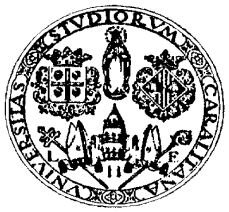

University of Cagliari

PhD in Computer Science

Analysis and Implementation of Methods for the Text Categorization

by

Stefania Dessì

A thesis submitted for the degree of

Philosophiæ Doctor

XXVII Cycle

Supervisor: Prof. Nicoletta Dessì

PhD Coordinator: Prof. Giovanni Michele Pinna

INF/01

2013 – 2014

Abstract

Text Categorization (TC) is the automatic classification of text documents under pre-defined categories, or classes. Popular TC approaches map categories into symbolic labels and use a training set of documents, previously labeled by human experts, to build a classifier which enables the automatic TC of unlabeled documents. Suitable TC methods come from the field of data mining and information retrieval, however the following issues remain unsolved.

First, the classifier performance depends heavily on hand-labeled documents that are the only source of knowledge for learning the classifier. Being a labor-intensive and time consuming activity, the manual attribution of documents to categories is extremely costly. This creates a serious limitations when a set of manual labeled data is not available, as it happens in most cases.

Second, even a moderately sized text collection often has tens of thousands of terms in that making the classification cost prohibitive for learning algorithms that do not scale well to large problem sizes.

Most important, TC should be based on the text content rather than on a set of hand-labeled documents whose categorization depends on the subjective judgment of a human classifier.

This thesis aims at facing the above issues by proposing innovative approaches which leverage techniques from data mining and information retrieval.

To face problems about both the high dimensionality of the text collection and the large number of terms in a single text, the thesis proposes a hybrid model for term selection which combines and takes advantage of both filter and wrapper approaches. In detail, the proposed model uses a filter to rank the list of terms present in documents to ensure that useful terms are unlikely to be screened out. Next, to limit classification problems due to the correlation among terms, this ranked list is refined by a wrapper that uses a Genetic Algorithm (GA) to retaining the most informative and discriminative terms. Experimental results compare well

with some of the top-performing learning algorithms for TC and seems to confirm the effectiveness of the proposed model.

To face the issues about the lack and the subjectivity of manually labeled datasets, the basic idea is to use an ontology-based approach which does not depend on the existence of a training set and relies solely on a set of concepts within a given domain and the relationships between concepts.

In this regard, the thesis proposes a text categorization approach that applies WordNet for selecting the correct sense of words in a document, and utilizes domain names in WordNet Domains for classification purposes. Experiments show that the proposed approach performs well in classifying a large corpus of documents.

This thesis contributes to the area of data mining and information retrieval. Specifically, it introduces and evaluates novel techniques to the field of text categorization. The primary objective of this thesis is to test the hypothesis that:

- text categorization requires and benefits from techniques designed to exploit document content.
- hybrid methods from data mining and information retrieval can better support problems about high dimensionality that is the main aspect of large document collections.
- in absence of manually annotated documents, WordNet domain abstraction can be used that is both useful and general enough to categorize any documents collection.

As a final remark, it is important to acknowledge that much of the inspiration and motivation for this work derived from the vision of the future of text categorization processes which are related to specific application domains such as the business area and the industrial sectors, just to cite a few.

In the end, it is this vision that provided the guiding framework. However, it is equally important to understand that many of the results and techniques developed in this thesis are not limited to text categorization. For example, the evaluation of disambiguation methods is interesting in its own right and is likely to be relevant to other application fields.

Contents

List of Figures

List of Tables

Acronyms

BB	Building Block
BOS	Bag of Synset
BOW	Bag of Word
CHI	Chi Square
CSV	Categorization Status Value
GA	Genetic Algorithm
GWM	Genetic Wrapper Model
IG	Information Gain
IR	Information Retrieval
MDR	Machine Readable Dictionary
ML	Machine Learning
NCBO	National Center for Biotechnology Information
POS	Part of Speech
RDF	Resource Description Framework
SVM	Support Vector Machine
TC	Text Categorization
WSD	Word Sense Disambiguation

Introduction

With the rapid growth of the Internet, digital text documents are increasingly replacing the printed ones. Today, searching books and news electronically is becoming the most popular way for capturing document and information.

Almost all companies have a web page and share their information in Internet.

This “deluge” of documents originates needs for their automatic classification in order to accelerate the search of specific information. Organizing a large amount of documents manually is extremely expensive, time consuming, difficult and, is often impossible to do. Automated text categorization could help to do this hard task.

Specifically, Text Categorization (TC) is the automatic classification of text documents under pre-defined categories, or classes, by combining Information Retrieval (IR) and Machine Learning (ML) techniques. TC is receiving a crescent interest from researchers and developers.

The dominant approach considers to assign keywords to document and then building a classifier by learning, from these set of pre-classified documents, the characteristics of the categories[SF02].

Many information retrieval, statistical classification and machine learning techniques have been applied to TC domains. However, most algorithms may not be completely suitable when the problem of high dimensionality occurs[YP97] [FG03]. A moderately sized text collection often has tens of thousands of terms which makes the classification cost prohibitive for learning algorithms that do not scale well to large problem sizes. In addition, it is known that most terms are

irrelevant for the classification task and some of them even introduce noise that may decrease the overall performance [SM83].

Furthermore, such approaches treat categories as symbolic labels and use a training set, which consists of documents previously assigned to the target categories by human experts.

The drawback of these approaches is that the classifier performance depends heavily on the large amount of hand-labeled documents as they are the only source of knowledge for learning the classifier. Being a labor-intensive and time consuming activity, the manual attribution of documents to categories is extremely costly.

Most important, text categorization should be based on the knowledge that can be extracted from the text content rather than on a set of documents where a text could be attributed to one or another category, depending on the subjective judgment of a human classifier.

To overcome this problem, semi-supervised learning techniques have been proposed that require only a small set of labeled data for each category [ZXJ07].

These methods require a training set of pre-classified documents and it is often the case that a suitable set of well categorized, typically by humans, training documents is not available. This creates serious limitations for the usefulness of the above learning techniques in several operational scenarios ranging from the management of web-documents to the classification of incoming news into categories, such as business, sport, politics, etc.

This thesis aims at solving the problems described above and proposes some methods for facing both the high dimensionality of the text collection, and the difficulty of find hand-labeled resources to train and test the classifier.

About the high dimensional, this thesis proposes dimensionality reduction techniques (i.e. feature selection or feature extraction) which are beneficial for increasing scalability, reliability, efficiency and accuracy of text classification algorithms [ZXJ07]. In particular, the proposed techniques deal with feature selection (namely term selection in TC) i.e. process that reduces the dimensionality of the feature space by only retaining the most informative or discriminative terms.

Generally, feature selection algorithms can be broadly divided in two categories: filters and wrappers. Filter approaches evaluate the relevance of each single term according to a particular feature scoring metric and retain the best terms set. Although simple and fast, filters lack robustness against correlations between terms and it is not clear how to determine the optimal number of the retained best terms, namely the threshold value. Conversely, wrappers compare different term subsets and evaluate them using the classification algorithm that will be employed to build the final classifier. Being exhaustive search impractical, greedy procedures or meta-heuristics are usually employed to guide combinatorial search through the space of candidate term subsets looking for a good trade-off between performance and computational cost. Even if wrapper methods have been shown to generally perform better than filters [FG03], their time-consuming behavior has made prominent the use of filter approaches in TC area.

In particular, I present a hybrid model for term selection which combines and takes advantage of both filter and wrapper approaches in order to overcome their limitations.

In detail, the model uses a filter to rank the list of terms present in documents. Then, terms with the highest score values are selected, in an incremental way, resulting in a set of nested term subsets. The preliminary use of the filter ensures that useful terms are unlikely to be screened out. Differently from most filter-

based approaches, the ranked list is not cut off according to a single (somewhat arbitrary) threshold value. To limit classification problems due to the correlation among terms, the proposed approach considers refining the selection process by employing a wrapper that uses a Genetic Algorithm (GA) as search strategy. Unlike traditional wrappers that select the features linearly, a GA performs a random terms combination and shows its potentiality in exploring features set of high dimensionality. For its characteristics, this method is named Genetic Wrapper Model (GWM).

To evaluate the proposed approach I chose the standard test sets Reuters-21578 [LDD97]. Experimental results compare well with some of the top-performing learning algorithms for TC and confirm the effectiveness of the proposed model.

This approach was also used to extract the most relevant annotations within a gene family, i.e. a group of genes sharing similar functions. The study considers 5 families described by a set of gene summaries [S2]. These summaries are first annotated using NCBO annotator[JS+09], a public tool which uses biomedical ontologies as existing knowledge resources. Then, I applied the Genetic Wrapper Model to resulted annotations in order to extract the most representative concepts for every family.

This approach stresses and demonstrates that text categorization should be based on the knowledge that can be extracted from the text content rather than on a set of documents where a text could be attributed to one or another category, depending on the subjective judgment of a human classifier.

Going beyond, recent research introduced text categorization methods based on leveraging the existing knowledge represented in a domain ontology [BWL10]. The basic idea is to use an ontology for providing a functionality that

is similar to the knowledge provided by human experts with a manual document classification.

Ontologies are used as data-models or taxonomies to provide the text with a semantic structure by annotating it with unambiguous topics about the thematic content of the document. The novelty of these ontology-based approaches is that they are independent of the existence of a training set and rely solely on a set of concepts within a given domain and the relationships between concepts.

One of the best known sources of external knowledge is WordNet [MGA95] [FC98], a network of related words, that organizes English nouns, verbs, adjectives and adverbs into synonym sets, called synsets, and defines relations between these synsets.

In this regard, the second part of the thesis proposes a text categorization approach that is designed to fully exploiting semantic resources as it employs the ontological knowledge not only as lexical support, but also for deriving the final categorization of documents in topics categories.

Specifically, my work relates to apply WordNet for selecting the correct sense of words in a document, and utilizes domain names in WordNet Domains [MC00][BF+04] for classification purposes. Experiments show how the approach performs well in classifying a large corpus of documents.

This thesis is organized in to 5 different chapters. The first two chapters present an overview which summaries the state of the art of Text Categorization approaches, with attention to Machine Learning techniques and Word Sense disambiguation methods [SF02][SF05][NR09]. Chapter 3 proposes and discusses in detail a hybrid model for the text categorization that combines and take advantage of two feature selection techniques: filter and wrapper. This approach is applied to bag of words extracted from articles of news data collection. In the

4th chapter, the above hybrid model is used to study the bag of annotations obtained from a collection of gene summaries. An ontology based approach for the TC is exposed in the 5th chapter. Firstly the approach considers the context of a word to disambiguate its sense. Secondly, it exploits semantic resources for obtaining lexical support to derive the final categorization of documents in topic categories. The last chapter presents the conclusions.

Analysis and results described in this thesis have been presented in the following papers:

- Laura Maria Cannas, Nicoletta Dessì, Stefania Dessì: A Model for Term Selection in Text Categorization Problems. DEXA Workshops 2012 (TIR'12): 169-173;
- Nicoletta Dessì, Stefania Dessì, and Barbara Pes: A Fully Semantic Approach to Large Scale Text Categorization. ISCIS 2013: 149-157;
- Nicoletta Dessì, Stefania Dessì, Emanuele Pascariello, and Barbara Pes: Exploring The Relatedness of Gene Sets. Submitted to CIBB 2014 on 12/11/2014 – accepted.

1 Text Categorization

Text categorization is the task of automatically sorting a set of document into categories (or classes, or topic) from a pre-defined set [SF02].

This discipline is obtaining increasing interest in the last ten years from researchers of Information Retrieval (IR) and Machine Learning (ML). Specifically, IR is the activity of obtaining information resources relevant to an information need from a collection of information resources, and ML is a scientific discipline that deals with the construction and study of algorithms that learn knowledge from data. The TC process is a general inductive task that automatically builds an automatic text classifier by learning, from a set of pre-classified documents, the characteristics of the categories of interest. The advantages of this approach are an accuracy comparable to that achieved by human experts, and a considerable savings in term of expert labor power, since no intervention from either knowledge engineers or domain expert is needed for the construction of the classifier.

1.1 Text Categorization process

A Text Categorization process assigns a Boolean value to each pair $<d_j,c_i> \in DxC$, where D is a domain of documents and $C= \{c_1, \ldots, c_{|C|}\}$ is a set of pre-defined categories. Assign a True value to $<d_j,c_i>$ indicates that a document d_j belongs to the class c_i (positive example), while a False value indicates that d_j does not belong to c_i (negative example). More formally, the process is described as the task which approximates the unknown target function $\breve{\Phi} : DxC \rightarrow \{T, F\}$ (that describes how documents ought to be classified) by means of a function $\Phi : DxC \rightarrow \{T, F\}$ called the classifier (aka rule, or hypothesis, or model).

Categories are just symbolic labels: no additional knowledge (of a procedural or declarative nature) about their meaning is usually available, and it is often the case that no metadata (such as e.g. publication date, document type, and publication source) are available either. In these cases, classification must be accomplished only on the basis of the knowledge extracted from the documents themselves. When, in a given application, either external knowledge nor metadata is available, heuristic techniques of any nature may be adopted in order to leverage on these data, either in combination or in isolation using the IR and ML techniques.

TC is a subjective task. When two experts (human or artificial) decide about classifying a document d_j under a category c_i, they may disagree. This disagreement happens with relatively high frequency. As a consequence, the meaning of a category is subjective and, rather than attempting to produce a "gold standard" of dubious existence, the ML techniques aim to reproduce this very subjectivity by examining its manifestations, i.e. documents that the expert has manually classified. This kind of learning is usually called supervised learning, as it is supervised, or facilitated, by the knowledge of the pre-classified data.

1.2 Single or Multi label Text Classification

Depending on the application, TC may be either a single-label task, or a multi-label task. Single-label task (a.k.a. non-overlapping categories) is the case in which exactly one category must be assigned to each $d_j \in D$, while multi-label task (a.k.a. overlapping categories) is when any number of categories from 0 to $|C|$ may be assigned to the same $d_j \in D$.

A special case of single-label TC is binary TC, in which each $d_j \in D$ must be assigned either to the category c_i or to its complement $\overline{c}_i$. From the ML standpoint, learning a binary classifier (and hence a multi-label classifier) is

usually simpler than learning a single-label classifier. As a consequence, while all classes of supervised ML techniques deal with binary classification problems since their very invention, for some classes of techniques (e.g. support vector machines) a satisfactory solution of the single-class problem is still the object of several active investigations [CS01].

1.3 Machine Learning Approach to Text Categorization

Since the early ’90s, the ML approach to TC has gained popularity and has eventually become the dominant one, at least in the research community. According to this approach, a general inductive process (also called the learner) automatically builds a classifier for a category c_i by observing the characteristics of a set of documents manually classified under c_i or $\overline{c}_i$ by a domain expert; from these characteristics, the inductive process extracts the characteristics that a new unseen document should have in order to be classified under c_i .

In ML terminology, the classification problem is an activity of supervised learning, since the learning process is “supervised” by the knowledge of the categories and of the training instances that belong to them.

Within the ML approach, the pre-classified documents are then the key resource. In the most favorable case, they are already available but, in the less favorable case, no manually classified documents are available.

It is easier to manually classify a set of documents than to build and tune a set of rules: it is easier to characterize a concept extensionally (i.e., to select its instances) than intentionally (i.e., to describe the concept by a sentence, or to describe a procedure for recognizing its instances).

Built by means of ML techniques, classifiers achieve impressive levels of effectiveness making automatic classification a qualitatively, and not only economically, viable alternative to manual classification.

1.3.1 Training Set, Test Set, and Validation Set

The Machine Learning approach relies on the availability of an initial corpus of documents pre-classified under a set of categories. Being Φ the classifier we have to build, a document d_j is a positive example of c_i if the document d_j belongs to the class c_i , $\breve{\Phi}$ (d_j,c_i) = T. It is the case of a negative example of c_i if the document d_j does not belong to the class c_i , $\breve{\Phi}$ (d_j,c_i) = F.

Before training Φ, the initial corpus is split in two sets, not necessarily of equal size, namely the training (and validation) set, and the test set. The classifier is built inductively by observing the characteristics of the training set, and using test set for testing the effectiveness of the classifier. A measure of classification effectiveness is based the number of documents the classifier categorizes correctly. The described classification process is called the train-and-test approach. An alternative is the k-fold cross-validation approach [MTM96] in which k different classifier are built by splitting the initial corpus into k different sets and then iteratively applying the train-and-test approach by considering one set as test set, and the remaining k-1 set as training set. The final accuracy is evaluated by averaging the accuracy of the individual classifier. In this approach, a validation set is used for tuning parameters.

1.3.2 The TC process life cycle

We can distinguish three different phases in the life cycle of a TC process [SF05]:

- Document indexing;
- Classifier learning;
- Classifier evaluation.

1.3.2.1 Document Indexing

Because texts cannot be directly interpreted by a classifier or by a classifier-building algorithm, an indexing procedure is needed that maps a text d_j into a compact representation of its content.

For this purpose, similarity to what happens in IR, a text d_j is typically represented as a vector of weighted terms $\vec{d}_j = \langle w_{1j}, \ldots, w_{|T|j} \rangle$. Here T is the dictionary, i.e. the set of terms (a.k.a. features) that occur at least once in at least k training documents, and $0 \leq w_{kj} \leq 1$ quantifies the importance of t_k in characterizing the semantics of d_j. Typical values of k are between 1 and 5.

An indexing method is characterized by a definition of the term, and the method to compute term weights. Concerning term definition, the most frequent choice is to identify terms a) using words which occur in the document; the set of this terms is often called either set of words or the bag of words (with the exception of stopwords, i.e. topic-neutral words such as articles and prepositions, which are eliminated in a pre-processing phase), b) with their stems (i.e. their morphological roots, obtained by applying a stemming algorithm [FWB92]).

For the second issue, terms may be binary-valued (i.e. $w_{kj} \in \{0, 1\}$) or real-valued (i.e. $0 \leq w_{kj} \leq 1$), depending on whether the algorithm used to build the classifier and the classifiers, once they have been built, require binary input or not. Binary weights simply indicate presence/absence of the term in the document. Non-binary weights are computed by either statistical or probabilistic techniques, the former being the most common option.

In the case of non-binary indexing, for determining the weight w_{kj} of term t_k in document d_j any IR-style indexing technique may be used. The most common option is the standard tf*idf function (see [SB88]), which evaluates how many times a term occurs in a document.

Dimensionality Reduction

Unlike in text retrieval, in TC the high dimensionality of the term space (i.e., the large value of $|T|$) may be problematic. For that reason, a dimensionality reduction phase is often applied to reduce the size T i.e. the number of documents to be considered. This has both the effect of reducing overfitting (i.e. the tendency of the classifier to better classify the data over which it has been trained on than new unseen data), and to make the problem more manageable for the learning method, since many such methods are known not to scale well to high problem sizes.

Dimensionality reduction can be performed locally (i.e., for each individual category) or globally (i.e. under all categories $C = \{ c_1, \ldots, c_{|C|}\}$). Dimensionality reduction often takes the form of feature selection: each term is scored by means of a scoring function that captures its degree of correlation with c_i (positive, and sometimes also negative). Only the highest scoring terms are used for document representation. Alternatively, dimensionality reduction may take the form of feature extraction: a set of "artificial" terms is generated from the original term set where the new terms are both fewer and stochastically more independent from each other than the original ones. In this thesis, the former approach of dimensionality reduction is adopted.

Dimensionality Reduction by Term Selection

Techniques for term selection are applied to select, from the original dictionary *T*, the set *T'* of terms (with $|T'| \ll |T|$) that yields the highest effectiveness.

According to [YP97] term selection may even result in a moderate ($\leq$5%) increase in effectiveness, depending on the classifier, on the percentage of the reduction, and on the term selection technique used.

Moulinier et al. [MRG96] proposed the so called wrapper approach, where *T'* is identified by means of the same learning method that will be used for building the classifier [JKP94]. Starting from an initial term set, a new term set is generated by either adding or removing a term. When a new term set is generated, a classifier based on it is built and then tested on a validation set. The term set that results in the best effectiveness is chosen. This approach has the advantage of being tuned to the learning algorithm being used; moreover, if local dimensionality reduction is performed, different numbers of terms for different categories may be chosen, depending on whether a category is or not easily separable from the others.

However, the huge size of the space of different term sets makes its cost-prohibitive for standard TC applications. A computationally easier alternative is the filtering approach [JKP94], that is, keeping the $|T'| \ll |T|$ terms that receive the highest score according to a function that measures the "importance" of the term for the TC task.

