%	43%	20%	36%	36%	38%
113	40%	42%	47%	43%	33%	37%	41%	23%	34%	36%	38%
114	38%	40%	49%	43%	39%	39%	41%	27%	32%	41%	39%
115	36%	40%	49%	41%	41%	37%	43%	25%	34%	39%	38%
116	40%	40%	47%	41%	37%	39%	43%	23%	36%	36%	38%

117	44%	47%	47%	41%	35%	39%	41%	30%	30%	36%	39%
118	38%	40%	42%	41%	33%	39%	43%	23%	30%	36%	36%
119	38%	40%	47%	43%	37%	48%	43%	25%	32%	39%	39%
120	42%	40%	42%	43%	35%	48%	39%	23%	27%	36%	38%
121	40%	33%	42%	41%	33%	43%	43%	25%	34%	36%	37%
122	40%	38%	49%	43%	35%	39%	43%	27%	27%	36%	38%
123	44%	38%	51%	41%	37%	39%	43%	25%	32%	36%	39%
124	42%	33%	42%	43%	35%	39%	39%	27%	36%	36%	37%
125	40%	36%	42%	43%	35%	37%	39%	25%	34%	36%	37%
126	38%	31%	47%	43%	33%	37%	34%	25%	36%	36%	36%
127	40%	33%	44%	41%	30%	35%	39%	27%	39%	36%	37%
128	38%	40%	47%	43%	30%	37%	39%	25%	39%	36%	37%
129	40%	36%	49%	41%	33%	39%	39%	27%	41%	39%	38%
130	42%	33%	49%	41%	28%	39%	36%	27%	39%	36%	37%
131	38%	36%	42%	41%	35%	35%	43%	27%	34%	36%	37%
132	40%	33%	42%	43%	30%	37%	39%	25%	32%	36%	36%
133	40%	31%	42%	46%	33%	37%	39%	30%	25%	34%	36%
134	40%	31%	42%	41%	30%	37%	41%	27%	30%	34%	35%
135	40%	31%	44%	43%	30%	41%	41%	34%	27%	36%	37%
136	40%	31%	44%	41%	30%	37%	39%	30%	27%	34%	35%
137	38%	31%	42%	43%	30%	39%	39%	30%	25%	34%	35%
138	40%	33%	44%	41%	30%	39%	39%	27%	25%	34%	35%
139	42%	33%	47%	41%	30%	46%	36%	27%	27%	34%	36%
140	40%	33%	44%	43%	33%	39%	36%	25%	32%	32%	36%
141	38%	33%	42%	41%	35%	39%	36%	30%	32%	32%	36%
142	40%	31%	42%	39%	30%	39%	36%	27%	34%	32%	35%
143	40%	33%	44%	37%	37%	37%	39%	30%	34%	32%	36%

144	40%	33%	44%	41%	28%	39%	41%	27%	30%	32%	36%
145	40%	40%	49%	39%	35%	41%	39%	30%	32%	32%	38%
146	42%	36%	44%	41%	30%	39%	39%	27%	30%	34%	36%
147	40%	36%	42%	37%	28%	39%	39%	27%	32%	32%	35%
148	40%	40%	42%	35%	30%	37%	34%	25%	34%	32%	35%
149	40%	38%	40%	37%	26%	37%	36%	25%	32%	30%	34%
150	44%	36%	42%	37%	28%	37%	34%	23%	34%	30%	34%
151	42%	36%	44%	39%	28%	37%	39%	25%	30%	30%	35%
152	42%	36%	47%	37%	41%	33%	43%	23%	32%	30%	36%
153	40%	38%	44%	41%	30%	33%	43%	25%	30%	30%	35%
154	40%	33%	42%	37%	30%	35%	45%	23%	32%	32%	35%
155	38%	33%	40%	37%	35%	33%	39%	25%	32%	30%	34%
156	38%	36%	42%	43%	33%	37%	43%	23%	30%	30%	35%
157	38%	31%	40%	43%	33%	39%	45%	23%	32%	30%	35%
158	42%	33%	42%	39%	28%	37%	41%	25%	32%	30%	35%
159	42%	33%	42%	41%	30%	35%	36%	23%	32%	30%	34%
160	42%	38%	44%	41%	30%	35%	39%	25%	34%	30%	36%
161	42%	40%	44%	43%	30%	35%	39%	25%	32%	30%	36%
162	42%	36%	47%	41%	26%	35%	41%	23%	34%	30%	35%
163	44%	36%	36%	41%	26%	35%	41%	25%	34%	30%	35%
164	42%	38%	40%	39%	28%	35%	45%	25%	36%	30%	36%
165	40%	38%	40%	43%	30%	35%	43%	27%	34%	30%	36%
166	40%	42%	40%	43%	30%	35%	43%	27%	36%	30%	37%
167	42%	40%	40%	41%	33%	33%	36%	27%	34%	30%	36%
168	40%	40%	40%	46%	33%	35%	41%	27%	34%	30%	36%
169	40%	40%	42%	41%	33%	35%	39%	27%	32%	30%	36%
170	40%	40%	40%	37%	28%	33%	41%	25%	36%	32%	35%

