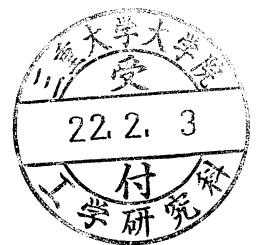


An Impact of Linguistic Features on Automated Classification of OCR Texts

A Thesis Submitted to Mie University in Partial Fulfillment of
the Requirements for the Degree of

Master of Engineering in Information Engineering

February, 2010
Graduate School of Engineering
Engineering Division
Mie University



Moshi Gudila Paul

Abstract

Optical Character reader (OCR) systems can be used in digitizing print documents. OCR texts are generated in the process of digitizing print documents. Usually these texts need to be indexed and organized to simplify their access and retrieval. This can be done by the use of automatic classification techniques. However it is currently impossible for OCR technology to recognize all characters with an accuracy of 100%.

Therefore the first objective of this research is to investigate how to automate the classification of these OCR texts effectively. To solve the problem of high dimensional feature space we reduce the dimensionality with PCA. In connection to that we also adopted discriminant analysis method which reduced dimensionality and extracted more informative features to improve textual data separability. Conventionally a number of researchers applied PCA to reduce the dimensionality but since PCA is an unsupervised technique it ignores category specific information. i.e. it seeks direction that are efficient for representation and does not include category information of the data. For example when first component is chosen along the largest variance line, category will strongly overlap.

In order to overcome this shortcoming we performed canonical discriminant analysis (CDA). CDA seeks direction that is efficient for discrimination i.e. it makes use of category information as it find the projection such that the instances of different categories have maximum separation between each other and at the same time it insures that instances in same category cluster closely together. But since CDA tends to have a singularity problem of the within-class covariance matrix due to higher dimensionality compared to the sample size we therefore experimentally study the combination of PCA and CDA (PCA+CDA) algorithm.

Our approach found out that PCA+CDA algorithm improved classification performance only when we used a weak classifier (in our case k NN) and PCA outperformed PCA+CDA algorithm when strong classifier was used (in our case SVM).

Furthermore it is not known whether part of speech (POS) analysis contributes to proper OCR texts representation in a discriminative way. Conventionally, the *bag-of-words* approach is used in OCR text classification. Therefore our second objective of this work is to experimentally evaluate POS analysis on OCR texts to formulate an informative feature set. Empirical results indicate that the combination of suitably selected POS improved classification performance of OCR texts.

Acknowledgments

First and foremost I offer my sincerest gratitude to my supervisor Professor Fumitaka Kimura and his Associate Professor Tetsushi Wakabayashi and Dr. Wataru Ohyama, for their guidance and supervision. They made this work possible. I am grateful to the many students in Human Interface Laboratory, without them this research would not have been possible, and their tolerance, good humor, and insight contributed much. I hope they felt as rewarded as I did for their time and effort. Each of the members of my Dissertation Committee has provided me extensive personal and professional guidance and taught me a great deal about both scientific research and life in general. I would especially like to thank Dr. Lazaro S. P. Busagala as he has taught and guided me a lot to achieve this work.

Nobody has been more important to me in the pursuit of this project than the members of my family. I would like to thank my parents, whose love and guidance are with me in whatever I pursue. They are the ultimate role models. Most importantly, I wish to thank my loving husband, Dr. Dida A. Francis, for his unwavering support and understanding during the many hours I dedicated to achieving this milestone in my life. And also many thanks goes to my wonderful son, Albert, who provides me with unending inspiration. This work would not have been possible without the financial support of the Mie University, and JASSO scholarship. I kindly thank them for their financial support.

Contents

Abstract	i
Acknowledgments	ii
Chapter 1 Introduction	1
1.1 Background	1
Chapter 2 Literature Review	3
2.1 Document Representations	3
2.2 Dimension Reduction	3
2.3 Learning and Classification methods	5
2.4 Performance measures	6
Chapter 3 Experiments	8
3.1 Data for experiments	8
3.2 Simulations and Pre-processing	8
3.3 Automatic text classification(ATC) procedures	8
3.4 Parts Of Speech Analysis (POSA)	13
3.5 Combination of suitable POS	15
3.6 Feature transformation in POSA	16
Chapter 4 Results	17
4.1 Results of classification of PCA, PCA+CDA reduced features	17
4.2 Results of Parts of Speech Analysis (POSA)	21
Chapter 5 Conclusion	26
Chapter 6 Related Works and future work	27

List of Tables

2.1	Contingency table for categorization.	7
3.1	Example of OCR ASCII text of the image in Figure 1.	9
3.2	Example of the tagging process of the OCR text at 130dpi of table 3.1.	15
4.1	Total and average number of each Parts of Speech used in the experiment. The abbreviations ANN, ANP, ANV AND and ANJ means the average number of nouns, pronouns, verbs, adverbs and adjectives per document respectively.	23

List of Figures

3.1	Example of text image at 130dpi.	9
3.2	Automatic text classification procedures	10
4.1	Empirical results of kNN classifier.	18
4.2	Empirical results of SVM classifier.	19
4.3	Classification performance after power transformed relative frequency (RFPT) features vs Word recognition rate F-measure of the data at 300dpi, 200dpi, 150dpi, 130dpi and 100dpi.	20
4.4	Effects of OCR errors in relation to Word Recognition (Micro-average F_1 in %) and Classification Performance (micro-average F_1 in %).	23
4.5	Empirical results of with and without POSA using kNN classifier.	24
4.6	Empirical results of with and without POSA using SVM classifier.	25

Chapter 1

Introduction

1.1 Background

In recent years Text Categorization (also known as Text Classification (TC) or Text Spotting and sometimes refers to Automatic Text Classification (ATC)) became a major sub-field of information systems due to the gradual increase of interest since the early 60's. In the beginning of TC technology, the approach known as knowledge engineering (KE) was used. KE involves the training professionals to define a set of classification rules manually which is very expensive in terms of time, knowledge experts and computing memory. In the late 80's TC became a fully blossomed research field which has delivered efficient, effective and overall promising solutions that have been used in tackling a wide variety of real world application domains [8]. This field became of interest because recent TC approach which uses the machine learning (ML) techniques have reached accuracy levels that outperforms the performance of trained professionals. The ML technique involves classification systems/algorithms that provide examples which are automatically learned by classifiers. This technique saves time and is manage-ably flexible.

The means of information exchange has been changing from print information to digital data which is faster, more flexible and easier to access. Increased availability of digitized information creates a room for flexible and convenient access [8]. This calls for the need to digitize print texts [14]. Digitization may involve creating digital images by a scanner and then generate ASCII texts by an Optical Character Reader (OCR) systems. However it is known that it is impossible for OCR systems to give 100% accuracy. In other words OCR texts usually contain errors due to misrecognized characters. Digitized texts need to be indexed and organized in databases so that they can be accessed easily through the Web or offline. Automatic text classification (ATC) is an important tool in indexing and organizing these documents effectively. ATC is a task of automatically assigning a set of documents to appropriate categories [8], [23].

In information retrieval for example, OCR errors affect the system's reliability and efficiency. Many OCR error removal systems [9] focus on pages with the combination of texts and images. When dealing with only texts, application of these OCR error removal systems may remove important information of the text. Therefore there is a great need of investigating these OCR texts without applying any error removal system. With the presence of these OCR errors it is important to explore different approaches so as to improve classification. Also it is essential to choose proper syntactic word categories that represent the document to train a classifier so that the classifier will be able to effectively classify new OCR texts.

Some achievements in ATC has been reported but the results show that there is still a great

need to improve the performance of classification systems. Some of the reported problems in ATC field are high dimensional feature space and poor selection of features in the presence of OCR errors.

Therefore, the first objective of this research is to investigate how to automate the classification of these OCR texts effectively. To solve the problem of high dimensional feature space, we reduced the dimensionality with PCA. In connection to that we also adopted discriminant analysis method which reduced dimensionality and extracted more informative features to improve textual data separability [13].

Conventionally a number of researchers applied PCA as a dimension reduction tool [26]. But since PCA is an unsupervised technique it ignores category specific information [25] i.e. it seeks direction that are efficient for representation and does not include category information of the data. For example when first component is chosen along the largest variance line, category will strongly overlap.

In order to overcome this shortcoming we performed canonical discriminant analysis (CDA). CDA seeks direction that is efficient for discrimination i.e. it makes use of category information as it finds the projection such that the instances of different categories have maximum separation between each other and at the same time it insures that instances in same category cluster closely together. But since CDA tends to have a singularity problem of the within-class covariance matrix due to higher dimensionality compared to the sample size we therefore experimentally study the combination of PCA and CDA (PCA+CDA algorithm) [13].

Our approach found out that PCA+CDA algorithm improved classification performance only when we used a weak classifier (in our case kNN) and in contrast, PCA outperformed PCA+CDA algorithm when strong classifier was used (in our case SVM).

The second objective was to find out the suitable feature selection in the presence of OCR errors in order to improve classification performance of ATC of OCR texts.

Conventionally, the *bag-of-words* (BOW) approach is used in generating features to represent OCR texts [5], [15]. Usually BOW does not take into considerations of syntactic word categories. Therefore BOW includes all words even those which can cause over fitting of document categories.

Unlike the conventional approach we evaluate how linguistic features improve classification performance of OCR texts based on parts of speech analysis(POSA). We used the absolute frequency of syntactic word categories in this feature set and transformed them into relative frequency and lastly we applied power transformation[13]. We found out that our approach improved classification effectiveness.