Dimensionality Reduction Functions

A simple and effective approach for dimensionality reduction consists on evaluating the frequency of a term t_k in a document and retaining only the terms that occur in the highest number of documents.

Other more sophisticated information-theoretic functions have been proposed in the literature, among them the DIA association factor [FB91], Chi-Square(CHI) [CMS01] [GSS00] [SHP95] [SSV00] [YP97] [YL99], NGL coefficient [NGL97] [RS99], Information Gain (IG) [CMS01] [LLS98] [LA92] [LR94] [MG96] [YP97] [YL99], mutual information [DP+98] [LLH97] [LC96] [LR94] [LJ98] [MRG96] [RS99] [TH99] [YP97], odds ratio [CMS01] [RS99], relevancy score [WPW95], and GSS coefficient [GSS00].

The approach presented in this thesis considers to evaluate probabilities on an event space of documents and are estimated by counting occurrences in the training set. All functions are specified "locally" to a specific category c_i; in order to assess the value of a term t_k in a "global" category independent sense. These functions try to formalize the intuition that the best terms for c_i are the ones distributed most differently in the sets of positive and negative examples of c_i.

However, interpretations about this intuition vary across different functions. For instance, in the experimental sciences Chi Square is used to measure how the results of an observation differ (i.e., are independent) from the results expected according to an initial hypothesis (lower values indicate lower dependence). In dimensionality reduction we measure how independent t_k and c_i are. Thus, the terms t_k with the lowest value for $CHI(t_k, c_i)$ are the most independent from c_i. Since we are interested in the terms which are not, we select the terms which result in the highest $CHI(t_k, c_i)$. [YP97] shows that, with various classifiers and various initial corpora, sophisticated techniques such as $IG_{sum}(t_k, c_i)$ or $CHI_{max}(t_k, c_i)$ can reduce the dimensionality of the term space by a factor of 100 with no loss (or even with a small increase) of effectiveness. In this thesis, the functions Chi Square and Information Gain are used for the experiments.

1.3.2.2 Classifier Learning

A text classifier for c_i is automatically built by a general inductive process (the learner) which a) considers the characteristics of a set of documents pre-classified under c_i or $\bar{c}_i$, b) extracts the characteristics that a new unseen document should have in order to belong to c_i [SF05]. Thus, in order to build classifiers for C, a set Ω of documents is needed such that the value of $\Phi(d_j, c_i)$ is known for every $\langle d_j, c_i \rangle \in \Omega \times C$.

As said before, in experimental TC it is usual to partition Ω into three disjoint sets: the training set, the validation set, and the test set. The training set is the set of documents used by the learner to build the classifier. The validation set is the set of documents on which the classifier is tuned. The test set is the set on which the effectiveness of the classifier is finally evaluated. In both the validation and test phase, evaluating the effectiveness means running the classifier on a set of pre-classified documents and checking the degree of correspondence between the output of the classifier and the pre-assigned classes.

Different learners have been applied in the TC literature. Some of these methods generate binary-valued classifiers of the required form $\widehat{\Phi}: D \times C \rightarrow \{T,F\}$, but some others generate real-valued functions of the form $CSV : D \times C \rightarrow [0, 1]$ (CSV standing for categorization status value). For these latter, a set of thresholds τ_i is determined (typically, by experimentation on a validation set) allowing to turn real-valued CSVs into the final binary decisions [YY01].

It is important to notice that in several applications, a method implementing a real-valued function can be profitably used. In this case, determining thresholds is not necessary. For instance, in applications in which the quality of the classification is of critical importance, post-editing the classifier outputs by a human professional it is often necessary. In this case, it may be useful ranking documents in terms of their estimated relevance to the category to support human editors in selection the most appropriate set of documents.

Classifier Learning Techniques

Classifier learning techniques include probabilistic methods, regression methods, decision tree and decision rule learners, neural networks, batch and incremental learners of linear classifiers, example-based methods, support vector machines, genetic algorithms, hidden Markov models, and classifier committees

(which include boosting methods)[SF05]. In the follows, the most used technologies are presented.

Support Vector Machine

The support vector machine (SVM) [JT98] [JT99], in geometrical terms, may be seen as the attempt to find, among all the surfaces σ^1, σ^2, … in $|T|$-dimensional space that separate the positive from the negative training examples (decision surfaces), the surface σ_i that separates the positives examples from the negatives one by the widest possible margin, i.e. they maximize the minimal distance between the hyper-plane and a training example. Results in computational learning theory indicate that this process tends to minimize the generalization error, i.e. the error of the resulting classifier over unseen examples. SVMs were usually conceived for binary classification problems [VVN95], and only recently they have been adapted to multiclass classification problems [CS01].

As regards to TC problems, one advantage is that SVMs methods do not require dimensionality reduction, as they tend to be fairly robust to words overfitting and can scale up to considerable dimensionalities [JT98].

Boosting

The so called ensemble classifiers are based on the idea that k different classifiers Φ_1, ... , Φ_k perform better than a single one if their individual judgments are appropriately combined. The boosting method [SSS98] [SS00][SSV00][NSS03] builds the k classifiers Φ_1, ... , Φ_k by the same learning method (here called the weak learner), and they are sequentially trained.

In training classifier Φ_t one may take into account how classifiers Φ_1, ... , Φ_{t-1} perform on the training examples, and concentrate on selecting those examples on which Φ_1 , ... , Φ_{t-1} perform best.

K-nearest Neighbor

The K-nearest Neighbor method assumes that the class label, which has yet to be assigned to a document, comes closest to those that have been assigned in its neighborhood in the space of function, to be able to predict the class of the new document. The algorithm identifies the k points closer to the target point, according to some similarity measures such as the Euclidean distance, and classifies the document with the class more likely among those of his neighbors. In case of equality, the test document is assigned to the class of the nearest point.

Naïve Bayes

A Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem (from Bayesian statistics) with strong (naïve) independence assumptions [NBC14]. A more descriptive term for the underlying probability model would be "independent feature model".

In simple terms, a naïve Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 4" in diameter. Even if these features depend on each other or upon the existence of the other features, a naïve Bayes classifier considers that all of these properties independently contribute to the probability that this fruit is an apple.

Depending on the precise nature of the probability model, naïve Bayes classifiers can be trained very efficiently in supervised learning settings. In many practical applications, parameter estimation for naïve Bayes models uses the method of maximum likelihood; in other words, one can work with the naïve Bayes model without believing in Bayesian probability or using any Bayesian method.

In spite of their naïve design and apparently over-simplified assumptions, naïve Bayes classifiers have worked quite well in many complex real-world situations. An advantage of the naïve Bayes classifier is that it requires a small amount of training data to estimate the classification parameters (means and variances of the variables). Because variables are assumed to be independent, only their variances for each class need to be determined instead of the entire covariance matrix.

Genetic Algorithms

Genetic algorithms are heuristic strategies for search and optimization that embody the principle of Darwinian natural selection, which regulates biological evolution. These particular algorithms differ significantly from the classic approach.

Genetic algorithms start from a number of possible solutions i.e. individuals classifiers, which form an initial population and provide a mechanism of evolution. It consists in combining individuals between them to simulate reproduction (crossover), where they can take over genetic mutations (mutation) which remove the solutions from possible local optima. The evolutionary cycle is repeated for a number of generations, until having an individual that is assumed as the best solution (i.e., the best classifier).

1.3.2.3 Classifier Evaluation

There are different measures of success for a learner[SF05] including:

- training efficiency: average time required to build a classifier from a given corpus;
- classification efficiency: average time required to classify a document by means of classifier;

- effectiveness: average correctness of classifier's classification behavior.

In TC research, effectiveness is usually considered the most important criterion, since it is the most reliable one when it comes to experimentally comparing different learners or different TC methodologies. On the contrary, efficiency depends on too volatile parameters (e.g. different sw/hw platforms).

Measures of Text Categorization Effectiveness

Precision and Recall

Classification effectiveness is usually measured in terms of the classic IR notions of precision (π) and recall (ρ), adapted to the case of TC. Precision is defined as the probability that if a random document d_x is classified under c_i and this decision is correct. Analogously, recall is defined as the probability that, if a random document d_x ought to be classified under c_i, this decision is taken [SF02]. Precision and recall values related to specific categories may be averaged to obtain global values of π and ρ, that is, global values to the entire set of categories.

Table 1.1. The Contingency Table for c_i

Category c_i		Expert Judgments	
		YES	**NO**
Classifier Judgments	**YES**	TP_i	FP_i
	NO	FN_i	TN_i

These probabilities may be estimated in terms of the contingency table for c_i on a given test set (see Table 1.1). Here, FP_i (false positives) is the number of test documents incorrectly classified under c_i; TN_i (true negatives), TP_i (true positives), and FN_i (false negatives) are defined accordingly.

Estimates (indicated by carets) of precision and recall may thus be obtained as

$$\hat{\pi}_i = \frac{TP_i}{TP_i + FP_i}, \quad \hat{\rho}_i = \frac{TP_i}{TP_i + FN_i}.$$

For obtaining estimates of π and ρ, two different methods may be adopted:

— microaveraging: π and ρ are obtained by summing over all individual decisions:

$$\hat{\pi}^{\mu} = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{|\mathcal{C}|} TP_i}{\sum_{i=1}^{|\mathcal{C}|}(TP_i + FP_i)},$$
$$\hat{\rho}^{\mu} = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{|\mathcal{C}|} TP_i}{\sum_{i=1}^{|\mathcal{C}|}(TP_i + FN_i)},$$

where "μ" indicates microaveraging. The "global" contingency table (Table 1.2) is thus obtained by summing over category-specific contingency tables;

— macroaveraging: precision and recall are first evaluated "locally" for each category, and then "globally" by averaging over the results of the different categories:

$$\hat{\pi}^{M} = \frac{\sum_{i=1}^{|\mathcal{C}|} \hat{\pi}_i}{|\mathcal{C}|}, \quad \hat{\rho}^{M} = \frac{\sum_{i=1}^{|\mathcal{C}|} \hat{\rho}_i}{|\mathcal{C}|},$$

where "M" indicates macroaveraging.

These two methods may give quite different results, especially if the different categories have very different generality.

Table 1.2. The Global Contingency Table

Category set $\mathcal{C} = \{c_1, \ldots, c_{\lvert\mathcal{C}\rvert}\}$		Expert Judgments	
		YES	**NO**
Classifier Judgments	**YES**	$TP = \sum_{i=1}^{\lvert\mathcal{C}\rvert} TP_i$	$FP = \sum_{i=1}^{\lvert\mathcal{C}\rvert} FP_i$
	NO	$FN = \sum_{i=1}^{\lvert\mathcal{C}\rvert} FN_i$	$TN = \sum_{i=1}^{\lvert\mathcal{C}\rvert} TN_i$

Other Measures of Effectiveness

Measures alternative to π and ρ are commonly used in the ML literature, such as accuracy (estimated as $\hat{A} = \frac{TP+TN}{TP+TN+FP+FN}$) and error (estimated as $\hat{E} = \frac{FP+FN}{TP+TN+FP+FN} = 1 - \hat{A}$). However they are not widely used in TC because the large value of their denominator typically in TC resulting makes them much more insensitive to variations in (TP+TN) than π and ρ [YL99].

Measures Alternative to Effectiveness

An important alternative to the effectiveness is the utility of a classifier, a class of measures from decision theory that extend effectiveness by economic criteria such as gain or loss.

Other effectiveness measures different from the ones discussed here have occasionally been used in the literature; these include adjacent score [LL98], coverage [SS00], one-error[SS00], Pearson product-moment correlation [LL98], recall at n [LC96], top candidate [LC96], and top n [LC96].

Combined Effectiveness Measures

Neither precision nor recall makes sense in isolation from each other. A classifier should thus be evaluated by means of a measure which combines π and ρ.

The most popular way to combine the two is the function $F_\beta = \frac{(\beta^2+1)\pi\rho}{\beta^2\pi+\rho}$, for some value $0 \leq \beta \leq \infty$; usually, β is taken to be equal to 1, which means that the F_β function becomes $F_1 = \frac{2\pi\rho}{\pi+\rho}$, i.e. the harmonic mean of precision and recall. Note that for the trivial rejector, π = 1 and ρ = 0, so $F_\beta = 0$ for any value of β (symmetrically, for the trivial acceptor it is true that π = 0, ρ = 1, and $F_\beta = 0$ for any value of β). The breakeven point is the measure in which the value of precision π is equals to the value of recall ρ.

As shown in [MRG96] and [YL99], the breakeven of a classifier is always less or equal than its F_1 value.

Once an effectiveness measure is chosen, a classifier can be tuned (e.g., thresholds and other parameters can be set) so that the resulting effectiveness is the best achievable by that classifier.

Benchmarks for Text Categorization

There are in literature standard benchmark collections that can be used to compare the performance of the classifier. These benchmark collections for TC are publically available for experimental purpose.

In general, different sets of experiments may be used for cross-classifier comparison only if the experiments have been performed [SF02]:

- on exactly the same collection (i.e., same documents and same categories);

- with the same “split” between training set and test set;
- with the same evaluation measure and, whenever this measure depends on some parameters (e.g., the utility matrix chosen), with the same parameter values.

The most widely used benchmark is the Reuters collection, consisting of a set of newswire stories classified under categories related to economics. There are 5 popular versions of Reuters [YL99] and it is usually difficult compare the performance of the classifier if it is not respected the three points above. For example, experiments performed on Reuters-21578(10) “Mode Apté” are not comparable with the other Reuters versions because this collection is the restriction of Reuters-21578 “Mode Apté” to the 10 categories with the highest generality, and is thus an easier collection.

Other test collections that have been frequently used are:

- The OHSUMED collection [HB+94] are titles or title-plus-abstracts from medical journals (OHSUMED is actually a subset of the Medline document base); the categories are the “postable terms” of the MESH thesaurus.
- The 20 Newsgroups collection [LK95] are messages posted to Usenet newsgroups, and the categories are the newsgroups themselves.
- the AP collection.

1.4 **Applications**

The applications of TC are manifold [SF05]. Common traits among all of them are:

- The need to handle and organize documents in which the textual component is either the unique, or dominant, or simplest to interpret, component.

- The need to handle and organize large quantities of such documents, i.e. large enough that their manual organization into classes is either too expensive or not feasible within the time constraints imposed by the application.
- The fact that the set of categories is known in advance, and its variation overtime is small.

Applications may instead vary along several dimensions:

- The nature of the documents; i.e. documents may be structured texts (such as e.g. scientific articles), newswire stories, classified ads, image captions, e-mail messages, transcripts of spoken texts, hypertexts, or other.
- The structure of the classification scheme, i.e. whether this is flat or hierarchical.
- The nature of the task, i.e. whether the task is single-label or multi-label.

The borders between the different classes of applications are not well defined, and some of these may be considered special cases of others. Bellow, there are shown some of the most common applications.

1.4.1 Document organization

Many issues pertaining to document organization and filing, be it for purposes of personal organization or structuring of a corporate document base, may be addressed by TC techniques. For instance, at the offices of a newspaper, it might be necessary to classify all past articles in order to ease future retrieval in the case of new events related to the ones described by the past articles.

Another possible application in the same range is the organization of patents into categories for making later access easier, and of patent applications for

allowing patent officers to discover possible prior work on the same topic [LLS99].

1.4.2 Text filtering

Text filtering is the activity of classifying a stream of incoming documents dispatched in an asynchronous way by an information producer to an information consumer. Typical cases of filtering systems are e-mail filters [WA+99] (in which case the producer is actually a multiplicity of producers), newsfeed filters [AD+97], or filters of unsuitable content [CA+00]. A filtering system should block the delivery of the documents the consumer is likely not interested in. Filtering is a case of binary TC, since it involves the classification of incoming documents in two disjoint categories, the relevant and the irrelevant. Additionally, a filtering system may also further classify the documents deemed relevant to the consumer into thematic categories of interest to the user.

The explosion in the availability of digital information has boosted the importance of such systems, which are nowadays being used in diverse contexts such as the creation of personalized Web newspapers, junk e-mail blocking, and Usenet news selection.

1.4.3 Word Sense Disambiguation

Word sense disambiguation (WSD) is the activity of finding, given the occurrence in a text of an ambiguous (i.e. polysemous or homonymous) word, the sense of this particular word occurrence. For instance, bank may have (at least) two different senses in English, as in the Bank of England (a financial institution) or the bank of river Thames (a hydraulic engineering artifact). It is thus a WSD task to decide which of the above senses the occurrence of bank in ‘Last week I borrowed some money from the bank’ has. WSD may be seen as a (single-label) TC task (see e.g. [EM00]) once, given a word w, we view the contexts of occurrence of w as documents and the senses of w as categories.

WSD is very important for many applications, including natural language processing, and indexing documents by word senses rather than by words for IR purposes. WSD is just an example of the more general issue of resolving natural language ambiguities, one of the most important problems in computational linguistics.

1.4.4 Other Applications

Other applications that are not explicitly discuss are automatic indexing for Boolean IR system [BB63][FB75][GH71][HH73][MM61], speech categorization by means of a combination of speech recognition and TC [MK00][SS00], multimedia document categorization through the analysis of textual captions [SH00], author identification for literary texts of unknown or disputed authorship [FRS99], language identification for texts of unknown language [CT99], automated identification of text genre [KNS97], and automated essay grading [LLS98].

1.5 Related Work

Approaches for text categorization can be broadly divided into three classes: supervised, semi-supervised, and unsupervised approaches. Both supervised [SF02][YL99] and semi-supervised methods [BM98][JT99][NM+00] require a certain amount of manual data labeling. No labeled documents are instead required in the unsupervised approaches [GSD05][KS00][LL+04] which are mainly similarity-driven. Moreover, they treat the category names as symbolic labels without assuming additional knowledge about their meanings: the knowledge implied in the document set is involved in the inductive process that builds the final document classifier. The most popular algorithms used for the classifier construction include Rocchio's algorithm [SF02], regression models [YP97], K-nearest neighbor [YP97], Naive Bayes [WL04], SVM [WL04] [FG03] [JT98], Decision trees (e.g. C4.5 decision tree algorithm [JT98]), and neural networks [YL99] etc.

However, most algorithms may not be completely suitable when the problem of high dimensionality occurs[YP97] [FG03], as even a moderately sized text collection often has tens of thousands of terms which make the classification cost prohibitive for many leaning algorithms that do not scale well to large problem sizes. In addition, it is known that most terms are irrelevant for the classification task and some of them even introduce noise that may decrease the overall performance [SM83].

Applying dimensionality reduction techniques (i.e. feature selection or feature extraction) is beneficial for the increasing scalability, reliability, efficiency and accuracy of text classification algorithms [LB92].

Text categorization applications generally have massive data samples and features, which makes wrapper methods rather time-consuming and impractical for these applications [TF07]. For this reason, the use of faster and simpler filter approaches is prominent in the domain [FG03][FS02][YP97][WL04][OO06]. Examples of hybrid techniques have been recently explored and have shown promising results [AV+08][RP+09][LW10][U11].

2 Word Sense Disambiguation

This section presents methods for the Word Sense Disambiguation to give a general framework about this topic.

2.1. Introduction

Human language is ambiguous, so that many words can be interpreted in multiple ways depending on the context in which they occur [NR09]. For instance, consider the following sentences:

1) I can hear bass sounds.
2) They like grilled bass.

The occurrences of the word bass in the two sentences clearly denote different meanings: low-frequency tones and a type of fish, respectively. Unfortunately, the identification of the specific meaning that a word assumes in a context is only apparently simple. Humans do not have to think about the ambiguities of language. Machines need to process unstructured textual information, transform them into data structures and analyze data in order to determine the underlying meaning. Word Sense Disambiguation (WSD) is the computational identification of meaning for words in context. For instance, the above sentence (2) should be ideally sense-tagged as "They like$_{/\ \text{ENJOY}}$ grilled$_{/\ \text{COOKED}}$ bass$_{/\ \text{FISH}}$." to be understood in non ambiguous fashion.

WSD heavily relies on knowledge. The skeletal procedure of any WSD system can be summarized as follows: given a set of words (e.g., a sentence or a bag of words), a technique is applied which makes use of one or more sources of

knowledge to associate the most appropriate senses with words in a context. Knowledge sources can vary considerably ranging from corpora (i.e., collections) of texts, either unlabeled or annotated with word senses, to more structured resources, such as machine-readable dictionaries, semantic networks, etc. For example, without additional knowledge, it would be very hard for both humans and machines to identify the meaning of the above sentences.

Unfortunately, the manual creation of knowledge resources is an expensive and time-consuming effort [NT97], which must be repeated every time the disambiguation scenario changes (e.g., in the presence of new domains, different languages, and even sense inventories). This is a fundamental problem which pervades the field of WSD, and is called the knowledge acquisition bottleneck [GCY92].