171	38%	40%	40%	48%	35%	35%	36%	25%	36%	30%	36%
172	38%	42%	40%	39%	35%	35%	36%	25%	34%	32%	36%
173	38%	44%	38%	43%	30%	33%	39%	25%	36%	32%	36%
174	36%	40%	42%	43%	33%	33%	43%	23%	36%	32%	36%
175	38%	40%	38%	43%	33%	35%	41%	25%	34%	32%	36%
176	36%	40%	42%	46%	30%	33%	39%	23%	34%	32%	35%
177	36%	42%	40%	43%	33%	35%	41%	25%	34%	32%	36%
178	31%	40%	38%	43%	28%	35%	36%	27%	32%	30%	34%
179	33%	42%	36%	41%	28%	33%	36%	25%	34%	30%	34%
180	31%	44%	38%	46%	26%	33%	36%	30%	34%	30%	35%
181	31%	44%	38%	43%	26%	35%	36%	27%	34%	30%	34%
182	33%	42%	38%	39%	24%	37%	39%	25%	34%	30%	34%
183	31%	47%	36%	41%	24%	35%	34%	27%	34%	32%	34%
184	31%	42%	38%	39%	24%	35%	36%	27%	34%	32%	34%
185	31%	44%	38%	37%	26%	37%	39%	27%	34%	27%	34%
186	31%	44%	36%	39%	28%	35%	39%	25%	32%	32%	34%
187	31%	44%	38%	37%	24%	39%	36%	27%	32%	32%	34%
188	29%	42%	40%	39%	28%	39%	36%	27%	32%	30%	34%
189	31%	44%	38%	41%	33%	41%	39%	25%	32%	32%	36%
190	31%	42%	38%	39%	28%	39%	36%	27%	32%	32%	34%
191	31%	42%	36%	35%	30%	43%	36%	25%	32%	30%	34%
192	29%	42%	38%	37%	28%	39%	39%	25%	32%	32%	34%
193	33%	42%	42%	39%	28%	41%	36%	25%	32%	30%	35%
194	31%	42%	44%	35%	26%	41%	39%	27%	32%	30%	35%
195	29%	44%	44%	35%	26%	39%	32%	25%	32%	30%	34%
196	29%	44%	44%	39%	30%	41%	32%	27%	32%	32%	35%
197	29%	47%	44%	37%	26%	39%	32%	25%	32%	27%	34%