The rest of the paper is organized as follows. Chapter 2 explains Literature review. In Chapter 3 experiments are described. Chapter 4 discusses the experimental results. We finally draw conclusions in chapter 5. In Chapter 6, a brief survey of related works is discussed. We also present future research directions.

Chapter 2

Literature Review

2.1 Document Representations

Document indexing means preparing the raw document collection into an easily accessible representation of documents. This transformation from a document text into a representation text is called indexing the documents. The most important point in indexing is to choose words (richer features) for indexing that can differentiate a given document or category of documents from all others in the same collection. There are various ways of indexing discussed in literature such as Bag-of-Words (BOW), parts of speech (POS) information, phrases, stemming, term weighted vectors, etc.

2.2 Dimension Reduction

2.2.1 Principal Component Analysis (PCA)

Classification in high dimensional data is extraordinarily difficult because of the curse of dimensionality. Therefore in this work we used PCA to reduce dimensionality as well as solving the problem of over-fitting. Dimension reduction also saves computational time and resources. PCA can be defined as an orthonormal transformation of the data, retaining only significant components. It transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components-PC (projection to the eigenvector with the largest eigenvalue). It reveals the internal structure of the data in a way which best explains the variance in the data.

Given a set of training documents $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ the total Covariance matrix(C) of the training sample is computed by;

$$C = \frac{1}{N} \sum_{\mathbf{x} \in \mathcal{X}} (\mathbf{x} - \bar{\mathbf{m}})(\mathbf{x} - \bar{\mathbf{m}})^T \quad (2.1)$$

$$\bar{\mathbf{m}} = \frac{1}{N} \sum_{\mathbf{x} \in \mathcal{X}} \mathbf{x} \quad (2.2)$$

Where $\bar{\mathbf{m}}$ is the mean feature vector of a training sample. N is the total number of the training documents.

Eigenvectors ϕ_i and eigenvalues λ_i ($i = 1, 2, \dots, n$) of total covariance matrix were obtained and sorted in decreasing order, given that eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$.

$$C\Phi_i = \lambda_i\Phi_i; \quad (i = 1, 2, \dots, n) \quad (2.3)$$

Eigenvectors of covariance matrix corresponding to larger eigenvalues were chosen to be the principal components (PC) computed as:

$$z_i = \Phi_i^T \mathbf{x} \quad (i = 1, 2, \dots, k) \quad (2.4)$$

The first principal component (PC) z_1 accounts for as much of the variability in the data as possible and each succeeding PC z_2, z_3, \dots, z_k accounts for as much of the remaining variability as possible. The reduced dimension of feature vector x compose of m-dimension ($k \leq n$)

2.2.2 PCA+CDA Algorithm

As it has been mentioned before, PCA does not consider the between class and the within class scatter matrices and hence it ignores category specific information [13]. Due this drawback of PCA, Canonical discriminant analysis (CDA) was applied on the features which were first reduced by PCA so as to avoid singularity problem of the within-class covariance matrix. CDA was also adopted by other researchers [24]. The combination of PCA and CDA is abbreviated as PCA+CDA. In the experiment CDA was applied on 500 principal components and 1000 principal components separately in order to observe the effect of dimensionality on this technique.

CDA preserves cluster structure by maximizing the scatter between classes (clusters) while minimizing the scatter within classes(cluster). The within-class scatter matrix S_w and the between-class scatter matrix S_b are defined as follows:

$$S_b = \sum_{i=1}^C \frac{N_i}{N} (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \quad (2.5)$$

$$S_w = \frac{1}{N} \sum_{i=1}^C \sum_{\mathbf{x} \in \mathcal{X}_i} \frac{N_i}{N} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \quad (2.6)$$

whereby, mean vector for each class is,

$$\mathbf{m}_i = \frac{1}{N_i} \sum_{\mathbf{x} \in \mathcal{X}_i} \mathbf{x} \quad (2.7)$$

N refer to the number of documents and \mathcal{X}_i is the set of text sample in a particular class w_i .

Eigenvectors vectors are calculated by solving the following equation;

$$S_b \Phi = \lambda_i S_w \Phi \quad (2.8)$$

Canonical discriminants (CD) are calculated in the same way as PC by equation (2.4).

2.3 Learning and Classification methods

2.3.1 k Nearest Neighbors (k NN)

Various classification methods have been proposed in the literature [3], [6], [8]. Among other classifiers, k nearest neighbors (k NN) method is one of the best performers. The k NN algorithm relies on the concept that given a test document x , the system finds the k nearest neighbors among the training documents to estimate its *a posteriori* probability $P(\omega_j|\mathbf{x})$ for each category [11].

Moreover, k NN can easily handle both multi-class and multi-label problems simultaneously as compared to other classification methods. Since the Reuters-21578 is a multi-class problem, hence k NN was used in the classification process.

We use an improved k NN learning method. The conventional k NN learning method has drawbacks. For example, it assumes that all examples in the set of k NN $D_k \subseteq D$ have equal importance in predicting the class of the incoming document. Thus it results in giving equal weight even to those instances that are far from the incoming pattern. Consequently, local-category information for correct prediction of a class can be distorted.

In an attempt to remove this drawback Lim [4] proposed a technique for weighting document similarities defined by

$$Z(\mathbf{x}, \omega_j) = \sum_{D_i \in D_k}^k \text{sim}(\mathbf{x}, D_i) y(D_i, \omega_j), \quad (2.9)$$

where $y(D_i, \omega_j) \in \{0, 1\}$ is a discrete function that refers to the classification of training document D_i belonging to a specific category such that $y(D_i, \omega_j) = 1$ for YES and $y(D_i, \omega_j) = 0$ for NO. The $\text{sim}(\mathbf{x}, D_i)$ is the similarity between the test document and the training document. This can be called the similarity weighted function (SWF). In general terms it can be called weighted metric function (WMF).

However, one can note that expression (2.9) can still result in noises and difficulties in determining the threshold. To solve these problems, we propose a normalized-weighted metric (NWM) function. The function can use a distance or similarity measure. Let our metric be similarity measure, $\text{sim}(\cdot)$ as in (2.9). NWM can be defined as

$$Z'(\mathbf{x}, \omega_j) = \frac{Z(\mathbf{x}, \omega_j)}{\sum_{D_i \in D_k} \text{sim}(\mathbf{x}, D_i)}. \quad (2.10)$$

This can be understood as the normalization of (2.9). This was used instead of the conventional voting strategy. Expression (2.10) has a property such that $(0 \leq Z'(\mathbf{x}, \omega_j) \leq 1)$. In other words probabilistic value will always be obtained. In doing so, the threshold will always be in the range of 0 to 1.

In general terms, $\text{sim}(\mathbf{x}, D_i)$ can be replaced with other metrics such as distance metrics. For example Euclidean distance can be used instead of cosine similarity. However the reader should note that the cosine similarity function was used in the experiments.

2.3.2 Support Vector Machines (SVM_s)

Support Vector Machines (SVMs) is the machine learning method which finds the optimal hyperplane with maximum margins. The nearest patterns to the decision boundaries are the support vectors. We used the SVM^{Light} package [2] in the experiments. We divide each classification task into C binary classification problems and adopt the one against the rest strategy. We mostly report results from linear kernel. This is because a considerable number of studies have shown that the linear kernel outperforms non-linear ones in ATC. For more theory details about SVM_s refer to [17], [2], [11].

2.4 Performance measures

There are various methods to judge the effectiveness of the TC classifiers built using the machine learning approach. The experimental evaluation of a classifier usually measures its effectiveness rather than its efficiency. In this case effectiveness means the ability of a classifier to take the right classification decision. Contingency table is used to evaluate the performance of a classifier for each category. The performance measures Precision (P), Recall (R) and f-measure (F_1) can be computed from the contingency table 2.1. Many researchers have adopted these measures in previous [19], [8] and [10].

In general there are two methods of computing recall and precision. These are the macro averaging and the micro averaging methods. Macro averaging gives equal weight to every *category* while micro averaging gives equal weight to every *document*. Literature [8] gives more details in these strategies. Precision is the percentage of system assigned categories that also appeared in the expert (human) indices.

In macro-averaging method precision P_i for each category is calculated as follows;

$$P_i = \frac{TP_i}{TP_i + FP_i} \quad (2.11)$$

The averaged precision P for all categories will be;

$$P = \frac{\sum_{i=1}^C P_i}{C} \quad (2.12)$$

The micro-averaging method can be computed as follows

$$P = \frac{\sum_{i=1}^C TP_i}{\sum_{i=1}^C (TP_i + FP_i)} \quad (2.13)$$

Recall is percentage of expert (human) assigned categories that the system also produced.

In macro-averaging method recall R_i for each category is calculated as follows;

$$R_i = \frac{TP_i}{TP_i + FN_i} \quad (2.14)$$

The averaged recall R for all categories will be;

$$R = \frac{\sum_{i=1}^C R_i}{C} \quad (2.15)$$

The micro-averaging method can be computed as follows

$$R = \frac{\sum_{i=1}^C TP_i}{\sum_{i=1}^C (TP_i + FN_i)} \quad (2.16)$$

The f-measure (F_1) is the special form of the \mathcal{F}_β measure, when $\beta = 1$. \mathcal{F}_β is defined as the harmonic mean of recall and precision.