The exponential growth of the Internet community, together with the fast pace development of several areas of information technology (IT), has led to the production of a vast amount of unstructured data, such as document warehouses, Web pages, collections of scientific articles, blog corpora, etc. As a result, there is an increasing urge to treat this mass of information by means of automatic methods. Traditional techniques for text mining and information retrieval reveal their limits when applied to a huge collections of data. Mostly based on lexicosyntactic analysis of text, these approaches do not go beyond the surface appearance of words and, consequently, fail in identifying relevant information formulated with different wordings and in discarding documents which are not pertinent to the user needs.

The results of recent comparative evaluations of WSD systems - mostly concerning a stand-alone assessment of WSD - show that most disambiguation methods have inherent limitations in terms, among others, of performance and generalization capability when fine-grained sense distinctions are employed. On

the other hand, the increasing availability of wide-coverage, rich lexical knowledge resources, as well as the construction of large-scale coarse-grained sense inventories, seems to open new opportunities for disambiguation approaches.

2.2. Task Description

Given a text T represented by a sequence of words (w_1, w_2, ... , w_n), the WSD is the task of assigning the appropriate sense(s) to all or some of the words in T. WSD can be viewed as a classification task where word senses are the classes, and an automatic classification method assigns each occurrence of a word to one or more classes based on the evidence from the context and from external knowledge sources.

An important difference between TC and WSD is that the former uses a single pre-defined set of classes, whereas in the latter the set of classes typically changes depending on the word to be classified. In this respect, WSD consists on n distinct classification tasks, where n is the size of the lexicon.

We can distinguish two variants of the generic WSD task[NR09]:

- Lexical sample (or targeted WSD), where a system is required to disambiguate a restricted set of target words usually occurring one per sentence;
- All-words WSD, where systems are expected to disambiguate all open-class words in a text (i.e., nouns, verbs, adjectives, and adverbs). Approaches, such as knowledge-lean systems rely on full-coverage knowledge resources, are used, whose availability must be assured.

There are four main elements of WSD:

  - the selection of word senses (i.e., classes),

- the use of external knowledge sources,
- the representation of context,
- the selection of an automatic classification method.

2.2.1. Selection of Word Sense

A word sense is a commonly accepted meaning of a word. For instance, consider the following two sentences:

1) She chopped the vegetables with a chef's knife.
2) A man was beaten and cut with a knife.

In the above sentences, the word knife is used with two different senses: a tool (a) and a weapon (2). The two senses are clearly related, as they possibly refer to the same object; however the object's intended uses are different. This example makes it clear that determining the sense inventory of a word is a key problem in word sense disambiguation.

A sense inventory partitions the range of meaning of a word into its senses. Word senses cannot be easily discretized, that is, reduced to a finite discrete set of entries, each encoding a distinct meaning. The main reason for this difficulty stems from the fact that the language is inherently subject to changes and interpretations. Also, given a word, it is arguable where one sense ends and the next begins.

2.2.2. Use of External Knowledge Sources

Knowledge is a fundamental component of WSD. Knowledge sources provide data which are essential to associate senses with words. Knowledge sources include:

Structured resources include:

- Thesauri, which provide information about relationships between words, like synonymy (e.g., car n is a synonym of motorcar n), antonymy (representing opposite meanings, e.g., ugly a is an antonym of beautiful a) and, possibly, further relations [KY00].
- Machine-readable dictionaries (MRDs), which have become a popular source of knowledge for natural language processing since the 1980s, when the first dictionaries were made available in electronic format. Nowadays, WordNet [MB+90][FC98] is the most prominent MRD for word sense disambiguation in English.
- Ontologies are specifications of conceptualizations of specific domains of interest [GTR93], usually including a taxonomy and a set of semantic relations. In this respect, WordNet can be considered as an ontology. A dept WordNet description is in Appendix A.

Unstructured resources includes:

- Corpora, that is, collections of texts used for learning language models. Corpora can be sense-annotated or raw (i.e., unlabeled).
- Collocation resources, which register the tendency for words to occur regularly with others.

2.2.3. Representation of Context

As text is an unstructured source of information, to make it a suitable input to an automatic method it is usually transformed into a structured format. To this end, a preprocessing of the input text is usually performed, which typically (but not necessarily) includes the following steps:

- tokenization, a normalization step, which splits up the text into a set of tokens (usually words);

- part-of-speech tagging, consisting in the assignment of a grammatical category to each word (e.g., "the$_{/DT}$ bar$_{/NN}$ was$_{/VBD}$ crowded$_{/JJ}$," where DT, NN, VBD and JJ are tags for determiners, nouns, verbs, and adjectives, respectively);
- lemmatization, that is, the reduction of morphological variants to their base form (e.g. was → be, bars → bar);
- chunking, which consists of dividing a text in syntactically correlated parts (e.g., [the bar] NP [was crowded] VP, respectively the noun phrase and the verb phrase of the example).
- parsing, whose aim is to identify the syntactic structure of a sentence (usually involving the generation of a parse tree of the sentence structure).

A set of features is chosen to represent the context. These include (but are not limited to) information resulting from the above-mentioned preprocessing steps, such as part-of-speech tags, grammatical relations, lemmas, etc. We can group these features as follows:

- local features, which represent the local context of a word usage, that is, features of a small number of words surrounding the target word, including part-of-speech tags, word forms, positions with respect to the target word, etc.;
- topical features, which - in contrast to local features - define the general topic of a text or discourse, thus representing more general contexts (e.g., a window of words, a sentence, a phrase, a paragraph, etc.), usually as bags of words;
- syntactic features, representing syntactic cues and argument-head relations between the target word and other words within the same sentence (note that these words might be outside the local context);

- semantic features, representing semantic information, such as previously established senses of words in context, domain indicators, etc.

2.2.4. Choice of a Classification Method

The final step is the choice of a classification method. Most of the approaches to the resolution of word ambiguity stem from the field of machine learning.

We can broadly distinguish two main approaches to WSD:

- supervised WSD: these approaches use machine-learning techniques to learn a classifier from labeled training sets, that is, sets of examples encoded in terms of a number of features together with their appropriate sense label (or class);
- unsupervised WSD: these methods are based on unlabeled corpora, and do not exploit any manually sense-tagged corpus to provide a sense choice for a word in context.

We can also distinguish between knowledge-based (or knowledge-rich, or dictionary-based) and corpus-based (or knowledge-poor) approaches. The former rely on the use of external lexical resources, such as machine-readable dictionaries, thesauri, ontologies, etc., whereas the latter do not make use of any of these resources for disambiguation.

Finally, we can categorize WSD approaches as token-based and type-based. Token-based approaches associate a specific meaning with each occurrence of a word depending on the context in which it appears. In contrast, type-based disambiguation is based on the assumption that a word is consensually referred with the same sense within a single text.

I will focus only on knowledge-based disambiguation because I use this approach in this thesis.

2.3. Knowledge-Based Disambiguation

The objective of knowledge-based or dictionary-based WSD is to exploit knowledge resources (such as dictionaries, thesauri, ontologies, collocations, etc.) to infer the senses of words in context [NR09]. Related methods usually have lower performance than their supervised alternatives, but they benefit from a wider coverage, thanks to the use of large-scale knowledge resources.

The first knowledge-based approaches to WSD date back to the 1970s and 1980s when experiments were conducted on extremely limited domains, as the lack of large-scale computational resources prevented a proper evaluation, comparison and exploitation of those methods in end-to-end applications.

Knowledge-Based Approaches

The simplest and intuitive knowledge-based approach relies on the calculation of the word overlap between the sense definitions of two or more target words. This approach is named gloss overlap or the Lesk algorithm after its author [LM86]. Given a two-word context (w_1, w_2), the senses of the target words is assumed to be the sense definition having the highest definitions overlap.

A variant of the Lesk algorithm is currently employed which identifies the sense of a word w whose textual definition has the highest overlap with the words in the context of w. The context of w is the bag of all content words in a context window around the target word w.

As an example, in Table 2.1 is shown the first three senses in WordNet of the word “key”, namely key_n, and mark in italic the words which overlap with the following input sentence: I inserted the key and locked the door.

Table 2.1. WordNet Sense Inventory for the First Three Sense of key_n

Sense	Definition an Examples
key_n^1	Metal device shaped in such a way that when it is *inserted* into the appropriate *lock* the lock's mechanism can be rotated
key_n^2	Something crucial for explaining; "the key to development is economic integration"
key_n^3	Pitch of the voice; "he spoke in a low key"

Sense 1 of "key" has 3 overlaps, whereas the other two senses have zero, so the first sense is selected.

The original method achieved 50–70% accuracy (depending on the word), using a relatively fine set of sense distinctions such as those found in a typical learner's dictionary [LM86]. Unfortunately, Lesk's approach is very sensitive to the exact wording of definitions, so the absence of a certain word can radically change the results.

Structural Approaches

Since the availability of computational lexicons like WordNet, a number of structural approaches have been developed to analyze and exploit the structure of the concept network made available in such lexicons. The recognition and measurement of patterns, both in a local and a global context, can be collocated in the field of structural pattern recognition [FK82][BS90], which aims at classifying data (specifically, senses) based on the structural interrelationships of features.

Similarity Measures

Since the early 1990s, when WordNet was introduced, a number of measures of semantic similarity have been developed to exploit the network of semantic

connections between word senses. Ranging in [0,1], the semantic similarity is a score, which is broadly defined by the following function:

$$\text{score} : \text{Senses}_D \times \text{Senses}_D \rightarrow [0, 1],$$

where Senses_D is the full set of senses listed in a reference lexicon. A general approach to disambiguate a target word w_i in a text $T = (w_1, \ldots, w_k)$ consists in choosing the sense $\hat{S}$ of w_i that corresponds to the maximum similarity between w_i and w_j ($j = i\text{-}N, \ldots, i\text{-}1, i+1, \ldots, i+N$) in the given text context.

In literature, the most popular measures of semantic similarity include the following approaches:

Rada et al. [R+89] introduced a simple metric based on the calculation of the shortest distance in WordNet between pairs of word senses.

Sussna's [SM93] approach is based on the observation that concepts located in depth in a taxonomy (e.g., limousine and car) appear to be more closely related to each another than concepts in the upper part of the same taxonomy (e.g., location and entity). An edge in the WordNet noun taxonomy is viewed as a pair of two directed edges representing inverse relations (e.g., kind-of and has-kind).

Inspired by Rada et al. [R+89], Leacock and Chodorow [LD98] developed a similarity measure based on the distance of two senses S_w and $S_{w'}$. They focused on hypernymy links and scaled the path length by the overall depth D of the taxonomy.

One of the issues of distance-based measures is that they do not take into account the density of concepts in a subtree rooted at a common ancestor. Agirre and Rigau [AR96] introduced a measure called conceptual density, which

measures the density of the senses of a word context in the sub-hierarchy of a specific synset.

Resnik [LD98] introduced the notion of information content shared by words in context. The proposed measure determines the specificity of the concept that subsumes the words in the WordNet taxonomy. It is based on the idea that, the more specific the concept that subsumes two or more words, the more semantically related they are assumed to be.

Jiang and Conrath's [JC97] approach also uses the notion of information content, expressed by the conditional probability of encountering an instance of a child sense given an instance of an ancestor sense. The measure takes into account the information content of the two senses, as well as that of their most specific ancestor in the noun taxonomy.

Finally, Lin's [LD98] similarity measure is based on the theory of similarity between arbitrary objects. It is essentially Jiang and Conrath's [JC97] measure, proposed in a different fashion.

Different similarity measures have been assessed in comparative experiments to determine which is the best measure. Budanitsky and Hirst [BH06] found that Jiang and Conrath's [JC97] measure is superior in the correction of word spelling errors compared to the measures proposed by Leacock and Chodorow [LD98], Lin [LD98], Resnik[LD98].

Pedersen et al [PBP05] made similar considerations and found that Jiang and Conrath outperforms the other measures in the disambiguation.

Most of the above-mentioned measures are implemented in the WordNet::Similarity package [PPM04].

2.4. Domain-Driven Disambiguation

Domain-driven disambiguation [GMS04][BM+06] is a WSD methodology that makes use of domain information. The sense of a target word is chosen based on a comparison between the domains of the context words and the domain of the target sense.

This approach achieves good precision and possibly low recall, due to the fact that domain information can be used to disambiguate mainly domain words. Domain information is represented in terms of domain vectors, that is, vectors whose components represent information from distinct domains. Given a word sense S, a synset vector is defined as $S = (R(D_1, S), R(D_2, S), \ldots , R(D_d, S))$, where D_i are the domains available ($i \in \{1, \ldots , d\}$) and $R(D_i, S)$ is defined as follows:

$$R(D_i , S) = \begin{cases} 1/ |Dom(S)| & \text{if } D_i \in Dom(S) \\ 1/d & \text{if } Dom(S) = \{ \text{FACTOTUM} \} \\ 0 & \text{otherwise} \end{cases}$$

where Dom(S) is the set of labels assigned to sense S in the WordNet Domain labels resource and the FACTOTUM label represents the absence of domain pertinence.

2.5. Evaluation Measures

The performance of word sense disambiguation systems is usually assessed by evaluation measures from the field of information retrieval, mainly including coverage, precision and recall.

Let $T = (w_1, \ldots , w_n)$ be a test set and A an "answer" function that associates with each word $w_i \in T$ the appropriate set of senses from the dictionary D. Given the sense assignments provided by an automatic WSD system, the coverage C is

the percentage of items in the test set for which the system provided a sense assignment that is:

$$C = \frac{\#\ answers\ provided}{\#total\ answers\ to\ provide}$$

The total number of answers is given by n=|T|.

The precision P is computed as the percentage of correct answers given by the automatic system, that is:

$$P = \frac{\#\ correct\ answers\ provided}{\#\ answers\ provided}$$

Precision determines the correctness of the given answers. The recall R is defined as the number of correct answers given by the automatic system over the total number of answers to be given:

$$R = \frac{\#\ correct\ answers\ provided}{\#\ total\ answers\ to\ provide}$$

According to the above definitions $R \leq P$. When coverage equals 100%, we have that P = R. In the WSD literature, recall is also referred to as accuracy, although these are two different measures in the machine learning and information retrieval literature.

Finally, the F_1-measure, or balanced F-score, evaluates the weighted harmonic mean of precision and recall and it is defined as:

$$F_1 = \frac{2PR}{P+R}$$

2.6. Baseline

A baseline is a standard method for comparing performance of different approaches. Here we present two basic baselines, the random baseline and the

first sense baseline. Other baselines have also been employed in the literature, such as the Lesk approach.

Let D be the reference dictionary and a test set $T = (w_1, w_2, \ldots, w_n)$ be a test set such that words w_i ($i \in \{1, \ldots, n\}$) are in the corpus. The chance or random baseline consists in choosing randomly a sense from available senses for each word w_i. Under the uniform distribution, for each word w_i the probability of success of such a choice is $1/|\text{Senses}_D(w_i)|$.

The first sense baseline (or most frequent sense baseline) relies on ranking word senses and choosing the first sense according to such a ranking.

For instance, in WordNet ranking, senses of the same word are based on the occurrence of each sense in the SemCor corpus. SemCor corpus description is in Appendix A.

2.7. Related Work

Recent research efforts [LZL09] [LZL12] attempt to explore the use of external semantic resources to automatically generate a set of representative words that properly describe the categories' meanings.

Based on the assumption that external knowledge can improve text categorization performance, a number of approaches have been proposed to incorporate extended features extracted from WordNet [BWL10][KP+03] [LCX11][PC05][MH06] and Wikipedia [GM05] [GM06] [WD08] into the text representation. Also, lexical databases like WordNet and its non-English counterparts (e.g. EuroWordNet, CoreNet and HowNet) have been applied in a variety of text processing tasks, such as document clustering [HSS03], word sense disambiguation [BP02][ZGW05], web search improvement [MM00], cross-lingual question answering [FT+09].

Recently, research on automatic text categorization has attempted to fully exploit existing knowledge represented in a domain ontology, using the ontology itself as the document classifier [JKA08][JKB08]. This approach requires a transformation of the document content into a graph structure: the categorization is based on measuring the semantic similarity between the created graph and the categories defined in the ontology. Moreover, similarly to WordNet, a domain ontology can be employed for vocabulary unification and word sense disambiguation, as discussed in [BH04]. Although originally not designed as an ontology, Wikipedia provides a comprehensive knowledge base for deriving RDF-based ontological descriptions [AL07] that can be successfully employed for enhancing automatic text categorization tasks [JKB08].

3 The Genetic Wrapper Model

This chapter presents and discusses in detail a hybrid model for the text categorization that combines and take advantage of two feature selection techniques: filter and wrapper. This combination helps to overcome their limitations and reach good results. In detail, the model uses a filter to rank the list of terms present in documents. Then, terms with the highest score values are selected, in an incremental way, resulting in a set of nested term subsets. The preliminary use of the filter ensures that useful terms are unlikely to be screened out. Differently from most filter-based approaches, the ranked list is not cut off using (somewhat arbitrary) threshold value. To limit classification problems due to the correlation among terms, the approach considers refining the selection process by employing a wrapper that uses a Genetic Algorithm (GA) as search strategy. Unlike traditional wrappers that select the features linearly, the GA performs a random terms combination and shows its potentiality in exploring features set of high dimensionality. From now, the above hybrid model is referred as Genetic Wrapper Model (GWM).

In the following, the steps of GWM are detailed and results are prominent about its application to a popular benchmark dataset.

3.1. The GWM Steps

GWM addresses a multi-label TC problem by resolving |C| binary problems. The model first selects the most representative terms for a given category c_i and then performs a binary classification process on this selection.

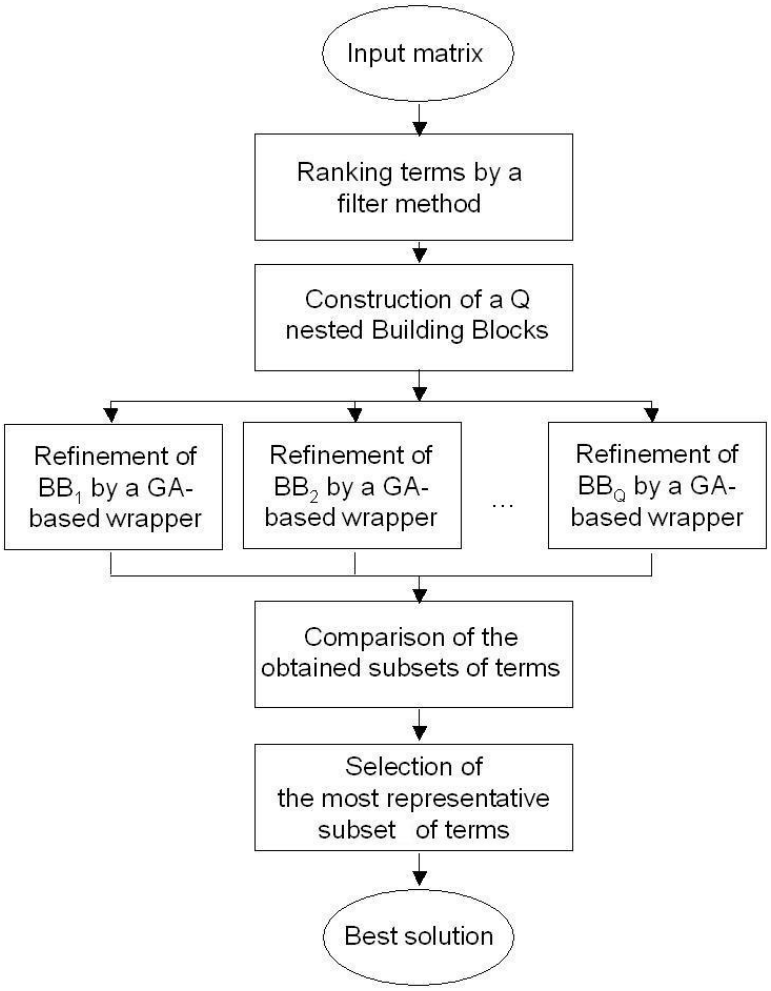

Figure 3.1. Steps of the GWM

Figure 3.1 shows the basic steps of GWM. The model input, i.e. the training set, is a matrix where each row represents a document d_j and columns are the related terms $\{w_1, w_2, \ldots, w_M\}$. Each document is assigned to either the category c_i or its complement $\bar{c}_i$.

First, a filter method assesses the scores of individual terms according to their power in discriminating c_i. This results in an ordered list where terms appear in descending order of relevance. The aim is to guide the term research at initial stage and ensure that useful terms are unlikely to be discarded.