198	31%	44%	40%	37%	28%	41%	30%	25%	34%	27%	34%
199	31%	44%	44%	35%	26%	39%	30%	25%	34%	27%	34%
200	31%	42%	44%	35%	26%	37%	32%	27%	34%	25%	33%
201	31%	44%	42%	39%	30%	39%	32%	27%	32%	25%	34%
202	31%	44%	44%	41%	26%	37%	36%	30%	34%	25%	35%
203	27%	44%	44%	39%	24%	39%	30%	25%	34%	27%	33%
204	29%	40%	42%	39%	26%	39%	30%	25%	36%	27%	33%
205	33%	40%	42%	39%	22%	37%	30%	25%	32%	25%	32%
206	31%	44%	42%	39%	22%	37%	27%	27%	34%	25%	33%
207	31%	44%	49%	39%	22%	39%	30%	27%	32%	30%	34%
208	33%	40%	49%	39%	24%	37%	30%	27%	34%	27%	34%
209	33%	42%	49%	39%	24%	39%	34%	30%	34%	30%	35%
210	33%	42%	49%	41%	24%	39%	30%	30%	36%	32%	36%
211	31%	47%	49%	41%	24%	39%	34%	30%	36%	32%	36%
212	33%	44%	49%	41%	24%	39%	30%	30%	34%	32%	36%
213	31%	47%	44%	41%	26%	39%	30%	32%	32%	32%	35%
214	31%	44%	47%	39%	24%	37%	30%	25%	36%	32%	34%
215	31%	44%	47%	35%	22%	39%	34%	25%	36%	32%	35%
216	33%	44%	49%	33%	22%	39%	30%	23%	36%	30%	34%
217	27%	44%	47%	37%	22%	37%	30%	25%	34%	32%	33%
218	29%	49%	44%	33%	24%	37%	32%	27%	36%	32%	34%
219	29%	47%	47%	33%	24%	37%	30%	27%	32%	30%	33%
220	31%	47%	47%	30%	22%	37%	30%	25%	34%	32%	33%
221	27%	49%	44%	30%	22%	37%	32%	25%	36%	30%	33%
222	27%	49%	49%	35%	24%	39%	32%	25%	39%	30%	35%
223	29%	47%	42%	37%	22%	35%	34%	25%	36%	30%	34%
224	27%	44%	51%	35%	24%	35%	32%	25%	36%	32%	34%

225	27%	44%	49%	37%	28%	35%	34%	23%	39%	32%	35%
226	27%	44%	47%	35%	22%	35%	32%	25%	36%	30%	33%
227	27%	49%	49%	35%	24%	35%	32%	25%	36%	30%	34%
228	29%	47%	47%	35%	22%	35%	34%	23%	39%	30%	34%
229	27%	47%	44%	37%	24%	35%	34%	25%	39%	30%	34%
230	29%	49%	42%	35%	26%	35%	34%	30%	41%	30%	35%
231	27%	49%	44%	35%	22%	35%	34%	25%	41%	27%	34%
232	27%	49%	40%	35%	22%	35%	34%	25%	36%	30%	33%
233	27%	47%	42%	35%	22%	35%	34%	25%	41%	30%	34%
234	27%	47%	42%	35%	24%	35%	36%	27%	41%	30%	34%
235	27%	49%	42%	35%	24%	35%	34%	27%	39%	30%	34%
236	27%	47%	40%	35%	24%	35%	34%	30%	39%	30%	34%
237	27%	47%	42%	35%	26%	35%	34%	30%	41%	30%	35%
238	27%	49%	44%	35%	26%	35%	34%	27%	39%	30%	35%
239	27%	49%	40%	35%	24%	35%	34%	27%	41%	30%	34%
240	27%	47%	42%	37%	22%	35%	34%	30%	36%	30%	34%
241	31%	49%	42%	35%	26%	35%	34%	30%	34%	27%	34%
242	29%	49%	40%	30%	24%	35%	34%	27%	36%	30%	33%
243	29%	51%	44%	33%	24%	35%	34%	30%	39%	30%	35%
244	31%	51%	47%	33%	24%	35%	34%	32%	34%	32%	35%
245	31%	51%	44%	33%	22%	35%	36%	32%	36%	32%	35%
246	31%	51%	47%	33%	20%	35%	34%	30%	34%	27%	34%
247	36%	49%	47%	35%	22%	35%	34%	30%	34%	27%	35%
248	31%	49%	42%	33%	22%	37%	34%	32%	36%	25%	34%
249	31%	49%	42%	35%	20%	35%	36%	30%	34%	25%	34%
250	29%	47%	44%	35%	20%	37%	36%	30%	39%	25%	34%
251	29%	49%	42%	35%	20%	37%	36%	30%	39%	25%	34%