Category w_i	Expert Decisions		
		Yes	No
Classifier Decisions	Yes	TP_i	FP_i
	No	FN_i	TN_i

Table. 2.1: Contingency table for categorization.

TP/ True Positive: When the case was positive and predicted positive i.e. decision of an expert and that of a classifier is **Yes**.

TN/ True Negative : When the case was negative and predicted negative i.e. both classifier and expert's decisions are **No**.

FP/ False Positive: When the case was negative but predicted positive i.e. classifier's decision is **Yes** and the expert's decision is **No**.

FN/ False Negative: When the case was positive but predicted negative i.e. the decision of a classifier is **No** while expert's is **Yes**.

Chapter 3

Experiments

3.1 Data for experiments

Documents were obtained from Reuters 21578 collection of English news wire articles which is composed of 21578 articles manually classified into 135 categories. Many text classification methods have been tested using this corpus [13], [27]. We used articles from ModApte split belonging to the largest 10 categories(i.e. acq,crude, earn, grain, interest, money-fx, money-supply, ship, sugar, trade).

In this work 6491 articles were used as train data set and 2545 articles were used as test data set.

3.2 Simulations and Pre-processing

Test data documents from Reuters collection were printed out in order to simulate the process of generating OCR texts. The print texts were digitized using a scanner into images of resolution of 100, 130, 150, 200 and 300 dpi and they were converted into ASCII texts by e-typist OCR software [5]. These OCR texts were used as test data. Figure 3.1 shows the example of the text image scanned at 130 dpi. Table 3.1 shows the OCR text generated from the text image in Figure 3.1. The misrecognized words in boldface are called OCR errors. Feature vectors were generated as explained in the section 3.3.1 and transformed as in section 3.3.2. Before generating feature vectors, stop words (functional words, general words) were removed with reference to a stop list of 572 words which is used by typical retrieval system SMART [23] for retrieving English documents. Also words which appeared less than 6 in all training data were removed before generating word list. By removing stop words, the problem of over-fitting is also reduced.

PCA is explained in section 2.2.1. The obtained data followed the Automatic text classification procedures as described in the following section.

3.3 Automatic text classification(ATC) procedures

In our work ATC procedure is composed of five steps; feature vector generation, feature transformation, dimension reduction, learning and classification. Word list or vocabulary list were generated according to train data. Figure 3.2 shows the ATC procedures. Feature vectors of both train and test data were generated and then they were transformed. In feature transformation, the absolute word frequency were normalized to relative word frequency (RF) and then power

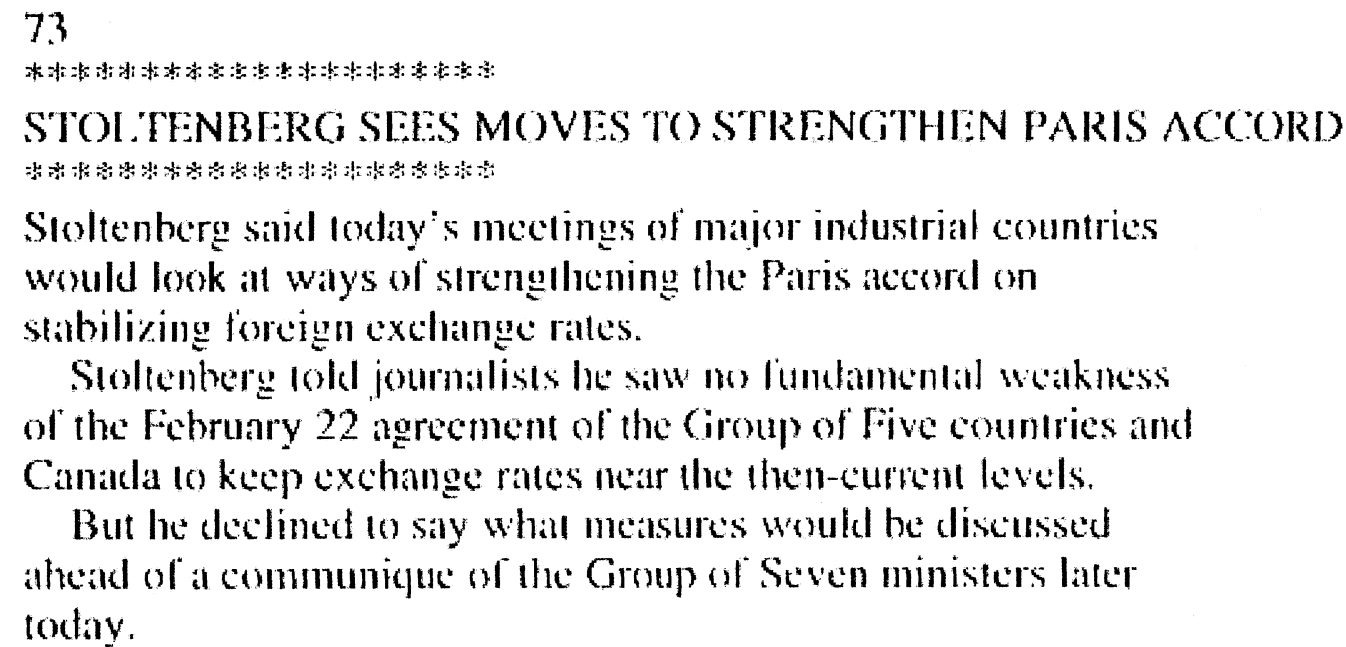


Fig. 3.1: Example of text image at 130dpi.

<p>STOL NRf RC SEES MOVES TO STRENGTHEN PARIS ACCORD</p> <p>Stoltenberg said today's meetings of major i1 st 'd countriev would look at ways of strengthening the Paris accord on stabilizing foteien exchange rates.</p> <p>Stoltenberg told journalists he saw no fundamental weakness of the February 22 agreement of the Group of Five countries and Canada to keep exchange rates near the then-current levels. But he declined to say what measures would he discussed ahead of a communiyne of the Group of Seven ministers later today</p>
--

Table. 3.1: Example of OCR ASCII text of the image in Figure 1.

transformed (RFPT) [13]. For comparison reasons, dimensionality was reduced using two algorithms separately i.e. PCA only and PCA+CDA algorithm. In learning, the weight vectors are calculated from the training data so that the SVM classifier can later classify test data. In this work we also have used k-NN method.

3.3.1 Feature vector generation

The first step in TC is to transform documents, which typically are strings of characters, into a representation suitable for the learning algorithm and the classification task. Feature generation represents textual data in such a way that the learning algorithm can recognize them easily. Vector model approach using words or terms is one of the commonly used method to generate features from textual data. In this method every document is converted into word tokens which

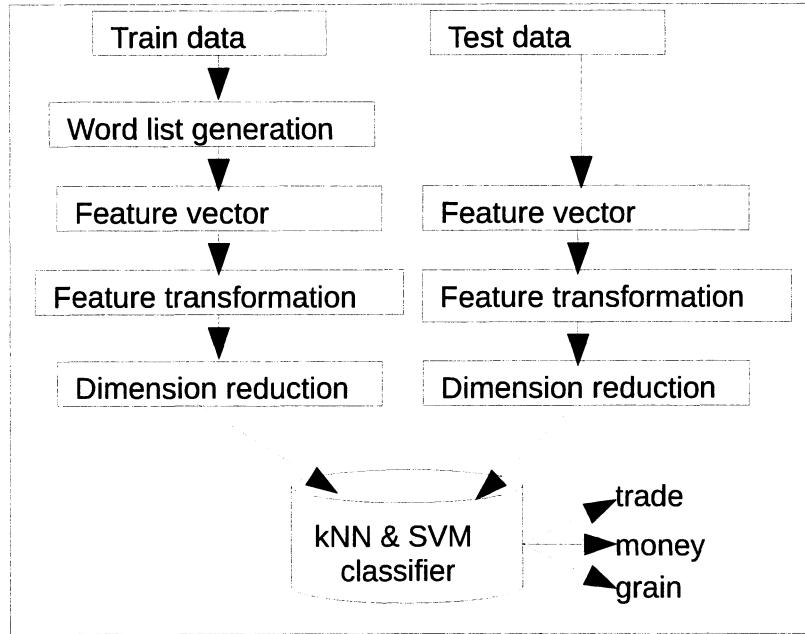


Fig. 3.2: Automatic text classification procedures

form the feature vector i.e. Each document is represented by a vector of words. The example below illustrates document representation using the feature vector model. Assume that the feature vectors are defined as $\mathbf{x} = [x_1, \dots, x_n]^T$, where x is the term frequency in every vector formed from each document. Assume that the following three documents represent a text collection.

1. This is a book.
2. The economy is in recession.
3. She is reading a book.

The vocabulary list generated in alphabetical order will be as follows { a, book, economy, in, is, reading, recession, she, the, this } From this vocabulary list the feature vector obtained from each document will be:

1. $[1100100001]^T$
2. $[0011101010]^T$
3. $[1100110100]^T$

Generated feature vectors were transformed in order to improve the learning ability of the classifiers. Details about feature transformation will be discussed in the following section.

3.3.2 Feature transformation

Feature transformation is a process through which a new set of features is created. In this work feature transformation refers to transforming absolute term frequency (AF) to relative term frequency (RF) and then power transformation (PT) [13].