Fixed a threshold value R, different term subsets of increasing size, namely Building Blocks (BBs), are progressively constructed by adding to the first R terms of the ordered list, additional terms less and less correlated with the category. It results in a sequence of Q nested BBs:

$$BB_1 \subset BB_2 \subset BB_3 \subset \ldots \subset BB_Q$$

where BB_1 includes the first R top-ranked terms, BB_2 includes the first 2*R top-ranked term, etc.

Then, such BBs are refined by a wrapper that uses a GA as search strategy, with the intent of removing redundant terms and obtaining more accurate and small-sized subsets of terms for categorization. Specifically, for each BB_i (i=1, …, Q), the GA initializes a population of individuals randomly, each individual being codified by a binary vector whose dimension equals the size of the BB. In the binary vector, the value 1 means that the respective term is selected, otherwise the value is 0. A fitness function evaluates the individuals by means of a classifier and selects the individuals that maximize the classification accuracy. Then, the current population undergoes genetic operations (i.e. selection, mutation, and crossover) and a new population is generated and evaluated. This evolution process is repeated within a pre-defined number of generations and it outputs the best individual, i.e. the subset of terms that best categorizes the BB.

Using popular metrics, the accuracy of solutions expressed for each BB are evaluated and compared with a test set. The solution with the highest value of accuracy is selected, which consists of the subset of terms that best categorizes the given category c_i.

3.2. Important Aspects

It’s important to underline some keys aspects of the model which are determinant to reach good performance.

First of all, the model is independent from a specific implementation. Different feature selection algorithm could be chosen, and different classifiers could be used, on the basis of the specific case. This means that, the approach is not closely related to a particular implementation, but is only a procedural schema to follow.

The method is hybrid because it take advantage both from filter-based and wrapper-based feature selection, with the aim of overcoming their limitation. In detail, the preliminary aim of the filter is to guide the features research at the initial stage, ensuring that useful features are unlikely to be discarded. At this point, the successive use of the wrapper permits to refine the previously subspaces in that reducing also the computational cost required from the wrapper approach. The intent is to remove redundant features and obtain more accurate and small-sized predictors for the classification. This double feature selection step, therefore, makes more efficient the use of a wrapper.

Furthermore, after the use of a filter, the method applies several thresholds of increasing size to originate Building Blocks. These BB explorations aims to discover a potentially high number of solutions.

In addition, the wrapper employs a GA as search strategy. Greedy procedures or heuristics are usually employed to guide the combinatorial search through the space of feature subset, with the aims of find a good compromise between performance and computational cost in high dimensional set.

3.3. Experiments

The performance of the method has been evaluated by performing two classes of experiments:

1. Baseline experiments. A classifier is trained directly on every Building Block without applying the wrapper approach. The related classification performance is considered as baseline.
2. GWM experiments. GWM is applied and the classification performance is compared with baseline experiments.

3.3.1. Dataset

Experiments were conducted on Reuters-21578 text collection [LD97], a benchmark which consists in 21,578 news stories published by Reuters in 1987, classified according to 135 categories mostly concerning business and economy. The creators of the collection defined standard splits to create various subsets of the corpus and different splits have been used by researchers to test their systems. The Mod-Apté split is the most used, and it consists of 9,603 training documents and additional 3,299 test documents from 90 categories.

Table 3.1. R10 Categories

R10	
Category	No. of terms
acq	7,495
corn	8,302
crude	14,466
earn	9,500
grain	12,473
interest	10,458
money-fx	7,757
ship	9,930
trade	7,600
wheat	8,626

Presented experiments consider using the Mod-Apté split, and the dataset as pre-processed in [PP+08], which considers the 10 categories with the highest number of positive training examples. In the following I will refer to this subset as R10. For each category in R10, the GWM input (i.e. the training set) is a matrix where each row represents a document d_j and columns are the related …..terms $\{w_1, w_2, \ldots, w_M\}$. Each document is assigned to either the category c_i or its complement $\bar{c}_i$. Table 3.1 shows the number of terms for each category in R10.

3.3.2. Parameter Setup

For ranking, I experimented the filters χ^2 (CHI) and Information Gain (IG) because these are the most used approaches in literature. Hence, I implemented two versions of the method which differ in the choice of the filter technique, namely GWM(CHI) and GWM(IG).

For building the nested Building Blocks, I set R=10 and Q=10 in that considering the first 100 top-ranked terms. I also considered two additional BBs having size 150 and 200.

The wrapper is based on the GA search mechanism as proposed by Goldberg [GDE89]. Leveraging on previous studies about tuning GA parameters [CDP10], the set up values are the following: population size = 30, crossover probability = 1, mutation probability = 0.02, number of generations = 50. Since the GA performs a stochastic search, I considered the results over 3 trials. The fitness function was the Naïve Bayes Multinomial classifier [MN98] for accuracy estimation.

The evaluation and comparison of the solutions obtained from each BB was evaluated using the following popular metrics [SF02]:

- F-measure, which expresses the harmonic mean between precision and recall;
- Break Even Point (BEP), which expresses the mathematical mean between precision and recall;
- μ-BEP, which permits a global evaluation of BEP values across categories.

These metrics were used within baseline and method experiments. Experiments evaluated the effectiveness of GWM using the following GWM configurations:

- GWM(CHI): Filter CHI + Genetic Wrapper + Naïve Bayes Multinomial classifier;
- GWM(IG): Filter IG + Genetic Wrapper + Naïve Bayes Multinomial classifier.

The overall analysis was implemented using the Weka data mining environment [H+09].

3.4. Results and Discussion

For each BB, I compared results on 3 trials and chose the solution with the highest F-measure as the best one. As Table 3.2 shows, the best solution, in terms of both F-measure and number of selected terms, does not significantly differ from the corresponding averaged values. Table 3.2 details only results obtained by GWM(IG) for the category grain, similar trends have been noticed for all the categories irrespective of the implementation of GWM. For this reason, in the following I will consider and report only the best F-measure values.

Table 3.2. Averaged and best values obtained with GWM(IG) on category grain

	Averaged Values		Best Values	
BB size	**F-measure**	**Selected Terms**	**F-measure**	**Selected Terms**
10	53,16	9	53,16	9
20	65,48	18	65,48	18
30	92,28	12	92,78	13
40	91,59	15	92,45	14
50	91,16	16	91,56	17
60	90,77	19	92,26	17
70	90,30	24	91,66	21
80	89,03	24	90,36	24
90	89,61	30	91,61	29
100	90,09	27	92,26	19
150	89,70	46	92,26	36
200	89,16	63	90,37	58

Figure 3.2 shows the best value of F-measure obtained for GWM within each BB. The GWM(IG) version results in very high values of F-measure compared to those obtained by GWM(CHI).

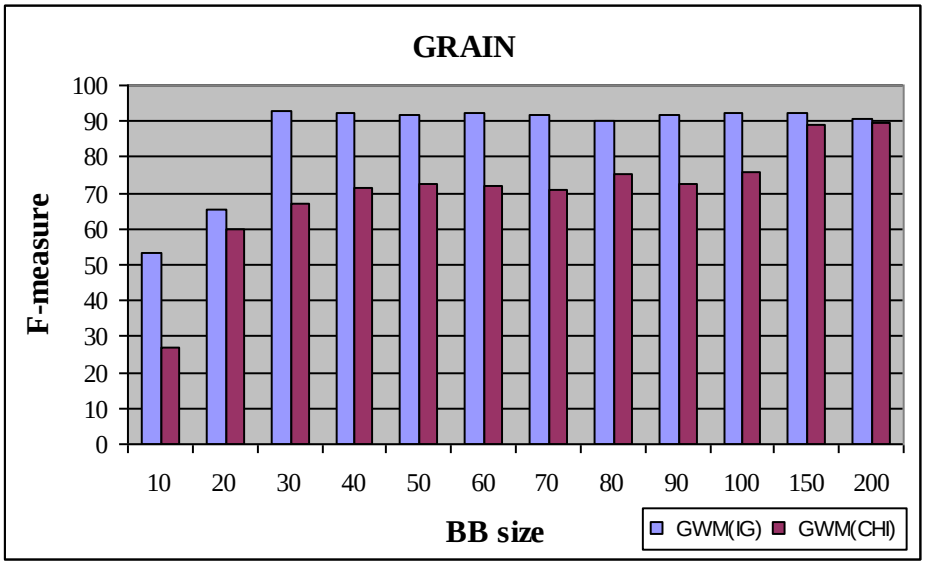

Figure 3.2. Best F-measure values obtained within each BB (cat. grain)

For each BB, Figure 3.3 shows the size of the solution expressed by the rate between the terms selected by the model and the respective size of the initial BB.

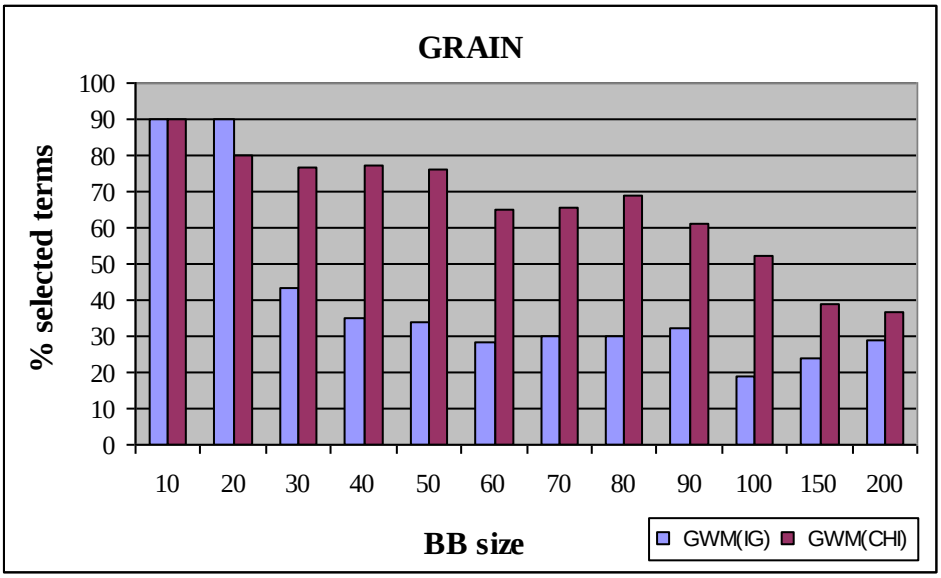

Figure 3.3. Percentage of selected terms from each BB (category grain)

The above results demonstrate that GWM(IG) is more specific than GWM(CHI) as it allows to select a lower number of terms.

To detail results, I report in the follow the solution obtained by GWM(IG) over the category grain, i.e. the subset of terms that best categorizes this category:

{ wheat, grain, tonnes, corn, maize, barley, rice, cts, program, company, shr, commodity, bushel }.

Table 3.3 compares the performance of the two GWM implementations obtained for the categories in R10. It shows that GWM(IG) is more effective than GWM(CHI) in all the categories, with the exception of the category earn. By using a different scale for F-measure, Figure 3.4 shows this small anomalous behavior.

Furthermore, Table 3.3 illustrates the performance of the proposed model in terms of BEP and computational time (using a 3.6 GHz AMD Phenom 4 GB RAM).

Table 3.3. F-measure value and related BEP value obtained for each category in R10

	GWM(IG)					GWM(CHI)				
Category	**BB size**	**Selected Terms**	**F-measure**	**BEP**	**Time (sec)**	**BB size**	**Selected Terms**	**F-measure**	**BEP**	**Time (sec)**
acq	200	105	90,36	**90,40**	115	200	107	88,46	88,55	117
corn	150	30	93,09	**93,20**	116	200	123	56,52	62,85	108
crude	50	33	86,52	**86,85**	64	200	111	79,91	80,75	95
earn	150	73	96,90	96,90	124	200	97	97,05	**97,05**	129
grain	30	13	92,79	**92,85**	60	200	73	89,82	90,05	115
interest	90	34	60,68	**60,70**	96	200	110	58,29	58,35	113
money-fx	150	69	66,51	**66,95**	125	200	111	63,21	63,70	148
ship	90	47	84,09	**84,10**	106	200	122	70,74	74,05	95
trade	60	30	67,29	**67,70**	77	200	101	60,48	63,00	110
wheat	40	5	90,81	**91,20**	75	150	98	59,29	65,80	92

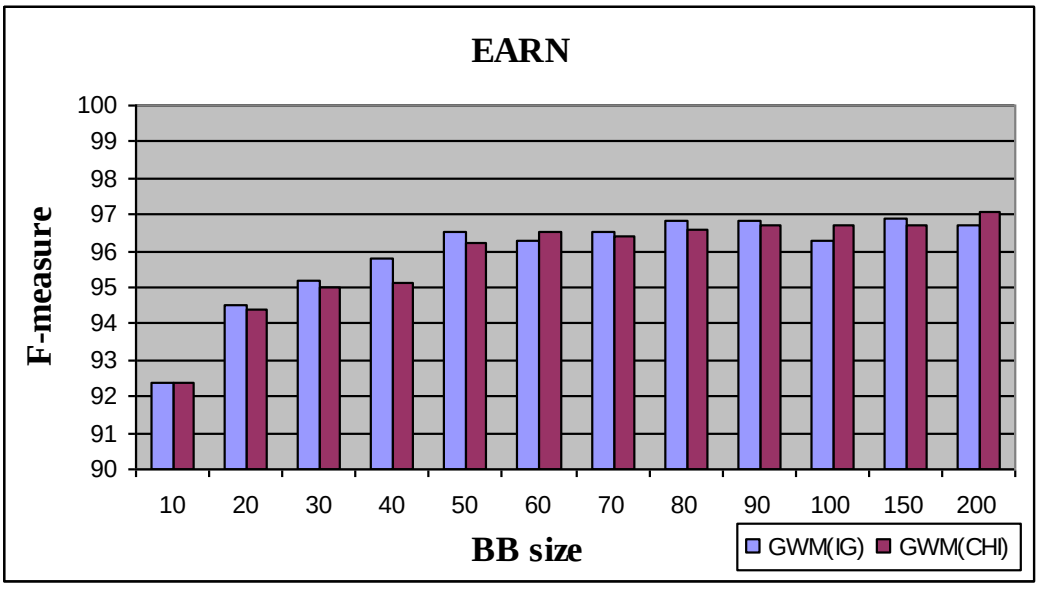

Figure 3.4. F-measure values obtained within each BB (category earn)

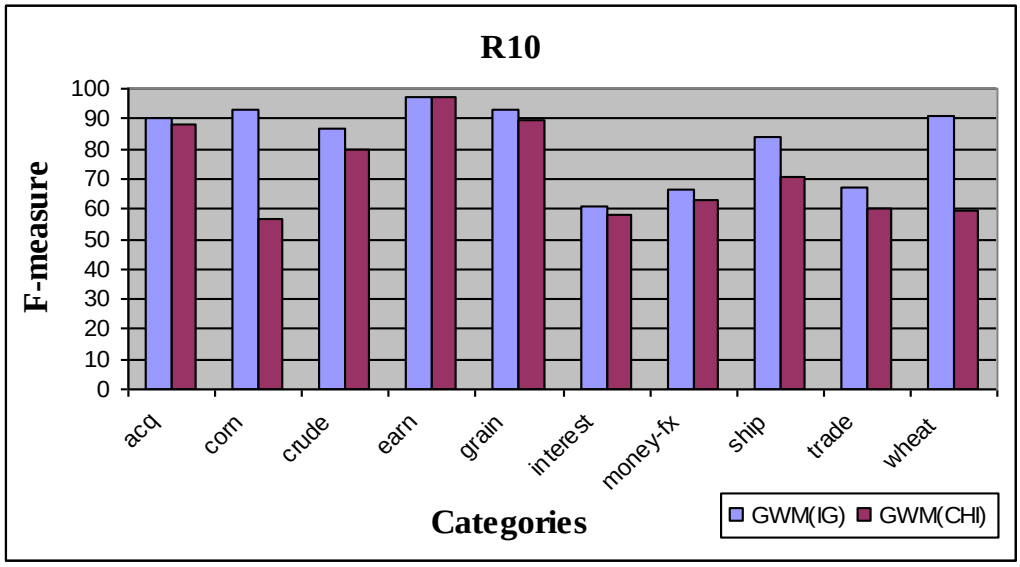

Figure 3.5. F-measure values for the two implementations (in R10)

From Table 3.3, Figure 3.5 shows the comparison between the F-measure values obtained by using GWM(IG) and GWM(CHI).

Because the GWM(IG) perform better then GWM(CHI), I consider only the former implementation for the further comparisons.

3.4.1. Comparison between Baseline and GWM

To have a first evaluation of the GWM, I first analyze the performance of the GWM compared with the baseline experiments.

Table 3.4 shows the F-measures obtained for every class of R10 from GWM and baseline experiments. As evidenced, GWM shows an increment of the classification performance.

This first comparison shows the effectiveness of GA approach in the wrapper.

Table 3.4. Baseline and GWM

	GWM (IG)			Baseline		Increment
Category	**BB size**	**Selected Terms**	**F-measure**	**BBsize**	**F-measure**	
acq	200	105	90.36	200	90.17	+ 0.21%
corn	150	30	93.09	150	39.70	+ 134.48%
crude	50	33	86.52	50	83.26	+ 3.92%
earn	150	73	96.90	150	95.47	+ 1.50%
grain	30	13	92.79	30	81.42	+ 13.96%
interest	90	34	60.68	90	54.47	+ 11.40%
money-fx	150	69	66.51	150	61.42	+ 8.29%
ship	90	47	84.09	90	75.75	+ 11.01%
trade	60	30	67.29	60	58.55	+ 14.93%
wheat	40	5	90.81	40	70.09	+ 29.56%

3.4.2. Comparison between GWM and literature

The GWM was compared with the following learning approaches proposed in the TC literature: Naïve Bayes, C4.5, Ripper, and SVM (both polynomial and radial basis function – rbf), plus two hybrid approaches named Olex-GA and Olex Greedy recently proposed in [RP+09]. Table 3.5 shows this comparison.

Results obtained by GWM(CHI), with a μ-BEP of 86.06, do not significantly emerge as they outperform only Naïve Bayes (82.52), C4.5 (85.82), and the hybrid approach Olex Greedy (84.80). Results from GWM(IG) compare well with the best results obtained from the other algorithms: with a μ-BEP equals to 89.06, this implementation of the model outperforms all the other approaches. Only the SVM poly (89.91) reached the best results, but this method is known to be computationally very expensive.

Although the comparison is based on the best results, for the sake of completeness Table 3.5 reports in brackets the corresponding average BEP values obtained from GWM(IG).

Table 3.5. Comparison using BEP and μ–BEP values

Category	Naïve Bayes	C4.5	Ripper	SVM		Olex		GWM	
				Poly	rbf	Greedy	GA	CHI	IG
acq	90,29	85,59	86,63	90,37	90,83	84,32	87,49	88,55	90,40 (89,93)
corn	59,41	86,73	91,79	87,16	84,74	89,38	91,07	62,85	93,20 (87,93)
crude	78,84	82,43	81,07	87,82	86,17	80,84	77,18	80,75	86,85 (83,58)
earn	96,61	95,77	95,31	97,32	96,57	93,13	95,34	97,05	97,05 (96,70)
grain	77,82	89,69	89,93	92,47	88,94	91,28	91,75	90,05	92,85 (92,35)
interest	61,71	52,93	63,15	68,16	58,71	55,96	64,59	58,35	60,70 (59,03)
money-fx	56,67	63,08	62,94	72,89	68,22	68,01	66,66	63,70	66,95 (63,82)
ship	68,68	71,72	75,91	82,66	80,40	78,49	74,81	74,05	84,10 (82,30)
trade	57,90	70,04	75,82	77,77	74,14	64,28	61,81	63,00	67,70 (64,33)
wheat	71,77	91,46	90,66	86,13	89,25	91,46	89,86	65,80	91,20 (90,78)
μ-BEP	**82,52**	**85,82**	**86,71**	**89,91**	**88,80**	**84,80**	**86,40**	**86,06**	**89,06 (88,03)**

3.5. Conclusions

This chapter has presented a model supporting TC problems. Specifically, the model selects the most representative terms for a given category and then performs a classification process on this selection. An extensive validation has been presented based on the standard data collection Reuters-21578. Experimental results confirm the effectiveness of the model which compares well with several learning algorithms used in the TC domain.

From a machine learning point of view, TC is a challenging research area as datasets consist of hundreds of thousands of documents characterized by tens of thousands of terms. This means that TC is a good benchmark for checking whether methods scale up to substantial sizes. Being an hybrid approach, the proposed model does not fall squarely under the classes of algorithms usually adopted to solving TC problems. Although many approaches have been proposed in TC literature, GA-based learning approaches have remained isolated attempts.