252	29%	49%	47%	35%	20%	35%	36%	30%	39%	27%	35%
253	31%	44%	42%	35%	17%	35%	36%	30%	39%	27%	34%
254	31%	47%	44%	33%	20%	35%	36%	32%	39%	27%	34%
255	31%	49%	44%	35%	17%	35%	36%	34%	34%	27%	34%
256	29%	47%	42%	33%	17%	35%	34%	32%	41%	27%	34%
257	29%	47%	44%	33%	17%	35%	34%	32%	39%	25%	33%
258	27%	47%	44%	33%	17%	35%	34%	30%	34%	27%	33%
259	29%	47%	44%	37%	17%	37%	34%	30%	39%	27%	34%
260	29%	49%	42%	35%	20%	37%	34%	30%	39%	25%	34%
261	29%	47%	44%	35%	17%	33%	34%	30%	36%	25%	33%
262	29%	47%	44%	35%	20%	33%	34%	30%	36%	27%	33%
263	29%	47%	44%	33%	22%	33%	34%	30%	39%	27%	34%
264	29%	49%	44%	35%	20%	33%	34%	30%	39%	25%	34%
265	29%	49%	49%	35%	20%	33%	34%	30%	39%	27%	34%
266	29%	47%	47%	35%	22%	33%	34%	30%	41%	25%	34%
267	31%	47%	49%	35%	20%	33%	34%	30%	36%	25%	34%
268	31%	49%	47%	33%	24%	33%	34%	27%	39%	27%	34%
269	31%	49%	49%	28%	24%	30%	34%	27%	39%	27%	34%
270	31%	49%	47%	28%	24%	33%	34%	27%	45%	27%	35%
271	29%	49%	49%	30%	26%	33%	34%	27%	43%	25%	35%
272	29%	44%	49%	28%	26%	33%	32%	27%	41%	25%	33%
273	29%	42%	49%	28%	26%	33%	32%	27%	45%	25%	34%
274	29%	42%	49%	30%	24%	30%	32%	27%	45%	25%	33%
275	31%	42%	47%	28%	26%	30%	32%	30%	43%	25%	33%
276	31%	42%	49%	28%	24%	33%	32%	30%	43%	25%	34%
277	29%	42%	49%	28%	26%	30%	32%	30%	45%	27%	34%
278	31%	42%	49%	28%	26%	33%	32%	27%	43%	27%	34%

279	29%	44%	47%	28%	24%	33%	34%	25%	45%	27%	34%
280	31%	44%	47%	28%	24%	30%	30%	27%	43%	27%	33%
281	31%	44%	47%	28%	26%	30%	30%	27%	45%	32%	34%
282	29%	42%	47%	30%	26%	33%	30%	27%	41%	30%	33%
283	31%	44%	49%	33%	28%	30%	30%	27%	39%	30%	34%
284	31%	47%	47%	30%	26%	33%	30%	30%	41%	32%	35%
285	31%	44%	44%	30%	26%	30%	34%	27%	36%	27%	33%
286	31%	42%	49%	30%	26%	30%	34%	27%	41%	32%	34%
287	31%	42%	49%	30%	30%	33%	36%	27%	39%	30%	35%
288	31%	44%	47%	33%	28%	33%	34%	30%	41%	25%	35%
289	29%	47%	49%	30%	28%	33%	34%	27%	41%	32%	35%
290	31%	44%	47%	30%	26%	33%	34%	30%	39%	30%	34%
291	29%	44%	47%	30%	26%	35%	34%	27%	39%	30%	34%
292	31%	44%	44%	30%	28%	33%	36%	27%	41%	30%	35%
293	31%	44%	47%	33%	24%	33%	34%	27%	39%	25%	34%
294	31%	44%	47%	30%	26%	33%	36%	30%	39%	27%	34%
295	31%	44%	49%	30%	24%	33%	36%	30%	39%	30%	35%
296	29%	44%	49%	33%	26%	33%	36%	30%	41%	30%	35%
297	31%	42%	49%	33%	26%	33%	34%	27%	39%	27%	34%
298	31%	44%	49%	30%	24%	33%	34%	30%	36%	30%	34%
299	31%	42%	49%	30%	24%	33%	36%	27%	36%	25%	33%
300	31%	44%	49%	30%	28%	33%	34%	27%	32%	27%	34%
301	31%	42%	47%	30%	26%	33%	32%	27%	32%	30%	33%
302	31%	42%	49%	33%	26%	33%	34%	27%	34%	30%	34%
303	31%	42%	49%	33%	26%	33%	36%	25%	32%	30%	34%
304	31%	44%	49%	30%	26%	35%	34%	25%	34%	27%	34%
305	31%	42%	49%	30%	26%	35%	34%	27%	32%	27%	33%