Absolute term Frequency (AF)

In definition the absolute frequency is the total amount of occurrences of one variable. Let us consider a set of n sample texts, $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ with N -dimensional text space. Assume that every textual document belongs to one of the C classes $\{w_1, w_2, \dots, w_C\}$. Each text can be represented as a feature vector,

$$\mathbf{x}_k = [x_1 x_2 \dots x_N]^T \quad (3.1)$$

whereby, N is dimensionality, x_i is the frequency value of the i^{th} word and T refers to the transpose of a vector. These feature vectors are termed as absolute term frequencies. AF tends to depend on text length leading into lower classification performance. This is due to the fact that text length differs within the same class of documents which makes the learning process difficult [13] [8].

Normalization to Relative Frequency (RF)

The relative frequency is the absolute frequency divided by the total amount of occurrences of ALL variables. In order to solve the problem of dependency on text length, the length variation in absolute term frequency need to be reduced. Therefore absolute term frequency were transformed to relative term frequency as follows:

$$y_i = \frac{x_i}{\sum_{j=1}^n x_j} \quad (3.2)$$

x_i is the AF of the i^{th} word and n is the number of different words.

By transforming AF to RF, the length of documents are normalized and hence reduces the learning load of classifiers and improves classification accuracy [13].

Power transformation (PT)

This is a technique of performing power transformation on AF x_i features.

$$z_i = x_i^v \quad (0 < v < 1) \quad (3.3)$$

Power transformed relative frequency (RFPT)

It has been observed that even after obtaining the RF, the sample distribution for the documents may still be skewed. Therefore there is a need to make the distributions Gaussian-like [1] so as to be able to use them in parametric classifiers such as linear or quadratic ones. Power transformation was applied in order to obtain this effect. Power transformation is one of the variable transformations that maps data from one space to another using power functions. It is expressed as;

$$z_i = y_i^v \quad (0 < v < 1) \quad (3.4)$$

by substituting equation (3.2) into (3.4) then

$$z_i = \left(\frac{x_i}{\sum_{j=1}^n x_j} \right)^v, \quad (0 < v < 1) \quad (3.5)$$

Features generated by equation (3.5) are called relative frequency with power transformation (RFPT). It improves the symmetric of the distribution of the frequency $y_i > 0$ which is asymmetric near the origin and hence the distribution becomes Gaussian-like. In this experiment we used the power of 0.5.

3.4 Parts Of Speech Analysis (POSA)

Parts of Speech (POS) is the traditional term for the categories into which words are classified according to their functions in sentences. It is also known as word class or syntactic category. English language words are classified into eight POS. POSA in ATC is the analysis of the effectiveness of some POS in text classification. In this work we focused on five parts of speech- nouns, pronouns, verbs, adverbs and adjectives.

3.4.1 Pre-processing in POSA

Test data documents from Reuters collection were printed out in order to simulate the process of generating OCR texts. The print texts were digitized using a scanner into images of resolution of 130, 200 and 300 dpi and they were converted into ASCII texts by e-typist OCR software [5]. These OCR texts were used as test data. Figure 3.1 shows the example of the text image scanned at 130 dpi. Table 3.1 shows the OCR text generated from the text image in Figure 3.1. The words in boldface are the OCR errors.

In many text classification approaches, removal of stop words (functional words, general words) is carried out in data pre-processing before feature generation. But in order to judge the effectiveness of each POS contributing to text classification, in this approach, stop words were not removed. This decision lowered classification performance but enhanced coherent conclusions.

Each document was then tokenized by the Penn Treebank Project tokenizer in order to remove the adjoining words. The removal of the adjoining words before tagging improved the accuracy of the tagging process. Then tree-tagger [7] was used to identify the Parts-of-speech tags. The desired linguistic patterns (POS) were extracted. We experimented on different POS combination in order to find out the combination that will give higher results in OCR texts of different dpi values. The reason for using various dpi values include investigating the impact of OCR errors.

Dimensionality was reduced by Principal Component Analysis(PCA) through which we retain a set of principle components with highest variance.

3.4.2 POS information extraction

In this section we describe POSA and combination methods. Part-of-speech (POS) tagging of news articles is done. All POS elements are extracted based on the tree-tagger^{*1}. Table 3.2 shows the example of the POS-tagging process of the OCR text in table 3.1.

Firstly we found out which POS contributed more in classification of OCR texts as explained in section 3.4.3. Secondly, the POS which describe the category more, are extracted and used as our feature set in section 3.5. Five POS which are nouns, pronouns, verbs, adjectives and adverbs were used. Using the method explained in Section 3.3.1 feature vectors are constructed.

3.4.3 Selection of suitable POS

We followed two steps in choosing suitable POS combination. Firstly we found out which POS contribute more in classification performance of OCR texts. In this step we removed one group of POS and then evaluate the classification performance to find out the hierarchy of the POS in describing a category's content.

For simplicity let us assume that there are two sets of feature vectors: (1) the term frequency generated using nouns and (2) the feature set generated based on verbs present in a text.

We denote nouns with a superscript u and verb features with a superscript v in equations (3.6) to (3.12). Consequently, we can define the feature vectors as

$$\mathbf{x}^{(u)} = \begin{bmatrix} x_1^{(u)} & x_2^{(u)} & \dots & x_{n_1}^{(u)} \end{bmatrix}^T, \quad (3.6)$$

for the noun features. The verb features can be expressed as

$$\mathbf{x}^{(v)} = \begin{bmatrix} x_1^{(v)} & x_2^{(v)} & \dots & x_{n_2}^{(v)} \end{bmatrix}^T. \quad (3.7)$$

Let us denote *bag-of-words* feature set as \mathbf{A} and verb features as $\mathbf{x}^{(u)}$ the remaining feature vectors can be defined as

$$\mathbf{R} = \mathbf{A} \ominus \mathbf{x}^{(u)} \quad (3.8)$$

^{*1} <http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger>

WORD	TAG	LEMMA
STOL	NN	unknown
NRf	NP	unknown
RC	NP	RC
SEES	NP	Sees
MOVES	NP	unknown
TO	TO	TO
STRENGTHEN	NP	unknown
PARIS	NP	Paris
ACCORD	NP	Accord
Stoltenberg	NP	Stoltenberg
said	VBD	say
today	NN	today
's	POS	's
meetings	NNS	meeting
of	IN	of
major	JJ	major
il	JJ	unknown
st	NNS	unknown
,	“	,
d	SYM	d
countrierv	NN	unknown
would	MD	would
look	VB	look
at	IN	at
ways	NNS	way
interest	NN	interest

Table. 3.2: Example of the tagging process of the OCR text at 130dpi of table 3.1.

Equation (3.8) can be used in removing verbs as well;

$$\mathbf{R} = \mathbf{A} \ominus \mathbf{x}^{(v)} \quad (3.9)$$

Equation (3.8) can be used in removing other POS elements such as pronouns, adjectives and adverbs.

3.5 Combination of suitable POS

Secondly, the POS which describe the category more, are combined and their effect in classification is observed. The combination which suitably improve OCR text classification is used as a feature set.

The combination of noun and verb feature vectors (represented as in equation (3.6) and (3.7) respectively) can be defined as

$$\mathbf{Q} = \mathbf{x}^{(u)} \oplus \mathbf{x}^{(v)} \quad (3.10)$$

$$= \left[x_1^{(u)} \dots x_{n_1}^{(u)}, x_1^{(v)} \dots x_{n_2}^{(v)} \right]^T. \quad (3.11)$$

Therefore if we denote pronoun, adjective and adverb features with a superscript r , j and d respectively, then the combination of features used in this experiment can be expressed as;

$$\mathbf{Q} = \mathbf{x}^{(u)} \oplus \mathbf{x}^{(v)} \oplus \mathbf{x}^{(r)} \oplus \mathbf{x}^{(j)} \oplus \mathbf{x}^{(d)} \quad (3.12)$$

We use this technique to generate an informative features to improve classification performance. Equation (3.5) is applied to the composite feature set \mathbf{Q} to form a normalized and Gaussian-like distribution.

3.6 Feature transformation in POSA

Consider a set of N sample texts, $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ with n -dimensional text space. Assume that every textual document belongs to one of the C classes $\{\omega_1, \omega_2, \dots, \omega_C\}$. Each text can be represented as a feature vector, $\mathbf{x}_k = [x_1 x_2 \dots x_n]^T$, whereby, n is dimensionality (dimension of generated lexicon list), x_i is the AF of parts of speech (POS) which is the frequency value of i^{th} word or POS and T is the transpose of a vector. The AF features were normalized into RF using the equation (3.2) and power transformed using equation (3.5).

We used PCA for dimension reduction and for learning and classification we used the same classifiers (k NN and SVM_s) explained in section 2.3. We did not use PCA+CDA algorithm as our dimension reduction method because we have found out the PCA outperforms PCA+CDA algorithm when strong classifier was used.