As such, the proposal seems to offer several research perspectives. First, results show that the hybrid approach used for term selection combines

effectiveness and efficiency as the initial use of a filter permits to reduce the computational cost of the GA- based wrapper.

Second, note that the choice of the specific filter is significant, as it notably influences the model performance. Results confirm what has been already observed in literature [FG03]: sometimes CHI presents erratic behavior in the TC domain. In contrast, in these experiments IG turned out to be incisive in conjunction with the evolutionary wrapper.

4 The Genetic Wrapper Model for Gene Summaries

In this chapter, I present a work made in collaboration with a biologist, who was interested in discovering putative relationships about genes. Detecting common functions in a set of genes is a key activity for life scientists in order to assess the significance of experimentally derived gene sets and prioritizing those sets that deserve follow-up. This interest is shifting the focus on data analysis from individual genes to families of genes that are supposed to interact each other in determining a pathological state or influencing the outcome of a single trait (i.e. a phenotype). Because of the large number of genes and their multiple functions, discovering computational methods to detect a set of functionally coherent genes is still a critical issue in bioinformatics [RM+10]. Biologists have dealt with these challenges in part by leveraging the biological principle commonly referred to as ‘‘guilt by association’’ (GBA) [OS00]. GBA states that genes with related functions tend to share properties such as genetic or physical interactions. For example, if two genes interact, they can be inferred to play roles in a common process leading to the phenotype.

My work started from several set of annotations resulting from annotated gene summaries with about 20 BioPortal [S6] ontologies. Compiled by expert curators and freely available on the Internet [S2], a gene summary is a short text about a single gene which describes functions and processes related to that gene.

For classification purposes, gene were grouped in families. A family consists of a list of unique gene symbols and define a structural domain. Genes are grouped into families when they perform similar functions and share often also a

significant degree of resemblance in the sequences of the DNA-Building blocks encoding for proteins that derive from these genes.

4.1. Classification

As such, biologists are interested in discovering which are the essential processes, namely concepts from now on, that characterize the gene interactions within a family. This was the purpose of my work, which explores the BOWs resulting from the annotation process in order to extract the above concepts. The idea was exploiting GWM for classify genes within families based on their annotations, and testing results given that we know the family each gene belongs to. In few words, the problem of discovering relationship among genes was formulated as a TC process on annotations about genes.

4.1.1. Preprocessing

Experiments considered 5 families, and 10 genes per family. As Table 4.1 (first column) shows, the number of annotations is very high as it contains duplicated and auto-generated text. To reduce this number, I refined the BOW's content using a list of stopwords which contains not specialized terms (i.e. gene, gene name, family, etc). Table 4.1 (second column) shows the effect of this reduction process. However, the approach was guided by the motivation of discovering a potentially low number of concepts within each family in order to guide the biologist in choosing the most important ones. In spite of this, the number of concepts within each family continued to be very high.

To further reduce this number I adopted a TC process for assigning annotations to one or more pre-defined category labels expressed by families.

The dominant approach to categorization process considers the employment of a general inductive process that automatically builds a classifier by learning, from a set of pre-classified documents, the characteristics of the categories [SF02].

Table 4.1. Number of concepts extracted for each family

FAMILIES	**Total Concepts**	**Refined Concepts**	**Percentage Reduction**
A Kinase	250	113	54.80%
Class BGPCR	247	153	38.06%
Homeobox	183	56	69.40%
Mapk	307	117	61.89%
MHC	215	58	73.02%

However, these algorithms may be not completely suitable to this case, as the text collection (i.e. the corpus of 50 summaries) has a moderate size and hundreds of terms (i.e. the refined concepts) most of which are irrelevant for the classification task and some of them even introduce noise that may decrease the overall performance.

To face this problem, I considered a single list which contains all the refined concepts (see Table 4.1, second column). After the elimination of duplicate concepts, the list resulted in 346 terms which were assumed to be the features of the categorization process.

4.1.2. GWM for Classification Purposes

Next step was applying the GWM to extract the most representative annotations for each family.

First, I constructed a reference matrix M(50x346) where rows represent summaries and, columns are the features. A generic term M(i,j) is equal to 1 if the term in the j-column belongs to the BOW of the summary at the i-row, to 0 otherwise.

Then, I applied the Information Gain to score features according to their discriminative power, i.e. their capacity of separating the five families. It resulted in an ordered list where features appear in descending order of relevance.

Because I was interested in obtaining an average number of 8 features per family, I considered only the first 40 elements of the scoring list and I reduced the size of the matrix M by considering only the columns which represent these elements.

As a family can be categorized by different groups of features, the use of a Genetic Algorithm [GDE89] is beneficial because the GA explores every possible solution and provides different best solutions. Additionally, it removes redundant terms and originates more accurate and small-sized subsets of terms for categorization.

According to GWM steps, I used the scoring list to construct nested subsets of features where the first set contained the first 3 top-ranked features and the remaining sets were built by progressively adding to the previous set the next feature in the list until obtaining a set which contained all the 40 features.

Leveraging on previous studies about tuning GA parameters [CDP210], I set the following values: population size = 30, crossover probability = 1, mutation probability = 0.02, number of generations = 50. Since the GA performs a stochastic search, I considered the results over 3 trials and a Naïve Bayes classifier [MN98] for evaluating the fitness. Using a 10-fold cross validation, I evaluate and compare solutions from each subset using the F-measure, a popular metric which rates the harmonic mean between precision and recall. The overall analysis was implemented using the Weka data mining environment [BF+10].

4.2. Results and discussion

Table 4.2 resumes the results. Specifically, the first column shows the number of perfect predictors within each family. A perfect predictor is a set of features (i.e. concepts) whereby the classifier reaches an F-measure equal to 1. The second column shows the total number of concepts belonging to these perfect predictors. As a single concept can belong to different predictors, I eliminate duplicates. The third column depicts the number of distinct concepts within the perfect predictors.

Table 4.2. Number of predictors and annotations selected for each family

FAMILIES	**Perfect Predictors**	**Total Annotations**	**Distinct Annotations**	**Expert Selections**
A Kinase family	13	56	19	9
Class BGPCR	10	55	20	11
Homeobox	18	76	26	8
MapK	3	11	7	1
MHC	2	6	6	3

A comparison of Table 4.1 with Table 4.2 shows a drastic reduction in the number of concepts that best represent a specific family. In particular, in MapK and MHC families only few concepts are enough to characterize the family. As well the number of concepts increases when the collaboration is more articulate, as it happens in the branched family Class_BGPCR.

According to a common practice in bioinformatics, concepts were further examined and refined by a domain expert. The last column in Table 4.2 shows the result of this refinement and Table 4.3 details the concepts within each family.

Table 4.3. Final list of annotations for each family

Family	**List of Annotations**
A Kinase	akap1, camp, extracellular adherence protein, flagellum, mitochondrion, protein kinase, protein kinase a, receptor, sperm
Class BGPCR	adenylate cyclase, adrenocorticotropic hormone, corticotropin releasing factor, cyclase, glucagon, homeostasis, hormone, microtubule associated protein, receptor, secretion, vasoactive intestinal peptide
Homeobox	anatomical structure morphogenesis, dwarfism, hindbrain, histogenesis, homeobox gene, lbx1, rhombencephalon, transcription factor
MapK	mitogen-activated protein kinase
MHC	peptide binding, receptor, tnfsf14

4.3. Conclusions

This section has presented an application of the Genetic Wrapper Model to categorize biological text. Previously text where subjected to a phase of annotation used by the GWM as input, while it outputs the most representative terms for every family.

These experiments confirm the effectiveness of the GWM in selecting the right annotation to help the human expert in the proper selection. Results show that the GWM is promising in other fields.

As future research I plan to scale up the experiments to much large gene organizations.

5 The Ontology Based Text Categorization Approach

This part of the thesis deal with exploring limitation of the classical Machine Learning approaches in Text Classification that I introduce in the follow.

The classifier performance depends heavily on the large amount of hand-labeled documents as they are the only source of knowledge for learning the classifier. Being a labor-intensive and time consuming activity, the manual attribution of documents to categories is extremely costly. To overcome these difficulties, semi-supervised learning techniques have been proposed that require only a small set of labeled data for each category [ZXJ07].

The problem is that all of these methods require a training set of pre-classified documents and it is often the case that a suitable set of well categorized (typically by humans) training documents is not available. This creates a serious limitation for the usefulness of the above learning techniques in operational scenarios ranging from the management of web-documents to the classification of incoming news into categories, such as business, sport, politics, etc.

Most important, text categorization should be based on the knowledge that can be extracted from the text content rather than on a set of documents where a text could be attributed to one or another category, depending on the subjective judgment of a human classifier.

Going beyond the above mentioned approaches, recent research introduced text categorization methods based on leveraging the existing knowledge represented in a domain ontology [BWL10]. The basic idea is to use an ontology

for providing a functionality that is similar to the knowledge provided by human experts with a manual document classification.

Ontologies are used as data-models or taxonomies to add a semantic structure to the text by annotating it with unambiguous topics about the thematic content of the document. The novelty is that ontology-based approaches are no dependent on the existence of a training set and rely solely on a set of concepts within a given domain and the relationships between concepts. One of the best known sources of external knowledge is WordNet [MGA95] [FC98], a network of related words, that organizes English nouns, verbs, adjectives and adverbs into synonym sets, called synsets, and defines relations between these synsets.

This chapter presents a text categorization approach designed firstly to consider the context of a word to disambiguate it sense and, secondly, to fully exploiting semantic resources. It employs the ontological knowledge not only as lexical support, but also for deriving the final categorization of documents in topics categories.

Specifically, the work relates to apply WordNet for selecting the correct sense of words in a document, and utilizes domain names in WordNet Domains [MC00] [BF+04] for classification purposes. Experiments show how the approach performs well in classifying a large corpus of documents.

5.1. The Approach

The method consists of three main steps:

1. Discovering the semantics of words in the document;
2. Disambiguating the words;
3. Categorizing the document.

5.1.1. Discovering the semantics of words in the document

This step finds out all the possible meanings (or senses) of a word in a document. Starting from a document represented by a vector d of its terms, i.e. $d = (t_1, t_2, \ldots, t_n)$, I adopt a popular approach to the analysis of unstructured text. It is based on the bag-of-words (BOW) paradigm that uses words as basic units of information. Disregarding grammar, a text is represented as a collection of words (i.e. the parts of speech (POS) as nouns, adjective, verbs, etc). The terms inside the BOW are suitably tagged for their POS. After the elimination of stopwords (conjunctions, propositions, pronouns, etc), the remaining words are used as concepts that represent the document.

Then, I used WordNet as the semantic resource that represents a set of concepts within the document, and the relationships between these concepts. It is a combination of dictionary and thesaurus that is more intuitively usable to support automatic text analysis.

WordNet provides the possible senses for a large number of words and additional knowledge (such as synonyms, hypernyms, hyponyms, etc) for each possible meaning of a word. The unique characteristic of WordNet is the presence of a wide network of relationships between words and meanings, including some compound nouns and proper nouns such as “credit card” and “Margareth Thatcher”.

Other work [KS09] [MC07] has confirmed that knowledge extracted from other semantic resources, such ODP [S5] and Wikipedia [BL+09], can facilitate text categorization. However, it has been observed [DM+97] that, being not structured thesauri as WordNet, these resources cannot resolve synonymy and polysemy directly, i.e. they have limits in disambiguating words.

The ontology entities (i.e. the concepts) occurring in the analyzed document are identified by matching document terms with entity literals (used as entity names) stored in WordNet. This process shifts the analysis focus from the terms occurring in a document to the entities and semantic relationships among them and produces a set of appropriate synsets from WordNet within each term. However, these synsets do not represent the unambiguous matching between the document terms and their sense, because multiple synsets can be identified by the same concepts. This drawback is motivated by the fact that documents often use synonyms, terms might be related to each other, or the term in one document is not well understood by WorldNet. In these circumstances, it is necessary to eliminate homonyms and polysemic words that negatively affect the categorization task.

5.1.2. Disambiguating the Word Sense

Usually denoted as Word Sense Disambiguation, this task aims to give the correct meaning to the ambiguous words, or to words with multiple meanings according to the context in which the word is used.

For each word w, assuming it has m senses or synsets (s_1, s_2, …, s_m), usually known as sense inventory, the WSD method selects only one correct sense in order to build the so-called Bag of Synsets (BOS) that univocally represents the ontological knowledge about the document. The semantic similarity between two terms is a function of distance between the terms in the WordNet hierarchical structure, but there is a wide variety of approaches for calculating semantic similarities of terms [GDJ13].

For WSD purposes, the method leverages on [BD+07] that proposes to disambiguate separately nouns, verbs, adjectives and adverbs using surrounding words in a sentence. The idea is selecting the most appropriate sense of w according to the semantic similarity between w and its context. For example, the

sense of word “star” in the sentence “the sun is a star that irradiates energy” is about an astronomic fact, while in the sentence “Marylin was a movie star” it is about an actress. In this case, it is possible for a human to select the correct sense. Therefore, the method tries to emulate this behavior by taking account the context in which the word appears.

Specifically, the context of each word w in a sentence is a 2N sized window which contains the N words that surround w to the left and the N words that surround w to the right. As the complexity of the disambiguation process can vary according to the size of N, I experimentally evaluated the optimal size of the context window. This approach is also used in [BEC13].

Setting N=0 means to analyze the first sense baseline, i.e. choosing the first sense that appears in the ontology, without considering a context for disambiguate the word. Taking the above example, in the sentence “Marylin was a movies star” the word “star” has two synsets in WordNet: astronomic fact and actress(movie star). Do not consider a context surround the word results in choosing the first sense, i.e. astronomic fact, that is not the correct sense. Maybe the method chooses the right sense, but it happens by chance. For these reasons, I did not consider N=0.

I tested the disambiguation process using four different similarity measures proposed by Jiang and Conrath [JC97], Lin [LD98], Resnik [RP95], and Leacock and Chodorow [LC98]. The first three measure fall in the category of information based methods that aim to give a measure of how specific and informative a term is. The semantic similarity between two terms is based on the information content of their lowest common ancestor node. As the occurrence probability of a node decreases when the layer of the node goes deeper, the lower a node in the hierarchy, the greater its information content. The Leacock and Chodorow measure falls in the category of methods based on the hierarchical

structure of an ontology. These methods typically measure the distance between nodes to quantify the similarity between two nodes in the directed acyclic graph of the ontology. Specifically, Leacock and Chodorow calculated the number of nodes in the shortest path between two terms and then scaled the number by the maximum depth of the ontology to quantify the relatedness of the terms.

Since each category of methods has its own advantages and disadvantages, I conducted experiments to choice the most suitable similarity measure for disambiguating words within their context windows.

Results will be presented in the follows.

5.1.3. Document Categorization

This step attributes categories to the documents resulting from previous process by considering their lexical annotations. The key part is the definition of the categories to be considered. As in the previous step, the method tries to emulate the behavior of human experts that manually label the documents as they have detailed knowledge about the document domain.

Instead of using labels or manually constructed catalogues, I rely on WordNet Domains [MC00] [BF+04], a lexical resource created in a semi-automatic way by augmenting WordNet with domain labels. I consider these labels as topics categories.

Specifically, a domain may include synsets of different syntactic categories and from different WordNet sub-hierarchies. Domains may group senses of the same word into homogeneous clusters, with the side effect of reducing word polysemy in WordNet.

Semantic domains are areas of human knowledge (such as POLITICS, ECONOMY, SPORT) exhibiting specific terminology and lexical coherence. The

label FACTOTUM clusters cases not classified by the other labels. As an example, Table 5.1 shows the WordNet Domains within the word "bank".

Table 5.1. WordNet Domains of the word "bank"

Sense Number	Synset (Gloss)	Domains
1	depository financial institution, bank, banking concern, banking company (a financial institution ...)	**Economy**
2	bank (sloping land ...)	**Geography, Geology**
3	bank (a supply or stock held in reserve...)	**Economy**
4	bank, bank building (a building...)	**Architecture, Economy**
5	bank (an arrangement of similar objects...)	**Factotum**
6	savings bank, coin bank, money box, bank (a container...)	**Economy**
7	bank (a long ridge or pile...)	**Geography, Geology**
8	bank (the funds held by a gambling house...)	**Economy, Play**
9	bank, cant, camber (a slope in the turn of a road...)	**Architecture**
10	bank (a flight maneuver...)	**Transport**

Using WordNet Domain, each document may be classified in one or more domains according to its relevance for these domains. Domains in top positions are considered more relevant for that document. Consequently, the categorization is independent from the existence of a training set and it relies solely on semantic resources as the ontology effectively becomes the classifier.

5.2. Experiments

5.2.1. Dataset

I used the dataset SemCor [ML+93], created by the Princeton University. SemCor is composed by 352 tagged-documents: 186 documents are sense-tagged for every POS, and only the verbs are sense-tagged in the remaining 166 documents. For experiments I used SemCor 2.1, and specifically I have considered the 186 documents that are sense-tagged for every POS. This complete dataset was assumed as test set to assert the precision of the method. Experiments are based on WordNet 2.1.

5.2.2. The application of the method

First, the Bag Of Words was built from the 186 documents of SemCor 2.1. Specifically, the POS, the lemma and the sense number of each word have been extracted, while ignoring no-tagged words, punctuations and stopwords. Finally, I obtained a total of 186 files, one for each document, that were used as input BOWs.

An example of BOW is as follows:

VB have 4

JJ overall 2

NN charge 10

…

In each line, the first term designates the POS (i.e. VV = verb, JJ = adjective, NN = noun), the second term is the word, and the final digit is the correct sense number of the word in WordNet. For example, the first line means that the word “have” is a verb and its sense number in WordNet is 4 (i.e. own, have, possess — have ownership of possession of).

Experiments disambiguated these 186 files using the approach proposed in the previous section. Specifically, it disambiguated separately nouns, verbs, adjectives and adverbs using four different methods to measure the semantic similarity between terms.

5.3. Results and discussion

The precision in disambiguating terms is calculated as the rate between the number of synsets that were correctly disambiguated by the algorithm and the total number of synsets in the BOS.

Figure 5.1 shows the overall precision reached by each method in disambiguating documents within contexts of increasing size. It is evident that the size of the context is an important parameter which greatly affects the disambiguation performance and it is also sensitive to the disambiguation method. Enlarging the context window introduces noise in disambiguation process and the minimum sized context is also the optimal one. As well, the method with the Jiang and Conrath measure outperforms the other implementations.

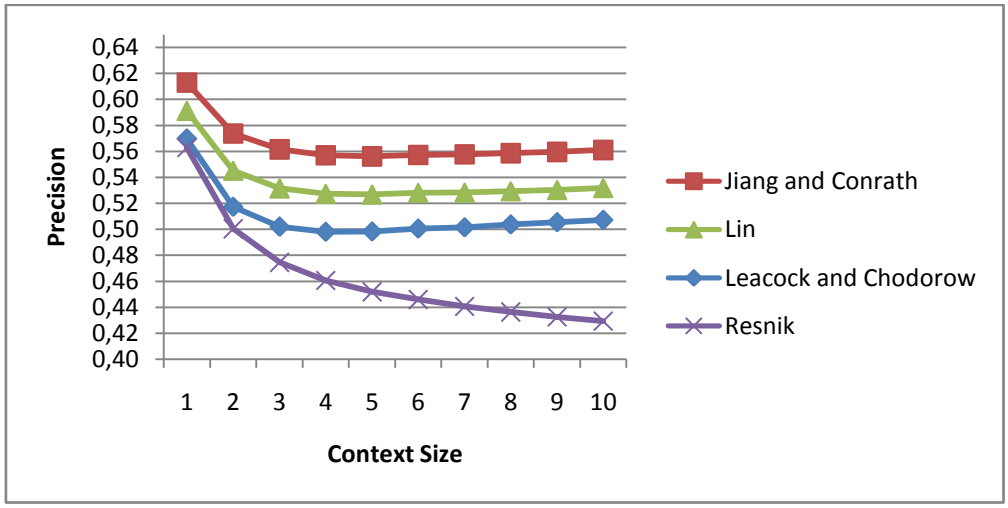

Figure 5.1. Overall precision of disambiguation methods within the context size

Figure 5.2 details the precision of the four implementations of the method only for nouns, and confirms previous results about the context window size.