306	31%	44%	47%	30%	26%	33%	34%	25%	32%	30%	33%
307	31%	44%	49%	30%	26%	33%	34%	23%	32%	25%	33%
308	31%	44%	49%	33%	24%	33%	36%	23%	32%	25%	33%
309	31%	42%	49%	30%	28%	33%	36%	23%	30%	23%	32%
310	31%	42%	49%	33%	26%	33%	36%	23%	30%	23%	32%
311	31%	40%	47%	35%	24%	30%	36%	25%	30%	25%	32%
312	31%	44%	49%	35%	24%	30%	34%	25%	30%	25%	33%
313	31%	44%	49%	33%	22%	30%	34%	23%	30%	25%	32%
314	31%	42%	47%	35%	24%	30%	36%	23%	27%	25%	32%
315	31%	40%	49%	33%	26%	30%	30%	23%	30%	25%	32%
316	31%	44%	49%	35%	22%	30%	32%	23%	30%	25%	32%
317	31%	42%	49%	33%	22%	30%	32%	25%	27%	27%	32%
318	31%	42%	47%	37%	24%	33%	30%	25%	30%	27%	32%
319	31%	44%	49%	30%	24%	30%	30%	25%	32%	25%	32%
320	31%	42%	49%	33%	24%	28%	32%	27%	30%	25%	32%
321	31%	42%	49%	33%	24%	28%	32%	25%	30%	25%	32%
322	31%	42%	47%	35%	24%	28%	32%	30%	30%	25%	32%
323	31%	42%	47%	33%	24%	28%	32%	30%	32%	25%	32%
324	31%	44%	47%	33%	22%	28%	34%	30%	30%	25%	32%
325	31%	40%	47%	37%	24%	28%	32%	27%	27%	27%	32%
326	31%	36%	47%	37%	24%	28%	32%	27%	27%	25%	31%
327	33%	38%	47%	39%	22%	28%	32%	27%	30%	25%	32%
328	33%	36%	47%	39%	26%	28%	32%	30%	30%	25%	32%
329	33%	38%	47%	37%	26%	28%	32%	25%	30%	20%	32%
330	31%	38%	47%	37%	24%	28%	32%	25%	30%	25%	32%
331	33%	36%	47%	39%	24%	28%	32%	25%	30%	23%	32%
332	33%	38%	42%	39%	24%	28%	32%	27%	32%	23%	32%

333	31%	36%	42%	33%	24%	28%	34%	25%	30%	23%	31%
334	31%	40%	44%	37%	22%	28%	32%	27%	27%	25%	31%
335	31%	38%	47%	39%	26%	28%	32%	27%	30%	30%	33%
336	36%	38%	44%	37%	24%	28%	34%	25%	27%	30%	32%
337	33%	38%	42%	41%	24%	28%	34%	25%	27%	27%	32%
338	31%	38%	44%	39%	26%	28%	30%	25%	32%	23%	32%
339	31%	36%	44%	41%	24%	28%	32%	25%	25%	20%	31%
340	31%	40%	44%	39%	26%	28%	30%	27%	30%	25%	32%
341	31%	38%	44%	39%	26%	28%	32%	25%	30%	20%	31%
342	31%	38%	44%	39%	28%	28%	30%	27%	27%	23%	32%
343	31%	38%	44%	43%	28%	28%	32%	27%	30%	25%	33%
344	31%	36%	42%	41%	28%	28%	30%	27%	27%	25%	32%
345	31%	38%	44%	41%	26%	28%	30%	27%	32%	25%	32%
346	31%	36%	44%	43%	28%	28%	34%	30%	30%	25%	33%
347	31%	31%	44%	41%	28%	28%	34%	27%	30%	25%	32%
348	31%	33%	44%	41%	28%	28%	32%	27%	32%	25%	32%
349	33%	31%	44%	41%	26%	28%	34%	27%	34%	25%	32%
350	31%	38%	44%	41%	26%	28%	34%	27%	32%	25%	33%
351	31%	31%	44%	41%	28%	28%	30%	27%	34%	25%	32%
352	31%	31%	40%	39%	28%	24%	32%	27%	34%	23%	31%
353	31%	40%	40%	43%	28%	24%	30%	27%	30%	25%	32%
354	31%	36%	40%	41%	28%	24%	32%	27%	34%	25%	32%
355	31%	40%	40%	41%	28%	24%	34%	27%	32%	25%	32%
356	31%	38%	40%	41%	30%	22%	30%	27%	32%	20%	31%
357	31%	31%	40%	41%	28%	22%	32%	27%	30%	23%	30%
358	33%	31%	40%	43%	30%	22%	34%	27%	32%	23%	32%
359	33%	29%	40%	41%	28%	22%	32%	27%	27%	23%	30%