Chapter 4

Results

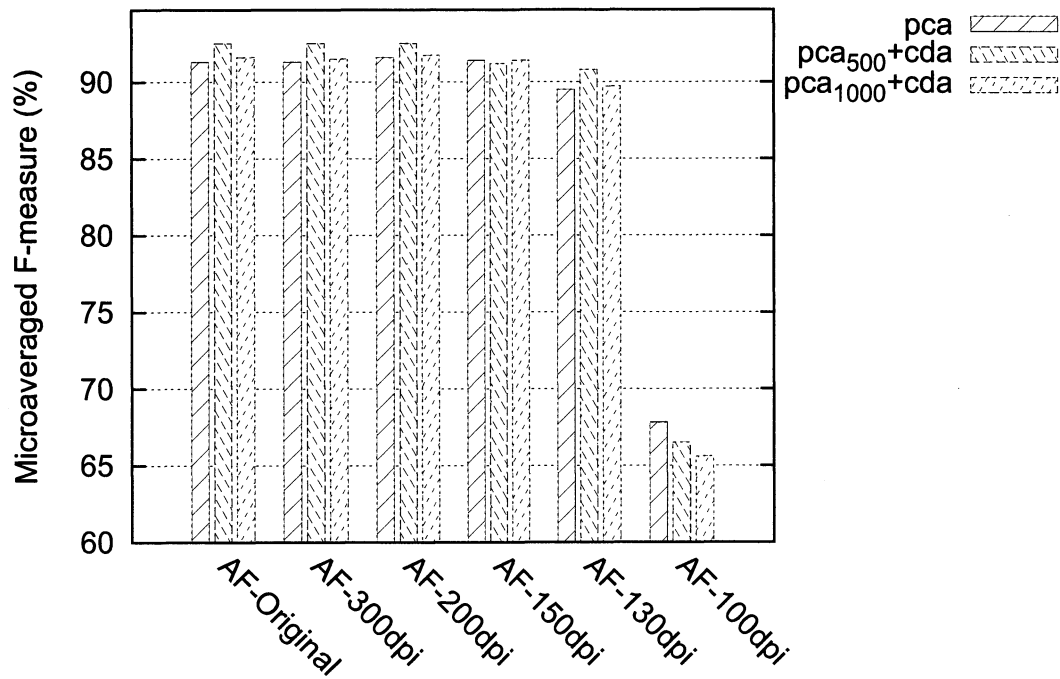
4.1 Results of classification of PCA, PCA+CDA reduced features

The empirical results of k NN on AF features and on RFPT are shown in figure 4.1(a) and 4.1(b) respectively. The comparison of the classification results of original test data and OCR texts at the resolutions of 300dpi, 200dpi, 150dpi, 130dpi and 100dpi values. These figures show the comparison of the classification results using PCA only and PCA+CDA algorithm for feature reduction. There are two sets of results of PCA+CDA algorithm; the results of applying CDA on 500 principal components (PC) ($pca_{500} + cda$) and that of applying CDA on 1000 PC ($pca_{1000} + cda$).

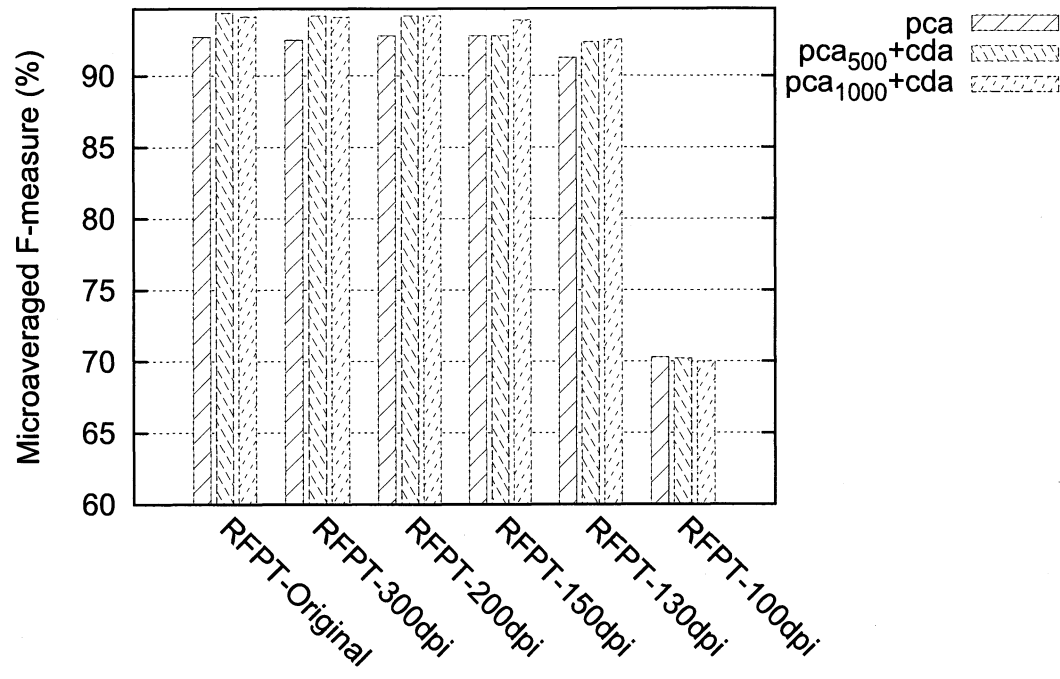
In figure 4.1(a) we can see that classification performance decrease with decrease in resolutions of the OCR texts. The application of CDA on 500 dimension of PC, yields the highest results.

In figure 4.1(b) the results of both ($pca_{500} + cda$) and ($pca_{1000} + cda$) does not have significant difference. The reason for this is due to the fact that feature transformation solves the problem of dependency on text length and therefore achieve higher results regardless of the dimensionality of the PC components used in the PCA+CDA algorithm. Further reading can be found in [12]. But note that the classification results of features reduced by PCA+CDA algorithm outperforms those of PCA reduced features. This is with exceptional of the results of the OCR texts at 150 dpi value where ($pca_{1000} + cda$) is relatively high.

Figure 4.2(a) and 4.2(b) show the empirical results of SVM on AF features and on RFPT features respectively. These figures show the comparison of the

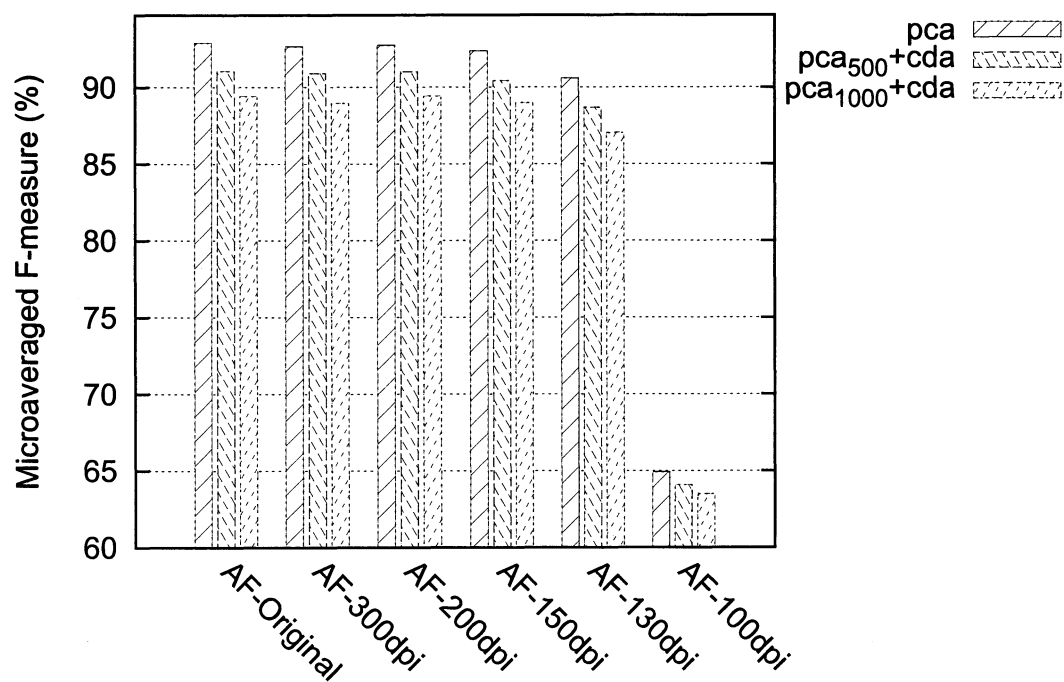


(a) Empirical results of kNN on Absolute frequency (AF)

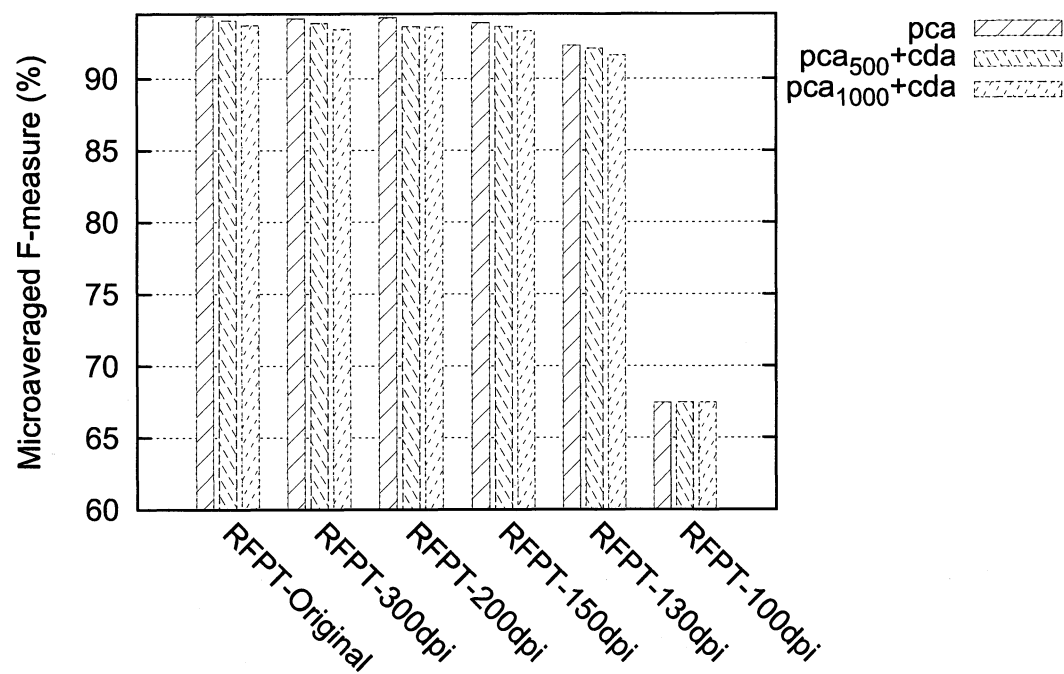


(b) Empirical results of kNN on relative frequency with power transformation (RFPT).