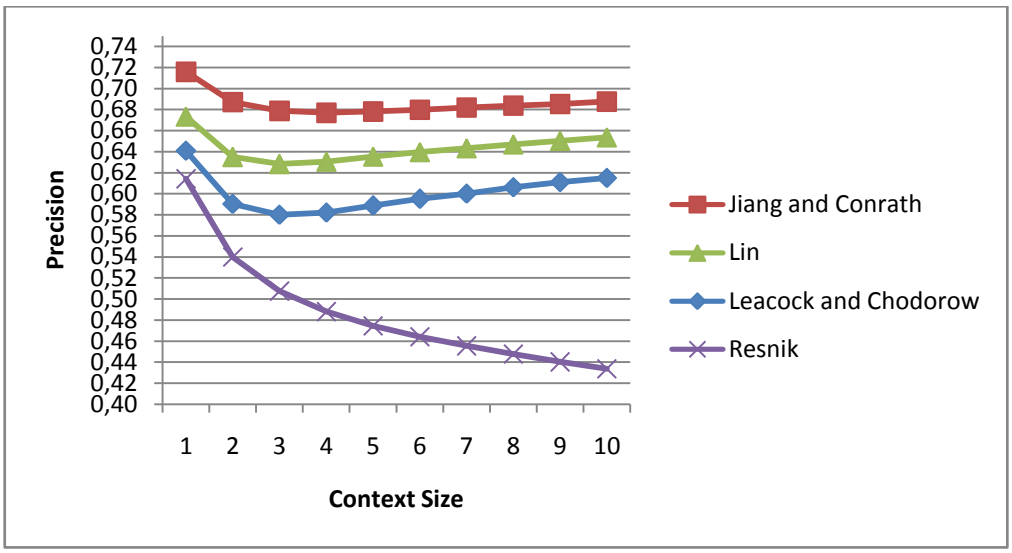

Figure 5.2. Overall precision of nouns reached by each disambiguation method

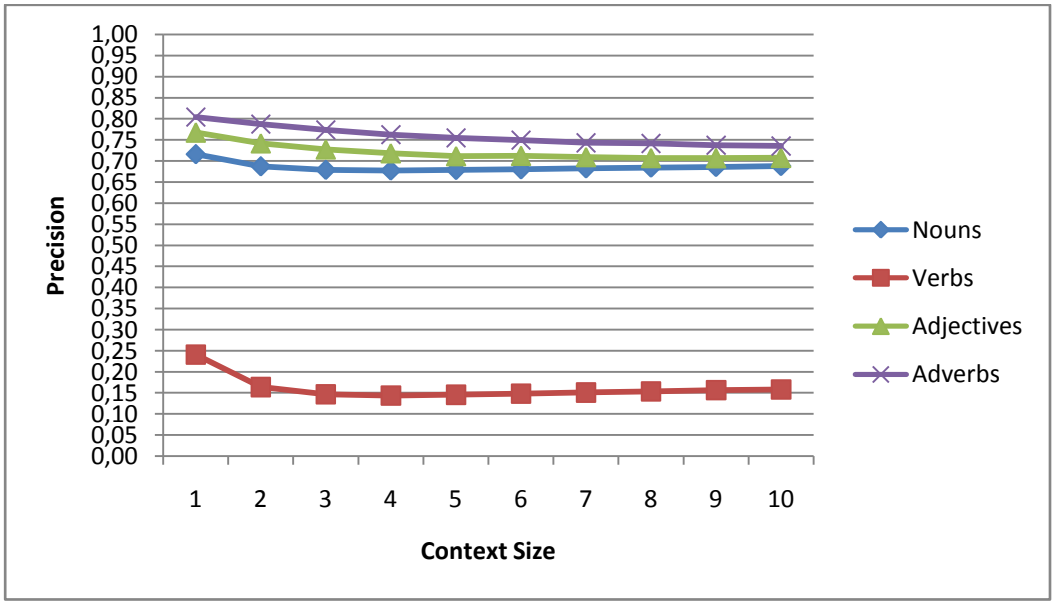

Figure 5.3. Overall precision reached by the method with the Jiang and Conrath measure for each POS

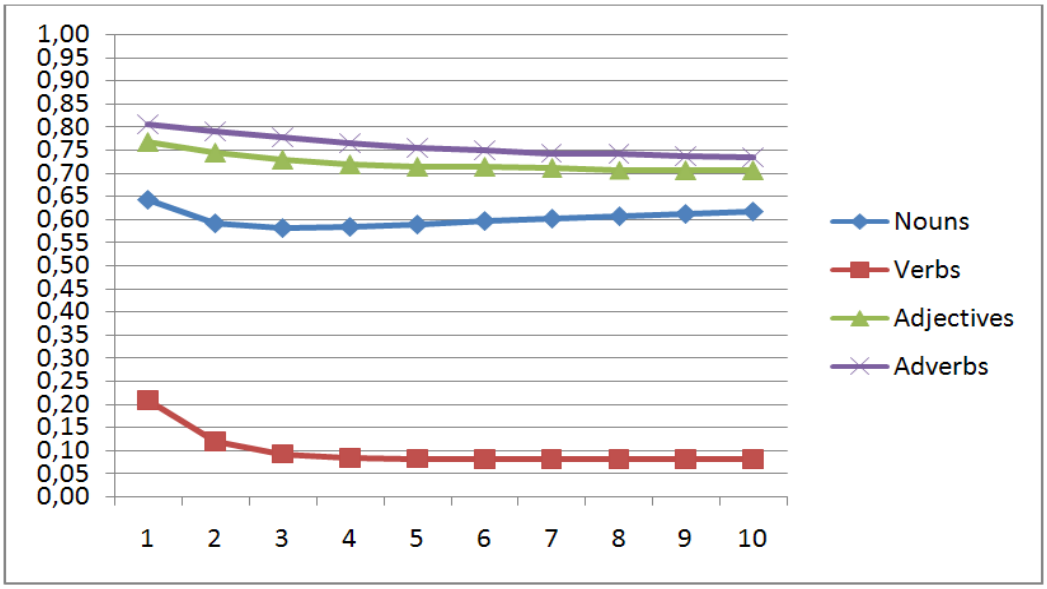

Figure 5.4. Overall precision reached by the method with the Leacock and Chodorow measure for each POS

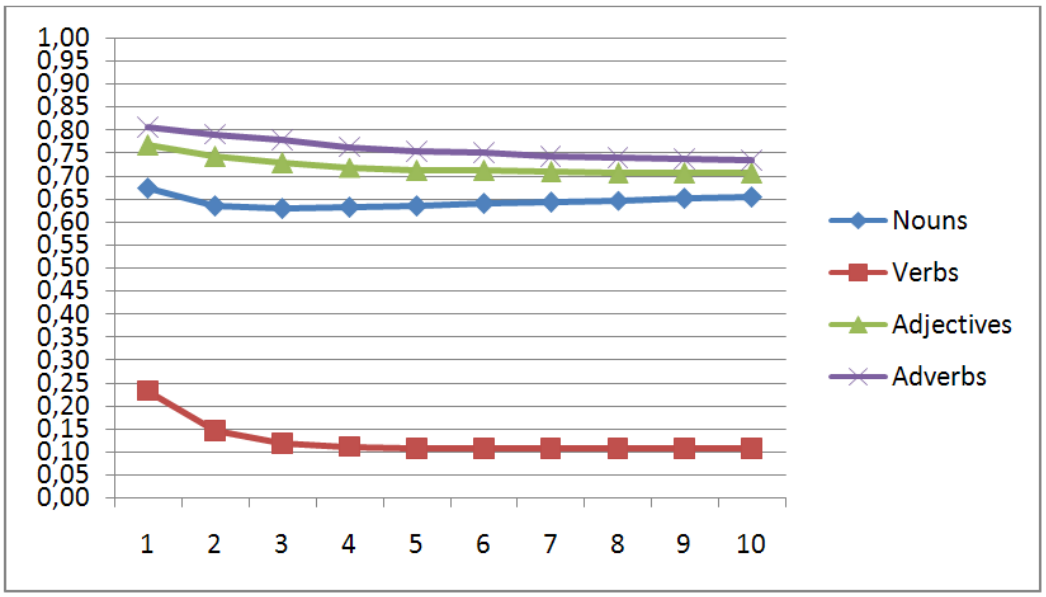

Figure 5.5. Overall precision reached by the method with the Lin measure for each POS

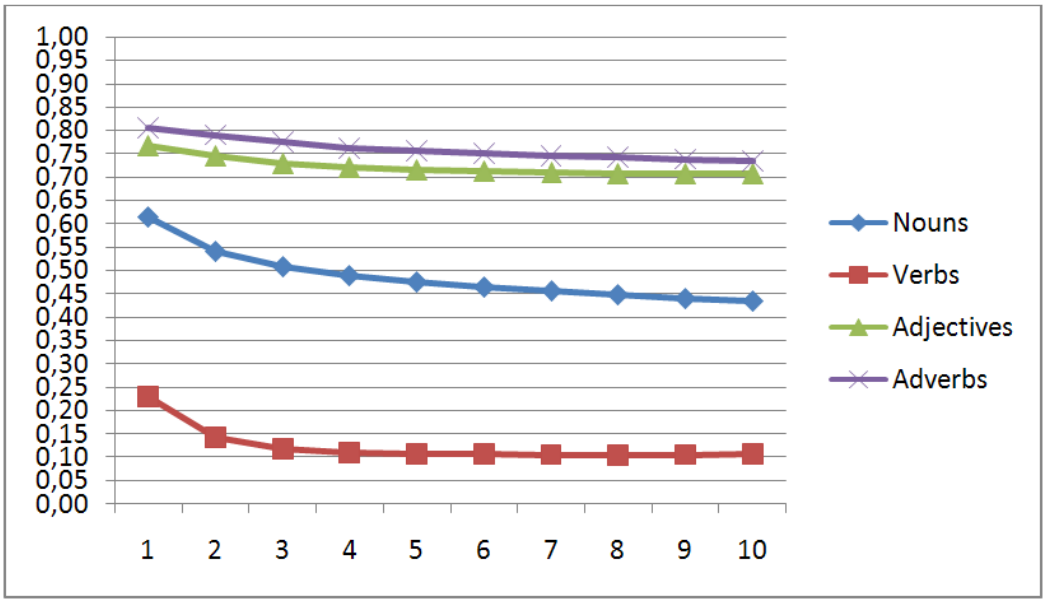

Figure 5.6. Overall precision reached by the method with the Resnik measure for each POS

Figure 5.3, Figure 5.4, Figure 5.5, and Figure 5.6 detail the precision reached by each implementation of the method (respectively with Jiang and Conrath, Leacock and Chodorow, Lin, and Resnik measure) in disambiguating each POS. It is evident that the best results are reached by the method with the Jiang and Conrath measure. In detail, as shown in Figure 5.3, adverbs are disambiguated better than other POS while disambiguating verbs results in a very low accuracy.

Finally, the classification step was performed by considering a BOW composed only by the disambiguated nouns. This choice was made because a high accuracy in disambiguating adjectives and adverbs is useful in other field, like sentiment analysis.

Over 186 documents, the proposed approach exactly classifies 144 documents into 3 different WordNet Domains. About the remaining 42 documents, the method correctly classifies 39 documents into 2 different domains, and only 1 domain was properly attributed to 3 documents.

Table 5.2 shows results about the classification of the first eight documents within the WordNet Domain. In brackets, the domain frequency, i.e. the number of nouns of the document that refer to that domain. The results of the classification for all documents are in Appendix B.

Table 5.2. Classification of the first 8 documents

Document	**1° Domain**	**2° Domain**	**3° Domain**
1	Politics (49)	Law (45)	Administration (39)
2	Music (24)	Person (22)	Metrology (21)
3	Politics (39)	Geography (21)	Anthropology (20)
4	Religion (69)	Psychological_features (25)	Mathematics (18)
5	Person (56)	Psychological_features (22)	Mathematics (21)
6	Person (31)	Administration (30)	Commerce (20)
7	Medicine (32)	Psychological_features (14)	Buildings (11)
8	Publishing (61)	Literature (50)	Linguistics (39)

5.4. Conclusions

This TC method has shown that the use of semantic relations terms drawing upon two kinds of thesauri, WordNet and WordNet Domains, results in a good accuracy. The approach is easy to implement (without document labeling efforts) and allows to cover multiple different domains within categorizing large documents sets.

The approach is very promising when applied in real world contexts where the growing body of documents makes them complex to be catalogued as it is the case of managing a large set of enterprise documents within this domain. A fully

semantic approach to text categorization can reduce the difficulty to retrieve and manage information in an automated manner, as the volume of data becomes unmanageable giving rise to inefficiencies and costs that are not easily measurable, but have a strong impact on productivity.

Conclusions

In this thesis I proposed two methods for the text categorization.

The first is an hybrid machine learning method that involves a filter and a genetic wrapper for the extraction of the words that best categorizes a specific target class. The aims is to balance the aspects of filter and wrapper approach. The use of a genetic algorithm is expensive, but permits to explore different solutions. A filter reduces costs by deriving a feature subspace of limited size.

Experiments on a popular benchmark, the Reuters collection, showed that the method is very competitive because it reached good performance in terms of F-measure, and BEP, overcoming problems caused by the high dimensionality of the text collection.

The above method was also applied to extract knowledge from biological text and find the most representative term for families of genes. Starting from an annotated corpus of gene summaries the method selects the annotations that best characterize the specific family.

Applied to the classification of news and, gene summaries, the proposed approach seems to be promising.

Finally, I explored the field of the Word Sense Disambiguation, using ontologies like knowledge sources for the text classification. I proposed a method that uses WordNet to disambiguate the words in a document, and WordNet Domains to categorize the documents.

Experiments showed how the method performs well in classifying a large corpus of documents from SemCor Collection.

The method enables the classification of documents, without train a classifier over a set of documents whose categorization depends on the subjective judgment of a human classifier. This approach considers only the knowledge that can be extracted from the text content using ontologies in that, overcoming problems related to the availability of a manually classified dataset.

Appendix A

WordNet

WordNet [MB+90][FC98] is a computational lexicon of English based on psycholinguistic principles, created and maintained at Princeton University. It encodes concepts in terms of sets of synonyms (called synsets). Its latest version, WordNet 3.0, contains about 155,000 words organized in over 117,000 synsets. For example, the concept of automobile is expressed with the following synset: {car, auto, automobile, machine, motorcar}. Where synset is the set of the meaning in WordNet.

We note that each word sense univocally identifies a single synset. For instance, given "car" the corresponding synset {car, auto, automobile, machine, motorcar} is univocally determined.

In Figure 1 we report an excerpt of the WordNet semantic network containing the car synset. For each synset, WordNet provides the following information:

- A gloss, that is, a textual definition of the synset possibly with a set of usage examples (e.g., the gloss of car is "a 4-wheeled motor vehicle; usually propelled by an internal combustion engine; 'he needs a car to get to work' ").
- Lexical and semantic relations, which connect pairs of word senses and synsets, respectively: while semantic relations apply to synsets in their entirety (i.e., to all members of a synset), lexical relations connect word senses included in the respective synsets.

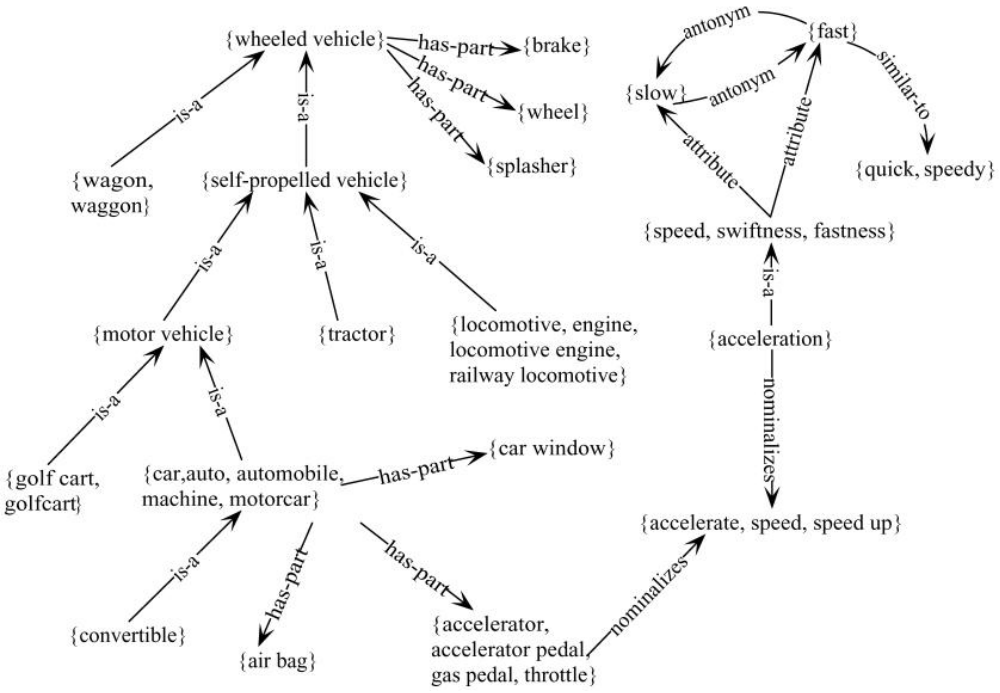

Figure 1. An excerpt of the WordNet semantic network.

Among the lexical relations, we have the following:

- Antonymy: X is an antonym of Y if it expresses the opposite concept (e.g., good is the antonym of bad).
- Pertainymy: X is an adjective which can be defined as "of or pertaining to" a noun (or, rarely, another adjective) Y (e.g., dental pertains to tooth).
- Nominalization: a noun X nominalizes a verb Y (e.g., service nominalizes the verb serve).

Among the semantic relations, we have the following:

- Hypernymy (also called kind-of or is-a): Y is a hypernym of X if every X is a (kind of) Y (motor vehicle is a hypernym of car). Hypernymy holds between pairs of nominal or verbal synsets.

- Hyponymy and troponymy: the inverse relations of hypernymy for nominal and verbal synsets, respectively.
- Meronymy (also called part-of): Y is a meronym of X if Y is a part of X (e.g., flesh is a meronym of fruit). Meronymy holds for nominal synsets only.
- Holonymy: Y is a holonym of X if X is a part of Y (the inverse of meronymy).
- Entailment: a verb Y is entailed by a verb X if by doing X you must be doing Y (e.g., snore entails sleep).
- Similarity: an adjective X is similar to an adjective Y (e.g., beautiful is similar to pretty).
- Attribute: a noun X is an attribute for which an adjective Y expresses a value (e.g., hot is a value of temperature).
- See also: this is a relation of relatedness between adjectives (e.g., beautiful is related to attractive through the see also relation).

Magnini and Cavaglià [MC00] developed a data set of domain labels for WordNet synsets. WordNet synsets have been semi-automatically annotated with one or more domain labels from a pre-defined set of about 200 tags from the Dewey Decimal Classification (e.g. FOOD, ARCHITECTURE, SPORT, etc.) plus a generic label (FACTOTUM) when no domain information is available. Labels are organized in a hierarchical structure (e.g., PSYCHOANALYSIS is a kind of PSYCHOLOGY domain).

Given its widespread diffusion within the research community, WordNet can be considered a de facto standard for English WSD. Following its success, wordnets for several languages have been developed and linked to the original Princeton WordNet.

SemCor

SemCor [ML+93] is a subset of the Brown Corpus [KF67] whose content words have been manually annotated with part-of-speech tags, lemmas, and word senses from the WordNet inventory. SemCor is composed of 352 texts: in 186 texts all the open-class words (nouns, verbs, adjectives, and adverbs) are annotated with these information, while in the remaining 166 texts only verbs are semantically annotated with word senses.

Overall, SemCor comprises a sample of around 234,000 semantically annotated words, thus constituting the largest sense-tagged corpus for training sense classifiers in supervised disambiguation settings. An excerpt of a text in the corpus is reported in Figure 2.

As of Sunday$_n^1$ night$_n^1$ there was$_v^4$ no word$_n^2$ of a resolution$_n^1$ being offered$_v^2$ there$_r^1$ to rescind$_v^1$ the action$_n^1$. Pelham pointed out$_v^1$ that Georgia$_n^1$ voters$_n^1$ last$_r^1$ November$_n^1$ rejected$_v^2$ a constitutional$_a^1$ amendment$_n^1$ to allow$_v^2$ legislators$_n^1$ to vote$_n^1$ on pay$_n^1$ raises$_n^1$ for future$_a^1$ Legislature$_n^1$ sessions$_n^2$.

Figure 2. An excerpt of the SemCor semantically annotated corpus.

For instance, word$_n$ is annotated in the first sentence with sense #2, defined in WordNet as “a brief statement” (compared, e.g., to sense #1 defined as “a unit of language that native speakers can identify”). The original SemCor was annotated according to WordNet 1.5. However, mappings exist to more recent versions (e.g., 2.0, 2.1, etc.).