360	31%	31%	40%	41%	28%	20%	32%	27%	32%	23%	30%
361	31%	33%	40%	39%	28%	20%	32%	23%	32%	25%	30%
362	31%	31%	40%	41%	26%	20%	32%	20%	32%	23%	30%
363	29%	31%	40%	41%	28%	20%	34%	20%	34%	23%	30%
364	31%	33%	40%	41%	28%	22%	34%	20%	32%	20%	30%
365	31%	40%	40%	41%	30%	22%	34%	18%	32%	20%	31%
366	29%	31%	40%	39%	24%	22%	34%	20%	30%	20%	29%
367	31%	31%	40%	39%	22%	22%	32%	23%	30%	20%	29%
368	29%	31%	40%	39%	20%	22%	32%	20%	34%	25%	29%
369	29%	40%	40%	39%	22%	22%	30%	20%	32%	25%	30%
370	29%	36%	40%	39%	24%	22%	30%	23%	34%	30%	31%
371	27%	36%	40%	39%	22%	22%	30%	18%	32%	27%	29%
372	27%	36%	42%	39%	22%	22%	30%	18%	32%	30%	30%
373	27%	33%	40%	41%	20%	22%	32%	18%	34%	30%	30%
374	27%	33%	40%	41%	22%	22%	27%	23%	32%	30%	30%
375	27%	33%	40%	39%	22%	22%	32%	18%	34%	27%	29%
376	29%	38%	40%	35%	24%	22%	32%	18%	36%	30%	30%
377	27%	38%	40%	37%	22%	22%	32%	18%	36%	25%	30%
378	27%	33%	40%	41%	20%	22%	32%	16%	39%	25%	29%
379	27%	36%	40%	39%	24%	22%	30%	16%	27%	25%	28%
380	27%	33%	42%	41%	26%	22%	32%	16%	27%	25%	29%
381	27%	33%	40%	41%	26%	22%	32%	16%	32%	27%	30%
382	27%	36%	40%	41%	28%	22%	30%	18%	34%	25%	30%
383	27%	33%	40%	43%	26%	22%	30%	18%	36%	25%	30%
384	27%	36%	40%	35%	26%	22%	30%	16%	30%	25%	28%
385	27%	38%	40%	41%	26%	22%	30%	18%	30%	25%	30%
386	27%	36%	40%	41%	26%	22%	30%	18%	30%	25%	29%

387	27%	36%	42%	39%	26%	22%	30%	18%	30%	25%	29%
388	27%	38%	40%	39%	28%	22%	27%	18%	30%	25%	29%
389	27%	36%	40%	39%	26%	22%	30%	18%	30%	25%	29%
390	27%	40%	40%	37%	26%	22%	30%	18%	30%	25%	29%
391	27%	36%	40%	37%	26%	22%	30%	18%	30%	25%	29%
392	27%	38%	40%	37%	26%	22%	30%	20%	32%	25%	30%
393	27%	36%	40%	39%	26%	22%	30%	20%	30%	23%	29%
394	27%	36%	40%	39%	26%	22%	30%	18%	30%	25%	29%
395	27%	36%	40%	39%	28%	22%	30%	20%	30%	25%	30%
396	27%	36%	40%	39%	26%	22%	27%	20%	30%	25%	29%
397	27%	40%	40%	37%	28%	22%	27%	18%	30%	25%	29%
398	27%	36%	42%	37%	26%	22%	30%	20%	30%	25%	29%
399	27%	38%	40%	37%	26%	22%	30%	18%	30%	25%	29%
400	27%	38%	40%	39%	28%	22%	30%	20%	36%	25%	30%
401	27%	40%	40%	35%	26%	22%	30%	20%	30%	25%	29%
402	27%	38%	40%	34%	34%	32%	30%	20%	30%	25%	31%
403			25%	34%	34%	32%	30%	20%	30%	25%	29%