Fig. 4.1: Empirical results of kNN classifier.



(a) Empirical results of SVM on Absolute frequency (AF)



(b) Empirical results of SVM on relative frequency with power transformation (RFPT).

Fig. 4.2: Empirical results of SVM classifier.

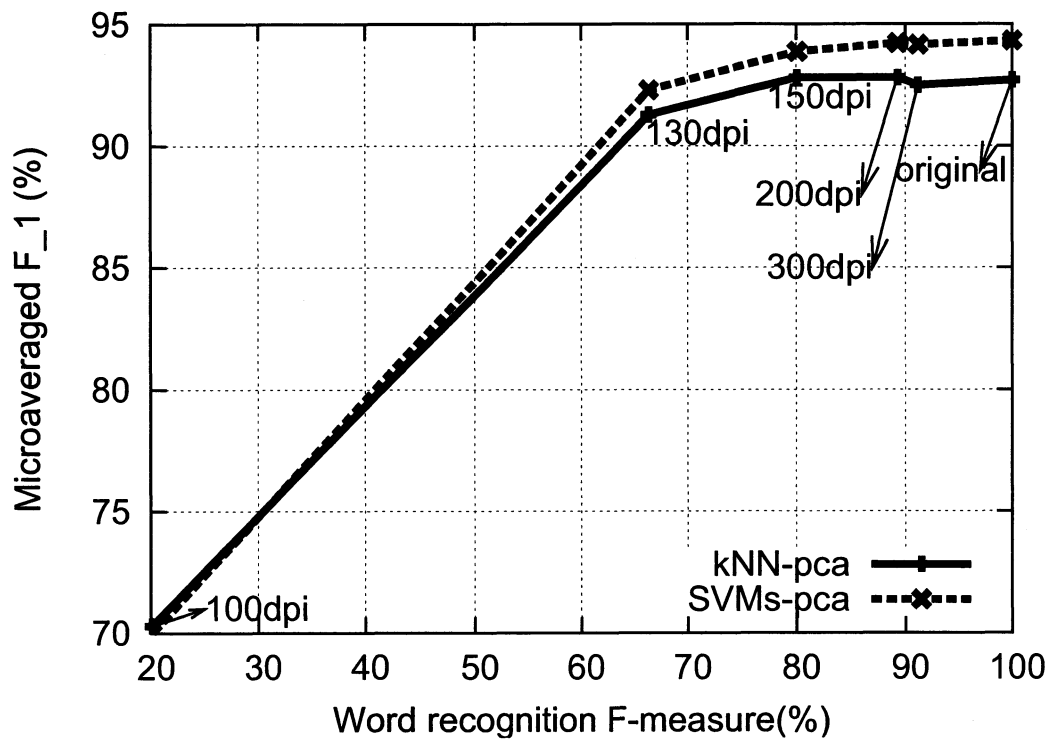


Fig. 4.3: Classification performance after power transformed relative frequency (RFPT) features vs Word recognition rate F-measure of the data at 300dpi, 200dpi, 150dpi, 130dpi and 100dpi.

results of using PCA only and PCA+CDA algorithm for feature reduction. It indicates the results of applying CDA on 500 PC ($pca_{500} + cda$) and that of applying CDA on 1000 PC ($pca_{1000} + cda$). It can be seen clearly in Figure 4.2 that the classification results of PCA reduced features performed best. Also the behavior of PCA+CDA algorithm is clearly revealed in figure 4.2 where by the classification results of ($pca_{500} + cda$) reduced features achieved higher results than those of ($pca_{1000} + cda$).

We found out that PCA+CDA algorithm on few categories (in our case 10 categories) improved the performance of weak classifiers like kNN (figure 4.1 where PCA+CDA outperforms PCA) but does not improve the classification performance of the strong classifier like SVM (figure 4.3 where PCA outperforms PCA+CDA).

Therefore we left out the PCA+CDA algorithm for this matter and we present the summary of classification results of PCA reduced features in figure 4.3. This

summarizes the relationship between classification performance of PCA reduced RFPT features and Word recognition rate.

4.2 Results of Parts of Speech Analysis (POSA)

Firstly, nouns were removed from a group containing *bag-of-words* and this yielded the lowest classification results. We then extracted verbs, adjectives and adverbs separately by using the same procedures followed in equation (3.8). When verbs were removed the classification results were higher than that of after removing nouns. When adjectives were removed the results were slightly low compared to that of when verbs were removed. And lastly when adverbs were removed the classification performance were of almost the same results as after removing verbs. This shows that nouns contributes more in describing a category's content. Then adjectives showed more contribution than verbs. Adverbs showed the contribution of almost the same as verbs. Pronouns showed more contribution than Adverbs and Adjectives.

After knowing the hierarchy of the POS contribution in OCR text classification, we combined some of POS to find out the suitable combination that will show higher classification performance in all OCR texts. Nouns were combined with verbs and the results were evaluated. The noun, verb combination were combined with adjectives and lastly pronouns and adverbs were added. After adding adverbs, the results did not show substantial improvement but it made the results stable in all OCR texts. The classification performance of nouns, verbs, adjectives, adverbs combination improved more when pronouns were added. Therefore in this work we used nouns, pronouns, verbs, adjectives and adverbs because they gave stable and encouraging results.

Empirical results shows that POSA improved classification performance of OCR texts. This can be seen in Figure 4.5 which is the graphical representation of Word Recognition rate against Classification performance. It also shows that OCR text classification performance decrease with increase in OCR errors. With POSA approach, classification performance improved even with the presence of OCR errors. From this graph we can say that POSA improved classification performance of OCR texts.

The richer the feature set used to train a classifier the higher the classification performance. This technique is more effective in OCR text classification than using *bag-of-words* as a feature set. The use of POSA reduces less important POS as well as OCR errors which can mis-represent documents. As it can be observed in the Figure 4.4, the classification results without POSA were low because *bag-of-words* were used as a feature set. This included less informative words which interfered classification.

Figure 4.6 represents the empirical results of kNN classification of data with and without POSA. Figure 4.5(a) shows the results of AF feature set. Figure 4.5(b) shows the results of RFPT feature set. These figures reveal that POSA application on AF and RFPT features improved classification of OCR texts. Consider the result of RFPT features at 130dpi in Figure 4.5(b), classification performance on features without POSA were $F_1 = 86.11\%$. After applying POSA, the results were as high as $F_1 = 89.08\%$. Moreover, when POSA of AF features were transformed to RFPT, in figure 4.5(b) the results microaverage $F_1 = 91.84\%$. We can see that the classification performance of OCR texts by kNN classifier were enhanced whenever POSA was applied.

Figure 4.5 shows the comparison of empirical results of SVM classification of data with and without POSA. Figure 4.6(a) shows the results of AF feature set. Figure 4.6(b) shows the results of RFPT feature set. Consider the results of RFPT features at 130dpi in Figure 4.6(a), classification performance on AF features without POSA were 90.3%. After applying POSA, 91.16% of micro-averaged F_1 was achieved and the results of their transformed features is 93.19% in Figure 4.6(b). This shows that classification of OCR texts performs best with the application of POSA.

In both kNN and SVM_s , RFPT features achieved highest classification performance. The encouraging point is that in all figures the results shows that POSA improved classification performance of OCR texts especially for degraded texts.

We have found out that the tagger tagged most of the unknown (OCR errors) words as either noun or proper nouns depending on their linguistic position and rules. For example the word *against* was misrecognized by an OCR system as *againt* and the tagger tagged or classified it as a noun instead of preposition. Let

us take a closer look at table 4.1, the total number of each POS in each test data is presented. It also shows the average number of each POS per page in each test data. Note how the number of nouns and proper nouns increased significantly on the OCR text of lower dpi value, 130. In the presence of many OCR errors its tagging efficiency decreased. Therefore OCR errors affected the performance of a tree tagger as well as the categorization performance.