Appendix B

The table reports for each document in SemCor (186 documents), the attributed domains by the proposed ontology based method, and the relative frequency.

Topics Doc1 1 - politics: 49 2 - law: 45 3 - administration: 39	Topics Doc2 1 - music: 24 2 - person: 22 3 - metrology: 21	Topics Doc3 1 - politics: 39 2 - geography: 21 3 - anthropology: 20
Topics Doc4 1 - religion: 69 2 - psychological_features: 25 3 - mathematics: 18	Topics Doc5 1 - person: 56 2 - psychological_features: 22 3 - mathematics: 21	Topics Doc6 1 - person: 31 2 - administration: 30 3 - commerce: 20
Topics Doc7 1 - medicine: 32 2 - psychological_features: 14 3 - buildings: 11	Topics Doc8 1 - publishing: 61 2 - literature: 50 3 - linguistics: 39	Topics Doc9 1 - art: 54 2 - linguistics: 41 3 - grammar: 27
Topics Doc10 1 - linguistics: 79 2 - art: 20	Topics Doc11 1 - linguistics: 87 2 - metrology: 24 3 - mathematics: 14	Topics Doc12 1 - politics: 83 2 - administration: 29
Topics Doc13 1 - music: 32 2 - racing: 15	Topics Doc14 1 - administration: 25 2 - pedagogy: 23 3 - person: 21	Topics Doc15 1 - economy: 125 2 - money: 32 3 - commerce: 31
Topics Doc16 1 - law: 62 2 - politics: 38 3 - geography: 17	Topics Doc17 1 - psychological_features: 29 2 - medicine: 18 3 - animals: 9	Topics Doc18 1 - medicine: 31 2 - anatomy: 17 3 - person: 11
Topics Doc19 1 - geography: 21 2 - literature: 16	Topics Doc20 1 - buildings: 41 2 - administration: 24 3 - military: 22	Topics Doc21 1 - geography: 35 2 - administration: 25 3 - buildings: 22
Topics Doc22 1 - economy: 32 2 - religion: 19 3 - politics: 14	Topics Doc23 1 - law: 40 2 - politics: 30	Topics Doc24 1 - person: 18 2 - literature: 17
Topics Doc25 1 - art: 21 2 - photography: 18 3 - mathematics: 9	Topics Doc26 1 - administration: 56 2 - geography: 32 3 - buildings: 31	Topics Doc27 1 - metrology: 80 2 - chemistry: 74 3 - geography: 47
Topics Doc28 1 - anatomy: 28 2 - person: 25 3 - buildings: 24	Topics Doc29 1 - military: 30 2 - anatomy: 24 3 - geography: 16	Topics Doc30 1 - military: 25 2 - person: 18

Topics Doc31 1 - religion: 35 2 - buildings: 24	Topics Doc32 1 - buildings: 25	Topics Doc33 1 - person: 37 2 - buildings: 31 3 - anatomy: 22
Topics Doc34 1 - person: 25 2 - psychological_features: 12 3 - anatomy: 9	Topics Doc35 1 - religion: 26 2 - psychological_features: 13 3 - person: 12	Topics Doc36 1 - literature: 30 2 - music: 25 3 - person: 18
Topics Doc37 1 - military: 42 2 - town_planning: 18 3 - anatomy: 17	Topics Doc38 1 - religion: 42 2 - anatomy: 25 3 - person: 18	Topics Doc39 1 - anatomy: 47 2 - buildings: 26 3 - chemistry: 16
Topics Doc40 1 - person: 29 2 - buildings: 24 3 - anatomy: 11	Topics Doc41 1 - buildings: 67 2 - furniture: 31	Topics Doc42 1 - religion: 30 2 - person: 25 3 - anatomy: 23
Topics Doc43 1 - buildings: 24 2 - person: 22 3 - psychological_features: 17	Topics Doc44 1 - geography: 43 2 - animals: 25 3 - anatomy: 19	Topics Doc45 1 - geography: 30 2 - religion: 18 3 - buildings: 14
Topics Doc46 1 - religion: 30 2 - theology: 10	Topics Doc47 1 - person: 25 2 - anatomy: 18 3 - buildings: 11	Topics Doc48 1 - person: 22 2 - religion: 19 3 - buildings: 17
Topics Doc49 1 - person: 25 2 - buildings: 18 3 - anatomy: 17	Topics Doc50 1 - military: 29 2 - anatomy: 28	Topics Doc51 1 - buildings: 30 2 - person: 19
Topics Doc52 1 - anatomy: 25 2 - person: 22 3 - buildings: 13	Topics Doc53 1 - person: 62 2 - anatomy: 50 3 - buildings: 13	Topics Doc54 1 - buildings: 30 2 - anatomy: 23 3 - psychological_features: 14
Topics Doc55 1 - buildings: 28 2 - person: 22 3 - gastronomy: 10	Topics Doc56 1 - buildings: 42 2 - animals: 42 3 - medicine: 32	Topics Doc57 1 - religion: 104 2 - person: 22
Topics Doc58 1 - buildings: 32 2 - fashion: 27 3 - anatomy: 23	Topics Doc59 1 - person: 27 2 - buildings: 14	Topics Doc60 1 - person: 16 2 - anatomy: 11 3 - money: 10
Topics Doc61 1 - buildings: 42 2 - chemistry: 27 3 - food: 11	Topics Doc62 1 - buildings: 16 2 - psychological_features: 13 3 - person: 11	Topics Doc63 1 - buildings: 24 2 - anatomy: 17 3 - person: 15
Topics Doc64 1 - person: 15 2 - buildings: 12 3 - literature: 9	Topics Doc65 1 - buildings: 32 2 - person: 20	Topics Doc66 1 - buildings: 22 2 - person: 19

Topics Doc67 1 - buildings: 26 2 - law: 18	Topics Doc68 1 - religion: 69 2 - biology: 10	Topics Doc69 1 - buildings: 72 2 - military: 14 3 - town_planning: 13
Topics Doc70 1 - anatomy: 28 2 - military: 24	Topics Doc71 1 - buildings: 18 2 - person: 9 3 - anatomy: 9	Topics Doc72 1 - person: 14 2 - religion: 11
Topics Doc73 1 - geography: 30 2 - grammar: 22 3 - linguistics: 18	Topics Doc74 1 - animals: 18 2 - anatomy: 17 3 - psychological_features: 17	Topics Doc75 1 - anatomy: 20 2 - animals: 10 3 - geography: 9
Topics Doc76 1 - buildings: 39 2 - person: 24 3 - anatomy: 15	Topics Doc77 1 - law: 26 2 - person: 24 3 - geography: 19	Topics Doc78 1 - buildings: 47 2 - anatomy: 41 3 - animals: 28
Topics Doc79 1 - anatomy: 68 2 - sport: 29 3 - health: 29	Topics Doc80 1 - buildings: 33 2 - anatomy: 19 3 - fashion: 13	Topics Doc81 1 - military: 26 2 - transport: 23
Topics Doc82 1 - transport: 41 2 - anatomy: 38 3 - person: 17	Topics Doc83 1 - anatomy: 15 2 - buildings: 15 3 - fashion: 14	Topics Doc84 1 - anatomy: 34 2 - geography: 30 3 - chemistry: 22
Topics Doc85 1 - anatomy: 37 2 - buildings: 33 3 - person: 28	Topics Doc86 1 - military: 19 2 - psychological_features: 16	Topics Doc87 1 - buildings: 26 2 - person: 15 3 - geography: 11
Topics Doc88 1 - art: 22 2 - painting: 18 3 - person: 18	Topics Doc89 1 - person: 18 2 - religion: 16	Topics Doc90 1 - chemistry: 56 2 - plants: 48 3 - food: 28
Topics Doc91 1 - baseball: 67 2 - play: 13	Topics Doc92 1 - buildings: 44 2 - furniture: 11 3 - animals: 8	Topics Doc93 1 - person: 31 2 - linguistics: 22
Topics Doc94 1 - literature: 19 2 - animals: 18 3 - person: 17	Topics Doc95 1 - person: 40 2 - anatomy: 25 3 - chemistry: 14	Topics Doc96 1 - person: 17 2 - geography: 16
Topics Doc97 1 - person: 22 2 - buildings: 18	Topics Doc98 1 - music: 69 2 - racing: 12	Topics Doc99 1 - law: 41 2 - geography: 38 3 - administration: 33
Topics Doc100 1 - geography: 61 2 - free_time: 26	Topics Doc101 1 - music: 63 2 - person: 22 3 - geography: 20	Topics Doc102 1 - color: 25 2 - painting: 23 3 - art: 21

Topics Doc103 1 - anatomy: 52 2 - person: 30 3 - sport: 29	Topics Doc104 1 - anatomy: 38 2 - physics: 36 3 - electronics: 29	Topics Doc105 1 - geography: 27 2 - chemistry: 22 3 - electricity: 18
Topics Doc106 1 - metrology: 78 2 - animals: 71 3 - medicine: 54	Topics Doc107 1 - commerce: 113 2 - economy: 37 3 - enterprise: 23	Topics Doc108 1 - buildings: 25 2 - pedagogy: 25 3 - architecture: 24
Topics Doc109 1 - person: 55 2 - economy: 40	Topics Doc110 1 - baseball: 58 2 - sport: 28	Topics Doc111 1 - tourism: 39 2 - military: 22 3 - person: 20
Topics Doc112 1 - psychology: 25 2 - physiology: 17 3 - psychological_features: 15	Topics Doc113 1 - law: 30 2 - psychological_features: 20	Topics Doc114 1 - medicine: 93 2 - law: 37 3 - person: 27
Topics Doc115 1 - agriculture: 42 2 - economy: 35 3 - buildings: 20	Topics Doc116 1 - law: 47 2 - person: 25 3 - geography: 25	Topics Doc117 1 - religion: 60 2 - law: 32 3 - medicine: 31
Topics Doc118 1 - geography: 63 2 - economy: 22	Topics Doc119 1 - geography: 79 2 - metrology: 28 3 - person: 25	Topics Doc120 1 - military: 42 2 - geography: 42 3 - person: 39
Topics Doc121 1 - play: 42 2 - sport: 29 3 - metrology: 27	Topics Doc122 1 - literature: 31 2 - folklore: 26 3 - person: 25	Topics Doc123 1 - law: 30 2 - administration: 25 3 - politics: 15
Topics Doc124 1 - geography: 62 2 - geology: 48 3 - metrology: 27	Topics Doc125 1 - military: 103 2 - geography: 22 3 - history: 12	Topics Doc126 1 - politics: 40 2 - geography: 33 3 - person: 12
Topics Doc127 1 - geography: 43 2 - buildings: 29 3 - military: 18	Topics Doc128 1 - person: 39 2 - school: 29 3 - religion: 15	Topics Doc129 1 - school: 48 2 - person: 25 3 - pedagogy: 23
Topics Doc130 1 - racing: 43 2 - photography: 20	Topics Doc131 1 - religion: 77 2 - person: 28 3 - sociology: 26	Topics Doc132 1 - baseball: 71 2 - play: 19 3 - sport: 14
Topics Doc133 1 - geography: 53 2 - politics: 45 3 - person: 17	Topics Doc134 1 - psychological_features: 37 2 - person: 15 3 - astronomy: 9	Topics Doc135 1 - psychological_features: 18 2 - person: 18
Topics Doc136 1 - plants: 38 2 - biology: 36 3 - buildings: 34	Topics Doc137 1 - psychological_features: 28 2 - person: 15	Topics Doc138 1 - psychology: 20 2 - psychological_features: 20 3 - art: 17

Topics Doc139 1 - geography: 50 2 - person: 35 3 - politics: 22	Topics Doc140 1 - literature: 27 2 - person: 26 3 - geography: 26	Topics Doc141 1 - law: 48 2 - literature: 41 3 - administration: 30
Topics Doc142 1 - person: 28 2 - free_time: 12 3 - publishing: 11	Topics Doc143 1 - sport: 32 2 - golf: 14	Topics Doc144 1 - politics: 56 2 - religion: 16
Topics Doc145 1 - enterprise: 31 2 - economy: 20 3 - commerce: 16	Topics Doc146 1 - person: 21 2 - literature: 16 3 - religion: 15	Topics Doc147 1 - literature: 38 2 - geography: 19 3 - person: 13
Topics Doc148 1 - person: 44 2 - religion: 11 3 - psychological_features: 11	Topics Doc149 1 - music: 47 2 - person: 24 3 - literature: 18	Topics Doc150 1 - literature: 38 2 - psychological_features: 37 3 - religion: 11
Topics Doc151 1 - literature: 32 2 - history: 18 3 - art: 12	Topics Doc152 1 - geography: 55 2 - enterprise: 40 3 - economy: 36	Topics Doc153 1 - administration: 43 2 - chemistry: 37 3 - economy: 33
Topics Doc154 1 - sport: 29 2 - baseball: 24 3 - person: 15	Topics Doc155 1 - physics: 66 2 - chemistry: 65 3 - metrology: 48	Topics Doc156 1 - administration: 75 2 - law: 53 3 - money: 28
Topics Doc157 1 - geography: 39 2 - military: 38 3 - person: 20	Topics Doc158 1 - politics: 42 2 - geography: 42 3 - money: 38	Topics Doc159 1 - buildings: 110 2 - physics: 16 3 - metrology: 14
Topics Doc160 1 - law: 55 2 - exchange: 31	Topics Doc161 1 - law: 54 2 - administration: 53 3 - person: 41	Topics Doc162 1 - politics: 39 2 - administration: 31 3 - person: 19
Topics Doc163 1 - military: 53 2 - economy: 29 3 - transport: 20	Topics Doc164 1 - tax: 51 2 - economy: 49 3 - law: 47	Topics Doc165 1 - anatomy: 34 2 - person: 28 3 - medicine: 16
Topics Doc166 1 - physics: 105 2 - metrology: 63 3 - astronomy: 22	Topics Doc167 1 - physics: 87 2 - electronics: 75 3 - electricity: 43	Topics Doc168 1 - physics: 107 2 - chemistry: 38 3 - metrology: 24
Topics Doc169 1 - chemistry: 77 2 - physics: 45 3 - metrology: 20	Topics Doc170 1 - chemistry: 151 2 - physics: 34 3 - medicine: 20	Topics Doc171 1 - chemistry: 159 2 - metrology: 31 3 - physics: 30
Topics Doc172 1 - metrology: 42 2 - physics: 41 3 - person: 29	Topics Doc173 1 - medicine: 25 2 - chemistry: 19 3 - military: 19	Topics Doc174 1 - chemistry: 160 2 - metrology: 33 3 - anatomy: 30

Topics Doc175 1 - biology: 81 2 - animals: 36 3 - plants: 28	Topics Doc176 1 - politics: 22 2 - military: 19 3 - person: 19	Topics Doc177 1 - animals: 95 2 - metrology: 49 3 - physics: 23
Topics Doc178 1 - anatomy: 154 2 - animals: 39 3 - publishing: 13	Topics Doc179 1 - publishing: 29 2 - anatomy: 29 3 - person: 28	Topics Doc180 1 - chemistry: 139 2 - anatomy: 91 3 - metrology: 17
Topics Doc181 1 - anatomy: 141 2 - medicine: 72 3 - metrology: 63	Topics Doc182 1 - chemistry: 94 2 - biology: 54 3 - anatomy: 31	Topics Doc183 1 - anatomy: 35 2 - physiology: 32 3 - psychological_features: 25
Topics Doc184 1 - mathematics: 121	Topics Doc185 1 - law: 33 2 - mathematics: 26	Topics Doc186 1 - geometry: 78 2 - mathematics: 55 3 - geography: 23

References

[AD+97] Amati, G., D'Aloisi, D., Giannini, V. & Ubaldini, F.: A framework for filtering news and managing distributed data. Journal of Universal Computer Science, 3(8), 1007–1021, 1997.

[AL07] Auer, S., Lehmann, J.: What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content. European Semantic Web Conference (ESWC'07), 503-517, 2007.

[AR96] Agirre, E. and Rigau, G.: Word sense disambiguation using conceptual density. In Proceedings of the 16th International Conference on Computational Linguistics (COLING, Copenhagen, Denmark), 16–22, 1996.

[AV+08] Pietramala, A., Policicchio, V., Rullo, P., Sidhu, I.: A genetic algorithm for text classification rule induction. Machine Learning and Knowledge Discovery in Databases, 188-203, 2008.

[BB63] Borko, H. and Bernick, M.: Automatic document classification. J. Assoc. Comput. Mach. 10, 2, 151–161, 1963.

[BD+07] Basile, P., De Gemmis, M., Gentile, A.L., Lops, P., Semeraro, G.: UNIBA: JIGSAW algorithm for Word Sense Disambiguation. Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), 398-401, 2007.

[BEC13] Ilgen, B., Adali, E., Tantug, A.C.: A Comparative Study to Determine the Effective Window Size of Turkish Word Sense Disambiguation Systems. ISCIS 2013, 169-176, 2013.

[BF+04] Bentivogli, L., Forner, P., Magnini, B., Pianta, E.: Revising WordNet Domains Hierarchy: Semantics, Coverage, and Balancing. Proceedings of COLING 2004 Workshop on Multilingual Linguistic Resources, Geneva, Switzerland, 101-108, 2004.

[BF+10] Bouckaert, R.R., Frank, E., Hall, M.A., et al.: WEKA - Experiences with a Java Open-Source Project. Journal of Machine Learning Research, 11, 2533-2541, 2010.

[BH04] Bloehdorn, S., Hotho, A.: Text Classification by Boosting Weak Learners based on Terms and Concepts. 4th IEEE International Conference on Data Mining, 2004.

[BH06] Budanitsky, A. and Hirst, G.: Evaluating WordNet-based measures of semantic distance. Computat. Ling. 32, 1, 13–47, 2006.

[BL+09] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia – A Crystallization Point for the Web of Data. Journal of Web Semantics: Science, Services and Agents on the World Wide Web, Issue 7, 154–165, 2009.

[BM98] Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proc. of the Workshop on Computational Learning Theory, 1998.

[BM+06] Buitelaar, P., Magnini, B., Strapparava, C., and Vossen, P.: Domain-specific WSD. In Word Sense Disambiguation: Algorithms and Applications, E. Agirre and P. Edmonds, Eds. Springer, New York, NY, 275–298, 2006.

[BP02] Banerjee, S., Pedersen, T.: An adapted lesk algorithm for word sense disambiguation using WordNet. In Proceedings of the Third International

Conference on Intelligent Text Processing and Computational Linguistics, 136-145, 2002.

[BS90] Bunke, H. and Sanfeliu, A.,: Syntactic and Structural Pattern Recognition: Theory and Applications. Vol. 7. World Scientific Series in Computer Science, World Scientific, Singapore, 1990.

[BWL10] Bai, R., Wang, X., Liao, J.: Extract semantic information from WordNet to improve text classification performance. In Proceedings of the international conference on Advances in computer science and information technology, LNCS 6059, 409–420, 2010.

[CA+00] Chandrinos, K.V., Androutsopoulos, I., Paliouras, G. and Spyropoulos, C.D.: Automatic Web rating: Filtering obscene content on the Web. Proceedings of ECDL-00, 4th European Conference on Research and Advanced Technology for Digital Libraries, eds. J.L. Borbinha & T. Baker, Springer Verlag, Heidelberg, DE: Lisbon, PT, 403–406, 2000. Published in the "Lecture Notes in Computer Science" series, number 1923.

[CDP10] Cannas, L.M., Dessì, N., Pes B.: A filter-based evolutionary approach for selecting features in high-dimensional micro-array data. IIP 2010, IFIP AICT 340, 297-307, 2010.

[CDP210] Cannas, L.M., Dessì, N., Pes, B.: Tuning Evolutionary Algorithms in High Dimensional Classification Problems (Extended Abstract). SEBD 2010, 142-149, 2010.

[CMS01] Caropreso, M.F., Matwin, S., and Sebastiani, F.: A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In Text Databases and Document Management: Theory

and Practice, A. G. Chin, ed. Idea Group Publishing, Hershey, PA, 78–102, 2001.

[CS01] Crammer, K. and Singer, Y.: On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research, 2, 265–292, 2001.

[CT99] Cavnar, W.B. and Trenkle, J.M.: N-gram-based text categorization. In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (Las Vegas, NV, 1994), 161–175, 1994.

[DM+97] De Buenaga Rodriguez, M., Gomez-Hidalgo, J., Diaz-Agudo, B.: Using WordNet to complement training information in text categorization. In Proceedings of the 2nd International Conference on Recent Advances in Natural Language Processing (RANLP'97), 150–157, 1997.