Annex III – Error rate per fold and per category

Fold1		1	2	3	4	5	6	Error Rate
	1	1	0	0	0	0	0	0%
	2	0	11	0	0	4	0	27%
	3	0	0	0	0	1	0	100%
	4	0	2	0	5	4	0	55%
	5	0	3	0	5	9	0	47%
	6	0	0	0	0	0	1	0%
							Average	**38%**

Fold2		1	2	3	4	5	6	
	1	1	0	0	0	0	0	0%
	2	0	7	0	0	3	0	30%
	3	0	0	0	0	0	0	0%
	4	0	0	0	9	2	0	18%
	5	1	3	0	1	16	0	24%
	6	0	0	0	0	0	2	0%
							Average	**12%**

Fold3		1	2	3	4	5	6	
	1	3	0	0	0	0	0	0%
	2	0	10	0	0	5	0	33%
	3	1	0	0	0	0	0	100%
	4	0	0	0	7	2	0	22%
	5	1	1	0	2	9	0	31%

	6	1	0	0	0	2	0	100%
							Average	**48%**

Fold4		1	2	3	4	5	6	
	1	2	0	0	0	0	0	0%
	2	0	5	0	1	7	0	62%
	3	0	0	0	0	2	0	100%
	4	0	1	0	79	5	0	7%
	5	0	2	0	0	15	0	12%
	6	0	0	0	0	2	1	67%
							Average	**41%**

Fold5		1	2	3	4	5	6	
	1	3	0	0	0	0	0	0%
	2	0	12	0	2	4	0	33%
	3	0	0	0	0	1	0	100%
	4	0	0	0	4	2	0	33%
	5	1	4	0	4	10	0	47%
	6	0	0	0	0	0	0	0%
							Average	**36%**

Fold6		1	2	3	4	5	6	
	1	3	0	0	0	0	0	0%
	2	0	16	0	0	2	0	11%
	3	0	0	0	0	0	0	0%
	4	0	0	0	3	2	0	40%
	5	0	7	0	2	10	0	47%
	6	0	0	0	0	0	1	0%
							Average	**16%**

Fold7		1	2	3	4	5	6	
	1	3	0	0	0	3	0	50%
	2	0	11	0	2	3	0	31%
	3	0	0	0	0	0	0	0%
	4	0	0	0	4	4	0	50%
	5	0	4	0	0	12	0	25%
	6	0	0	0	0	0	0	0%
							Average	**26%**

Fold8		1	2	3	4	5	6	
	1	27	0	0	0	0	0	0%
	2	1	146	0	0	4	0	3%
	3	1	0	1	0	3	0	80%

4	1	0	0	78	5	0	7%
5	0	6	0	0	164	0	4%
6	2	0	0	0	5	5	58%
						Average	**25%**

Fold9		1	2	3	4	5	6	
	1	3	0	0	0	0	0	0%
	2	0	14	0	2	3	0	26%
	3	1	0	0	0	0	0	100%
	4	0	0	0	8	2	0	20%
	5	0	1	0	1	10	0	17%
	6	1	0	0	0	0	0	100%
							Average	**44%**

Fold10		1	2	3	4	5	6	
	1	2	0	0	0	0	0	0%
	2	0	10	0	0	4	0	29%
	3	0	0	0	0	0	0	0%
	4	0	0	0	8	3	0	27%
	5	1	3	0	0	14	0	22%
	6	0	0	0	0	0	0	0%
							Average	**13%**