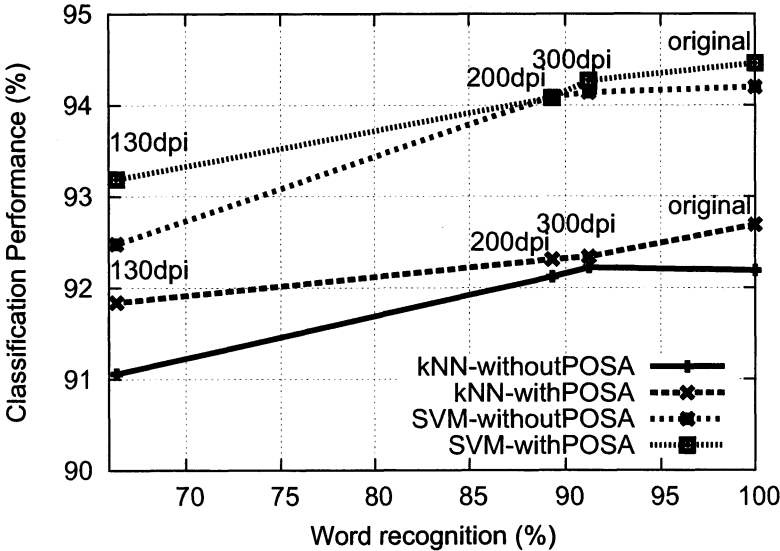
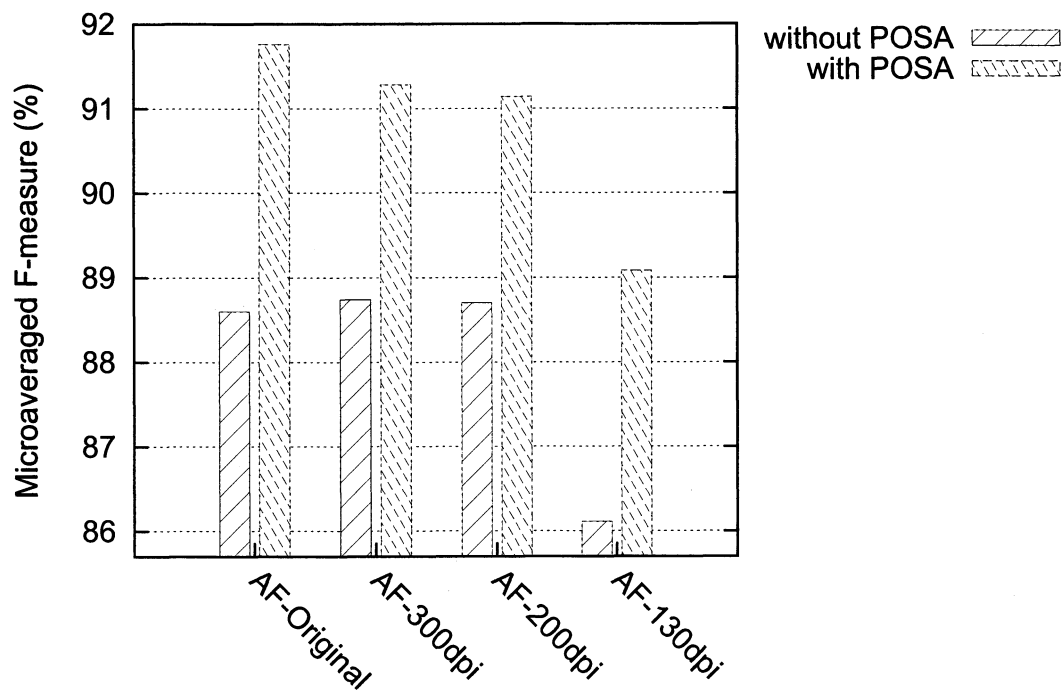


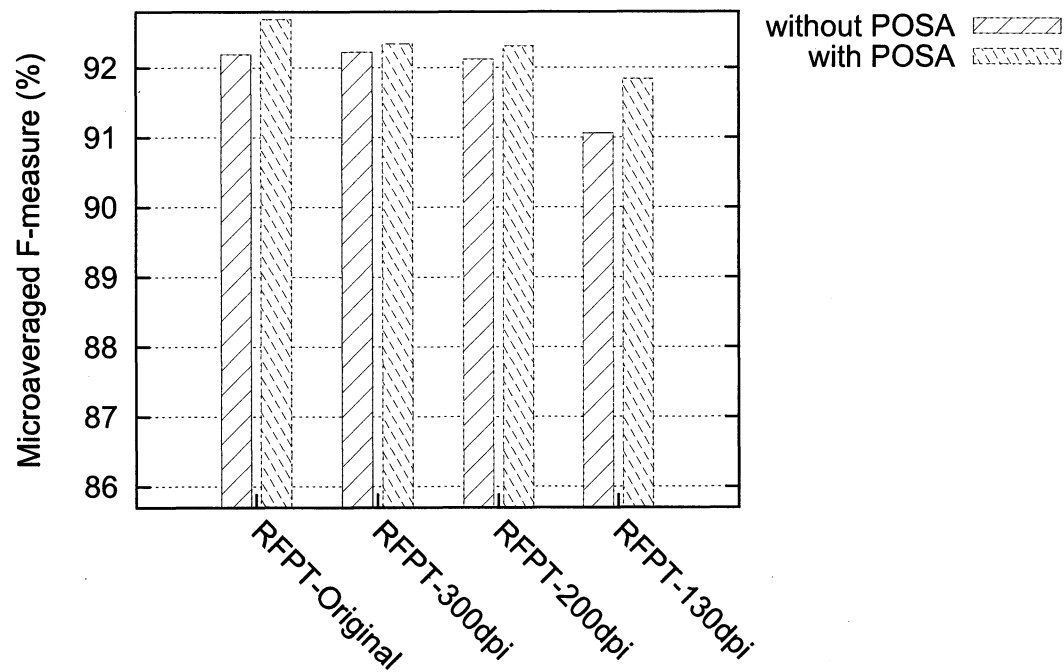
Fig. 4.4: Effects of OCR errors in relation to Word Recognition (Micro-average F_1) in % and Classification Performance (micro-average F_1 in %).

Test data	Noun		Pronoun		Verb		Adverb		Adjective	
	total	ANN	total	ANP	total	ANV	total	AND	total	ANJ
Origin	124,184	48.79	6,691	2.62	33,927	13.33	7,080	2.78	21,086	8.28
300dpi	123,545	49.16	7,659	3.04	33,644	13.38	7,075	2.81	21,182	8.42
200dpi	122,780	48.47	8,450	3.33	33,515	13.23	7,072	2.79	20,921	8.25
130dpi	128,751	50.60	7,315	2.87	31,955	12.56	6,736	2.64	19,441	7.64

Table. 4.1: Total and average number of each Parts of Speech used in the experiment. The abbreviations ANN, ANP, ANV AND and ANJ means the average number of nouns, pronouns, verbs, adverbs and adjectives per document respectively.

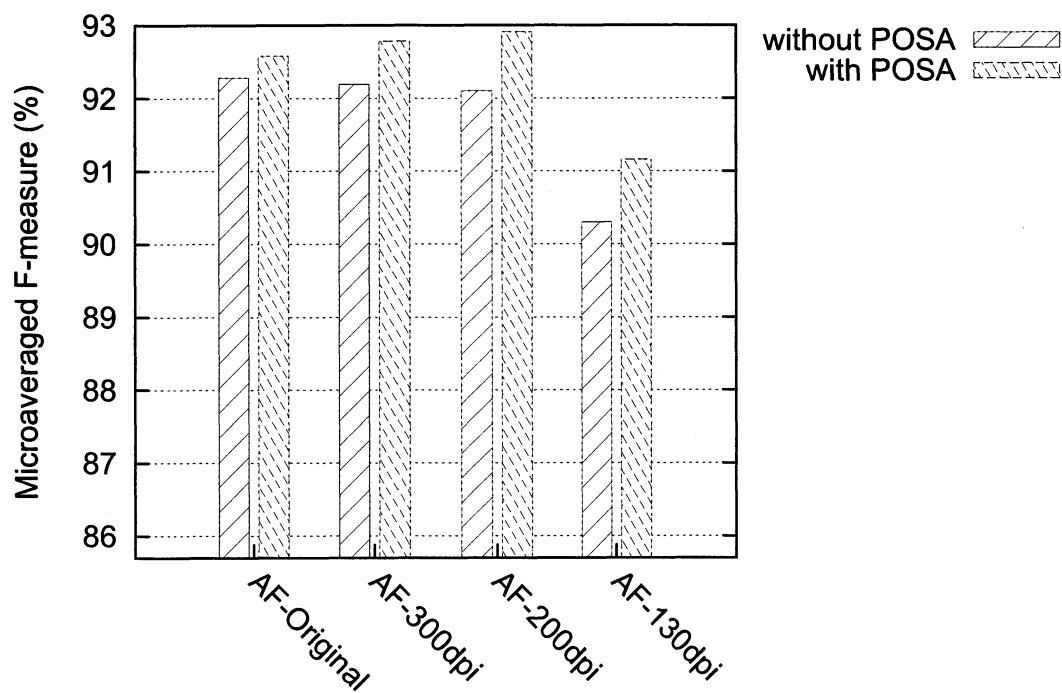


(a) Comparison of classification performance using AF Features with and without POSA.

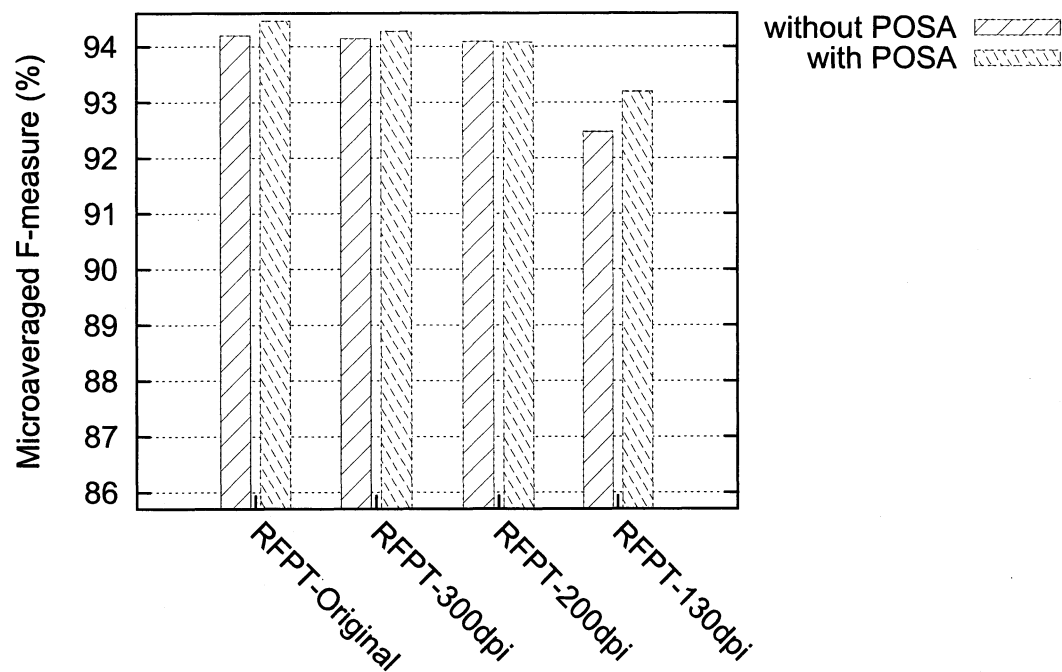


(b) Comparison of classification performance using RFPT Features with and without POSA.

Fig. 4.5: Empirical results of with and without POSA using kNN classifier.



(a) Comparison of classification performance using AF Features with and without POSA.



(b) Comparison of classification performance using RFPT Features with and without POSA.

Fig. 4.6: Empirical results of with and without POSA using SVM classifier.

Chapter 5

Conclusion

1. We found out that PCA+CDA algorithm on few categories (in our case 10 categories) improved the classification performance of weak classifier like k NN but does not improve the classification performance of the strong classifier like SVM. According to the results explained in the previous section, it can be clearly seen that classification performance decrease with decrease in resolutions of the OCR texts.
2. We have presented an impact of linguistic features in ATC. Our approach of generating linguistic features has given promising results. Rather than using the conventional approach i.e, *bag-of-words*, POSA and suitably combining POS elements can improve classification performance.
3. POSA also improved classification performance of OCR texts classification especially in the degraded features i.e, in the presence of many OCR errors.
4. Since we used five parts of speech leaving out the other three groups of POS namely prepositions, conjunctions and articles, and we were able to obtain better classification results than using all eight POS, therefore the removal of prepositions, conjunctions and articles improved text classification performance.

Chapter 6

Related Works and future work

The literature shows few research works done previously on PCA+CDA algorithm as a feature reduction tool. The example of the work which adopted this algorithm on OCR texts is [13]. They used 115 categories of the same Reuters-21578 collection as our work. They adopted PCA+CDA method which gave good results in improving TC performance. In our work we used only 10 categories on the same method and it does not improve classification when SVM classifier was used as compared to the most commonly used method, PCA.

The literature also shows few research works done previously on parts-of-speech analysis in relation to Automatic text classification. To the best of our knowledge, our work is the only literature of parts-of-speech analysis of OCR texts in relation to ATC. Therefore most of the works we survey differ from this paper.

The authors in, [5] proposed the use of transformed features for OCR texts. Their work shows that using transformed features generated using *bag-of-words* approach improved the classification performance. Our work in this paper differ from the work on [5] as we do not focus on *bag-of-words* as feature set. Instead we generate features based on POSA and suitably combine POS elements to improve classification performance of OCR texts.

Another example of works that used *bag-of-words* to represent the texts is found in [15]. They used untransformed features and classified multipage document by a hybrid naive Bayes HMM approach. In contrast we used parts of speech as features and *k*NN and SVM as classifiers. We present improved classification results.

Also [22] make use of on-line thesaurus and dictionaries such as Word-Net based POS feature selection for document representation. He did not use any degraded texts like what we presented in our work.

Other authors who adopted Parts of speech as document representation in different ways for Text classification is [21]. Their way of applying Parts of speech are completely different from ours except in the fact that they made use of POS.

Future work

1. Future work includes using TFIDF on POSA as a feature set and also the use of other classifiers.
2. Since few researchers have used PCA+CDA Algorithm for feature reduction in ATC, it will be of interest to adopt other classifiers on this approach for concrete conclusions.
3. As there are very few researches done on linguistic features representation of OCR texts, there is a great need for extensive experiments using more textual samples to effectively reflect the effect of this approach in the real world application.

References

- [1] K. Fukunaga. Introduction to Statistical Pattern Recognition. Academic Press, 2 edition, 1990.
- [2] T. Joachims. Learning to classify text using support vector machines: Methods, Theory and Algorithms. Kluwer Academic Publishers Boston Dordrecht London, 2001.
- [3] S. Lam and L. Lee. Feature reduction for neural network based text categorization. In Proceedings of DASFAA-99, 6th IEEE International Conference on Database Advanced systems for advanced applications, pages 195202, 1999. Hsinchu, TW.
- [4] H.-S. Lim. Improving kNN based text classification with well estimated parameters. In International Conference on Neural Information Processing, pages 516 523, 2004.
- [5] M. Murata, L. S. P. Busagala, W. Ohyama, T. Wakabayashi, and F. Kimura. The impact of ocr accuracy and feature transformation on automatic text classification. In Document Analysis Systems, pages 506517, 2006.
- [6] J. Rennie, L. Shih, J. Teevan, and D. Karger. Tackling the poor assumptions of naive bayes text classifiers. In Proceedings of the 12th International Conference on Machine Learning (ICML), pages 616623, Washington DC, 2003).
- [7] H. Schmid. Probabilistic part-of-speech tagging using decision trees, 1994.
- [8] F. Sebastiani and C. N. D. Ricerche. Machine learning in automated text categorization. ACM Computing Surveys, 34:147, 2002.
- [9] K. Taghva, T. Nartker, A. Condit, and J. Borsack. Automatic removal of “garbage strings” in ocr text: improved classification results. An implementation.
- [10] Y. Yang. An evaluation of statistical approaches to text categorization. Jour-

- nal of Information Retrieval, 1:6788, 1999.
- [11] Y. Yang and X. Liu. A re-examination of text categorization methods. In Proceedings of the Twenty-First International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 4249, 1999.
- [12] Busagala, L.S.P.; Ohyama, W.; Wakabayashi, T.; Kimura, F. (2005). Machine Learning with Transformed features in Automatic Text Classification; Proceedings of ECML/PKDD-05 workshop on Sub-symbolic Paradigms for Learning in Structured Domains (Relational Machine Learning). pp. 11-20. Oct. 3-7, 2005. Porto, Portugal
- [13] L. S. Busagala, W. Ohyama, T. Wakabayashi, and F. Kimura. Improving automatic text classification by integrated feature analysis. IEICE - Trans. Inf. Syst., E91-D(4):11011109, 2008.
- [14] S. Chapman. Measuring search retrieval accuracy of uncorrected ocr: Findings from the harvard-radcliffe online historical reference shelf digitization project. Harvard University Library, Harvard, 2001.
- [15] P. Frasconi, G. Soda, and A. Vullo. Text categorization for multi-page documents: A hybrid naive bayes hmm approach. In In: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries, pages 1120. ACM Press, 2001.
- [16] S. Scott and S. Matwin. Text Classification Using WordNet Hypernyms. In: Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems, August 16, 1998, Montreal, Canada Association for Computational Linguistics, Morristown, NJ, USA (1998).
- [17] Bishop, C. M. 2006 Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc.
- [18] Lewis, D. D. 1992. An evaluation of phrasal and clustered representations on a text categorization task. In Proceedings of the 15th Annual international ACM SIGIR Conference on Research and Development in information Retrieval (Copenhagen, Denmark, June 21 - 24, 1992).
- [19] Lewis, D. D. 1991. Evaluating text categorization. In Proceedings of the Workshop on Speech and Natural Language (Pacific Grove, California, February 19 - 22, 1991).

- [20] Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. 2000. Text Classification from Labeled and Unlabeled Documents using EM. *Mach. Learn.* 39, 2-3 (May. 2000), 103-134.
- [21] Alessandro Moschitti, Roberto Basili: Complex Linguistic Features for Text Classification: A Comprehensive Study. *ECIR 2004*: 181-196
- [22] Stephanie Chua: The Role of Parts-of-Speech in Feature Selection *Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I IMECS 2008*, 19-21 March, 2008, Hong Kong
- [23] Lam, W. and Han, Y. 2003. Automatic Textual Document Categorization Based on Generalized Instance Sets and a Metamodel. *IEEE Trans. Pattern Anal. Mach. Intell.* 25, 5 (May. 2003), 628-633.
- [24] Peter Verboon and I.A. van der Lans. Robust canonical discriminant analysis. *Py-chometrika Journal*, 59(4):48-507, 1994
- [25] R.O.Duda,P.E.Hart,D.G.Stork. *Pattern Classification(2nd Edition)*. P.117-125. J Wiley. 2000.
- [26] W.Duch. *Computational Intelligence: Methods and Applications*. Lecture 6: Principal Component Analysis. 2006
- [27] Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.* 34, 1 (Mar. 2002), 1-47.