[DP+98] Dumais, S.T., Platt, J., Heckerman, D., and Sahami, M.: Inductive learning algorithms and representations for text categorization. In Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management (Bethesda, MD, 1998), 148–155, 1998.

[EM00] Escudero, G., Marquez, L. and Rigau, G.: Boosting applied to word sense disambiguation. Proceedings of ECML-00, 11th European Conference on Machine Learning, eds. R.L.D. Mantaras and E. Plaza, Springer Verlag, Heidelberg, DE: Barcelona, ES, 129–141, 2000. Published in the "Lecture Notes in Computer Science" series, number 1810.

[FB75] Field, B.: Towards automatic indexing: automatic assignment of controlled-language indexing and classification from free indexing. J. Document. 31, 4, 246–265, 1975.

[FB91] Fihr, N. and Buckley, C.: A probabilistic learning approach for document indexing. ACM Trans. Inform. Syst. 9, 3, 223–248, 1991.

[FC98] Fellbaum, C., ed.: WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press, 1998.

[FG03] Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research, 3, 1289-1305, 2003.

[FK82] Fu, K.: Syntactic Pattern Recognition and Applications. Prentice-Hall, Engelwood Cliffs, NJ, 1982.

[FRS99] Forsyth, R.S.: New directions in text categorization. In Causal Models and Intelligent Data Management, A. Gammerman, ed. Springer, Heidelberg, Germany, 151–185, 1999.

[FT+09] Ferrández, S., Toral, A., Ferrández, O., Ferrández, A., Muñoz, R.: Exploiting Wikipedia and EuroWordNet to solve cross–lingual question answering. Information Sciences, 179(20), 3473–3488, 2009.

[FWB92] Frakes, W.B.: Stemming algorithms. Information Retrieval: Data Structures and Algorithms, eds. W.B. Frakes & R. Baeza-Yates, Prentice Hall: Engle-wood Cliffs, US, 131-160, 1992.

[GCY92] Gale, W.A., Church, K., and Yarowsky, D.: A method for disambiguating word senses in a corpus. Comput. Human. 26, 415–439, 1992.

[GD+13] Gray, K.A., Daugherty, L.C., Gordon, S.M., Seal, R.L., Wright, M.W., Bruford, E.A.: Genenames.org: the HGNC resources in 2013. Nucleic Acids Res. 41(Database issue):D545-52, 2013.

[GDE89] Goldberg, D.E.: Genetic algorithms in search, optimization and machine learning. Addison-Wesley, 1989.

[GDJ13] Gan, M., Dou, X., Jiang, R.: From Ontology to Semantic Similarity: Calculation of Ontology-Based Semantic Similarity. The Scientific World Journal, Volume 2013, Article ID 793091, 11 pages, 2013.

[GH71] Gray, W.A. and Harley, A.J.: Computer-assisted indexing. Inform. Storage Retrieval 7, 4, 167–174, 1971.

[GM05] Gabrilovich, E., Markovitch, S.: Feature generation for text categorization using world knowledge. In International joint conference on artificial intelligence, 2005.

[GM06] Gabrilovich, E., Markovitch, S.: Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In National conference on artificial intelligence (AAAI), 2006.

[GM+12] Guzzi, P.H., Mina, M., Guerra, C., Cannataro, M.: Semantic similarity analysis of protein data: assessment with biological features and issues. Briefings in Bioinformatics 13(5), 569-585, 2012.

[GMS04] Gliozzo, A., Magnini, B., and Strapparava, C.: Unsupervised domain relevance estimation for word sense disambiguation. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP, Barcelona, Spain), 380–387, 2004.

[GSD05] Gliozzo, A.M., Strapparava, C., Dagan, I.: Investigating Unsupervised Learning for Text Categorization Bootstrapping. In: Proc. of EMNLP, 2005.

[GSS00] Galavotti, L., Sebastiani , F., and Simi, M.: Experiments on the use of feature selection and negative evidence in automated text categorization. In Proceedings of ECDL-00, 4th European Conference on Research and

Advanced Technology for Digital Libraries (Lisbon, Portugal, 2000), 59–68, 2000.

[GTR93] Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. In Proceedings of the International Workshop on Formal Ontology (Padova, Italy), 1993.

[H+09] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations, vol. 11, no. 1, 2009.

[HB+94] Hersh, W., Buckley, C., Leone, T., and Hickman, D.: OHSUMED: an interactive retrieval evaluation and new large text collection for research. In Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval (Dublin, Ireland, 1994), 192–201, 1994.

[HH73] Heaps, H.: A theory of relevance for automatic document classification. Inform. Control 22, 3, 268–278, 1973.

[HSS03] Hotho, A., Staab, S., Stumme, G.: Wordnet improves text document clustering. In Proc. of the semantic web workshop at SIGIR, 541–544, 2003.

[JC97] Jiang, J., Conrath, D.: Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings on International Conference on Research in Computational Linguistics, 19–33, 1997.

[JKA08] Janik, M., Kochut, K.: Training-less Ontology-based Text Categorization, In Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR), 2008.

[JKB08] Janik, M., Kochut, K.: Wikipedia in action: Ontological knowledge in text categorization, IEEE International Conference on Semantic Computing, pp.268-75, 2008.

[JS+09] Jonquet, C., Shah, N.H., Musen, M.A.: The open biomedical annotator. Summit on Translat Bioinforma. 2009:56-60, 2009.

[JT98] Joachims, T.: Text categorization with support vector machines: learning with many relevant features. Proceedings of ECML-98, 10th European Conference on Machine Learning, eds. C. Nedellec and C. Rouveirol, Springer Verlag, Heidelberg, DE: Chemnitz, DE, 137–142, 1998. Published in the "Lecture Notes in Computer Science" series, number 1398.

[JT99] Joachims, T.: Transductive inference for text classification using support vector machines. Proceedings of ICML-99, 16th International Conference on Machine Learning, eds. I. Bratko & S. Dzeroski, Morgan Kaufmann Publishers, San Francisco, US: Bled, SL, 200–209, 1999.

[KF67] Kucera, H. and Francis, W.N.: Computational Analysis of Present-Day American English. Brown University Press, Providence, RI,1967.

[KNS97] Kessler, B., Nunberg, G., and Schutze, H.: Automatic detection of text genre. In Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics (Madrid, Spain, 1997), 32–38, 1997.

[KP+03] Kehagias, A., Petridis, V., Kaburlasos, V., Fragkou, P.: A comparison of word- and sense-based text classification using several classification algorithms. Journal of Intelligent Information Systems 21(3), 227–247, 2003.

[KS09] Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of Wikipedia entities in web text. In Proc. ACM KDD, 457–466, 2009.

[LC96] Larkey, L.S. and Croft, W.B.: Combining classifiers in text categorization. In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Zurich, Switzerland, 1996), 289–297, 1996.

[LCX11] Luo, Q., Chen, E., Xiong, H.: A semantic term weighting scheme for text categorization. Expert Systems with Applications, 38(10), 12708–12716, 2011.

[LD97] Lewis, D.D.: Reuters-21578 text categorization test collection, distribution 1.0. http://www.research.att.com/~ lewis/reuters21578.html , 1997.

[LD98] Lin, D.: An information-theoretic definition of similarity. In Proceedings of the 15th International Conference on Machine Learning (ICML, Madison, WI). 296–304, 1998.

[LK95] Lang, K.: NEWS WEEDER: learning to filter netnews. In Proceedings of ICML-95, 12th International Conference on Machine Learning (Lake Tahoe, CA, 1995), 331–339, 1995.

[LL98] Larkey, L.S.: Automatic essay grading using text categorization techniques. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, Australia, 1998), 90–95, 1998.

[LM86] Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In Proceedings of the 5th SIGDOC (New York, NY), 24–26, 1986.

[LZL09] Li, J.Q., Zhao, Y., Liu, B.: Fully automatic text categorization by exploiting WordNet. In Proceeding of Asia information retrieval societies conference, LNCS 5839, 1–12, 2009.

[LZL12] Li, J.Q., Zhao, Y., Liu, B.: Exploiting semantic resources for large scale text categorization, Intell Inf Syst 39: 763-788, 2012.

[MC00] Magnini, B., Cavaglià, G.: Integrating Subject Field Codes into WordNet. Proceedings of LREC-2000, Second International Conference on Language Resources and Evaluation, Athens, Greece, 1413-1418, 2000.

[MC07] Mihalcea, R., Csomai, A.: Wikify!: linking documents to encyclopedic knowledge. In Proc. ACM CIKM, 233–242, 2007.

[MGA95] Miller, G.A.: WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11, 39-41, 1995.

[MK00] Myers, K., Kearns, M., Singh, S., and Walker, M.A.: A boosting approach to topic spotting on subdialogues. In Proceedings of ICML-00, 17th International Conference on Machine Learning (Stanford, CA, 2000), 655– 662, 2000.

[ML+93] Miller, G.A., Leacock, C., Tengi, R., and Bunker, R.T.: A semantic concordance. In Proceedings of the ARPA Workshop on Human Language Technology, 303–308, 1993.

[MM61] Maron, M.: Automatic indexing: an experimental inquiry. J. Assoc. Comput. Mach. 8, 3, 404–417, 1961.

[MN98] Mccallum, A., Nigam, K.: A Comparison of Event Models for Naive Bayes Text Classification. AAAI-98 Workshop on ‘Learning for Text Categorization’, 1998.

[MRG96] Moulinier, I., Raškinis, G., and Ganascia, J.G.: Text categorization: a symbolic approach. In Proceedings of SDAIR-96, 5th Annual

Symposium on Document Analysis and Information Retrieval (Las Vegas, NV, 1996), 87–99, 1996.

[NBC14] Naïve Bayes Classifier. https://www.princeton.edu/~achaney/tmve/wiki100k/docs/Naive_Bayes_classifier.html, 2014.

[NGL97] Ng, H.T., Goh, W.B., and Low, K.L.: Feature selection, perceptron learning, and a usability case study for text categorization. In Proceedings of SIGIR-97, 20th ACM International Conference on Research and Development in Information Retrieval (Philadelphia, PA, 1997), 67–73, 1997.

[NM+00] Nigam, K., McCallum, A., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. Machine Learning, 103–134, 2000.

[NR09] Navigli, R.: Word sense disambiguation: A survey. ACM Comput. Surv. 41, 2., 2009.

[JC97] Jiang, J.J. and Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings of the 10th International Conference on Research in Computational Linguistics (Taiwan, ROC), 1997.

[JKP94] John, G.H., Kohavi, R., and Pfleger, K.: Irrelevant features and the subset selection problem. In Proceedings of ICML-94, 11th International Conference on Machine Learning (New Brunswick, NJ, 1994), 121–129, 1994.

[JT98] Joachims, T.: Text categorization with support vector machines: learning with many relevant features. Proc. of ECML-98, 10th European Conference on Machine Learning(Chemnitz, Germany), 137-142, 1998.

[KS00] Ko, Y., Seo, J.: Automatic text categorization by unsupervised learning. In: Proc. Of COLING, 2000.

[KY00] Kilgarriff, A. and Yallop, C.: What's in a thesaurus? In Proceedings of the 2nd Conference on Language Resources and Evaluation (LREC, Athens, Greece). 1371–1379, 2000.

[LA92] Lewis, D.D.: An evaluation of phrasal and clustered representations on a text categorization task. In Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval (Copenhagen,Denmark, 1992), 37–50, 1992.

[LB92] Lewis, D.D.: Feature selection and feature extraction for text categorization. In: Proceedings of the workshop on Speech and Natural Language. Association for Computational Linguistics, 212-217, 1992.

[LC96] Larkey, L.S. and Croft, W.B.: Combining classifiers in text categorization. In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Zurich, Switzerland, 1996), 289–297, 1996.

[LC98] Leacock, C. and Chodorow, M.: Combining local context and WordNet similarity for word sense identification. In WordNet: An electronic Lexical Database, C. Fellbaum, Ed. MIT Press, Cambridge, MA, 265–283, 1998.

[LDD97] Lewis, D.D.: Reuters-21578 text categorization test collection. Distribution 1.0, 1997.

[LJ98] Li, Y.H. and Jain, A.K.: Classification of text documents. Comput. J. 41, 8, 537–546, 1998.

[LL+04] Liu, B., Li, X., Lee, W.S., Yu, P.S.: Text Classification by Labeling Words. In: Proc. 19th Nat'l Conf. Artificial Intelligence, 2004.

[LLH97] Lam, W., Low, K.F., and Ho, C.Y.: Using a Bayesian network induction approach for text categorization. In Proceedings of IJCAI-97, 15th International Joint Conference on Artificial Intelligence (Nagoya, Japan, 1997), 745–750, 1997.

[LLS98] Larkey, L.S.: Automatic essay grading using text categorization techniques. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, Australia, 1998), 90–95, 1998.

[LLS99] Larkey, L.S.: A patent search and classification system. Proceedings of DL-99, 4th ACM Conference on Digital Libraries, eds. E.A. Fox & N. Rowe, ACM Press, New York, US: Berkeley, US, pp. 179–187, 1999.

[LR94] Lewis, D.D. and Ringuette, M. A comparison of two learning algorithms for text categorization. In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (Las Vegas, NV, 1994), 81–93, 1994.

[LW10] Liu, J., Wang, G.: A hybrid feature selection method for data sets of thousands of variables. In: Advanced Computer Control (ICACC), 2010 2nd International Conference on. IEEE, 288-291, 2010.

[MB+90] Miller, G.A., Beckwith, R., Fellbaum, C.D., Gross, D., and Miller, K.: WordNet: An online lexical database. Int. J. Lexicograph. 3, 4, 235–244, 1990.

[MC00] Magnini, B., Cavaglià, G.: Integrating Subject Field Codes into WordNet. Proceedings of LREC-2000, Second International Conference on Language Resources and Evaluation, Athens, Greece, 1413-1418, 2000.

[MG96] Moulinier, I. and Ganascia, J.G.: Applying an existing machine learning algorithm to text categorization. In Connectionist, Statistical, and

Symbolic Approaches to Learning for Natural Language Processing, S. Wermter, E. Riloff, and G. Schaler, eds. Springer Verlag, Heidelberg, Germany, 343–354, 1996.

[MGA95] Miller, G.A.: WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41, 1995.

[MH06] Mansuy, T.N., Hilderman, R.J.: A Characterization of Wordnet Features in Boolean Models For Text Classification. In: AusDM 2006, 103–109, 2006.

[MM00] Moldovan, D. I., Mihalcea, R.: Using WordNet and lexical operators to improve internet searches. IEEE Internet Computing, 4(1), 34–43, 2000.

[MN98] Mccallum, A., Nigam, K.: A Comparison of Event Models for Naive Bayes Text Classification. AAAI-98 Workshop on 'Learning for Text Categorization', 1998.

[MTM96] Mitchell , T. M.: Machine Learning. McGraw Hill, New York, NY, 1996.

[NSS03] Nardiello, P., Sebastiani, F. & Sperduti, A.: Discretizing continuous attributes in AdaBoost for text categorization. Proceedings of ECIR-03, 25th European Conference on Information Retrieval, ed. F. Sebastiani, Springer Verlag: Pisa, IT, 320–334, 2003.

[NT97] Ng, T.H.: Getting serious about word sense disambiguation. In Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How? (Washington D.C.), 1–7, 1997.

[OO06] Olsson, J., Oard, D.W.: Combining feature selectors for text classification. In: Proceedings of the 15th ACM international conference on Information and knowledge management. ACM, 798-799, 2006.

[OS00] Oliver, S.: Guilt-by-association goes global. Nature 403, 601–603, 2000.

[PBP05] Pedersen, T., Banerjee, S., and Patwardhan, S.: Maximizing semantic relatedness to perform word sense disambiguation. Res. rep. UMSI 2005/25. University of Minnesota Supercomputing Institute, Minneapolis, MN, 2005.

[PC05] Peng, X., Choi, B.: Document classifications based on word semantic hierarchies. In: Proc. of the International Conf. on Artificial Intelligence and Application (AIA 2005), 362–367, 2005.

[PP+08] Pietramala, A., Policicchio, V.L., Rullo, P., Sidhu, I.: A Genetic Algorithm for Text Classification Rule Induction. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2008). LNCS 5212, Springer 2008, Antwerp (Belgium, 2008), 188-203, 2008.

[PPM04] Pedersen, T., Patwardhan, S., and Michelizzi, J.: WordNet::Similarity-measuring the relatedness of concepts. In Proceedings of the 19th National Conference on Artificial Intelligence (AAAI, San Jose, CA) 144–152, 2004.

[R+89] Rada, R., Mili, H., Bicknell, E., and Blettner, M.: Development and application of a metric on semantic nets. IEEE Trans. Syst. Man Cybernet. 19, 1, 17–30, 1989.

[RM+10] Richards, A.J., Muller, B., Shotwell, M., Cowart, L.A., Rohrer, B., Lu, X.: Assessing the functional coherence of gene sets with metrics based on the Gene Ontology graph. Bioinformatics 26(12): i79-i87, 2010.

[RP95] Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI, Montreal, P.Q., Canada), 448–453, 1995.

[RP+09] Rullo, P., Policicchio, V.L., Cumbo, C., Iiritano, S.: Olex: Effective Rule Learning for Text Categorization. IEEE Trans. Knowl. Data Eng. 21(8): 1118-1132, 2009.

[RS99] Ruiz, M.E. and Srinivasan, P.,: Hierarchical neural networks for text categorization. In Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval (Berkeley, CA, 1999), 281–282, 1999.

[S1] http://www.geneontology.org/GO.cite.shtml

[S2] http://www.ncbi.nlm.nih.gov/gene

[S3] http://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/

[S4] http://portal.ncibi.org/gateway/

[S5] http://ontologydesignpatterns.org/

[S6] http://bioportal.bioontology.org/

[SM93] Sussna, M.: Word sense disambiguation for free-text indexing using a massive semantic network. In Proceedings of the 2nd International Conference on Information and Knowledge Base Management (Washington D.C.), 67–74, 1993.

[SB88] Salton, G. and Buckley, C.: Term-weighting approaches in automatic text retrieval. Inform. Process. Man. 24, 5, 513–523, 1988. Also reprinted in Sparck Jones and Willett [1997], 323–328.

[SF02] Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys, 34(1), 1–47, 2002.

[SF05] Sebastiani, F., Text categorization. In Alessandro Zanasi (ed.), Text Mining and its Applications, WIT Press, Southampton, UK, 109-129, 2005.

[SH00] Sable, C.L. and Hatzivassiloglou, V.: Text-based approaches for non-topical image categorization. Internat. J. Dig. Libr. 3, 3, 261–275, 2000.

[SHP95] Schutze, H., Hull, D.A., and Pedersen, J.O.: A comparison of classifiers and document representations for the routing problem. In Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval (Seattle, WA, 1995), 229–237, 1995.

[SM83] Salton, G. and McGill, M.J.: An Introduction to Modern Information Retrieval. McGrawHill, 1983.

[SS00] Schapire, R.E. and Singer, Y.: BoosTexter: a boosting-based system for text categorization. Mach. Learn. 39, 2/3, 135–168, 2000.

[SSS98] Schapire, R.E., Singer, Y. and Singhal, A.: Boosting and Rocchio applied to text filtering. Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval, eds. W.B. Croft, A. Moffat, C.J. van Rijsbergen, R. Wilkinson & J. Zobel, ACM Press, New York, US: Melbourne, AU, 215–223, 1998.

[SSV00] Sebastiani, F., Sperduti, A., and Valdambrini, N.: An improved boosting algorithm and its application to automated text categorization. In Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management (McLean, VA, 2000), 78–85, 2000.

[TF07] Tan, F.: Improving Feature Selection Techniques for Machine Learning. Computer Science Dissertations, Paper 27, 2007.

[TH99] Taira, H. and Haruno, M.: Feature selection in SVM text categorization. In Proceedings of AAAI-99, 16th Conference of the American

Association for Artificial Intelligence (Orlando, FL, 1999), 480–486, 1999.

[U11] Uguz, H.: A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. Knowledge-Based Systems, 24.7: 1024-1032, 2011.

[VVN95] Vapnik, V.N.: The nature of statistical learning theory. Springer Verlag: Heidelberg, DE, 1995.

[WA+99] Weiss, S.M., Apté, C., Damerau, F.J., Johnson, D.E., Oles, F.J., Goetz, T. and Hampp, T.: Maximizing text-mining performance. IEEE Intelligent Systems, 14(4), 63–69, 1999.

[WD08] Wang, P., Domeniconi, C.: Building semantic kernels for text classification using wikipedia. In the 14th ACM SIGKDD, pp. 713–721, 2008.

[WL04] Wang, G. and Lochovsky, F.H.: Feature selection with conditional mutual information maximin in text categorization. Proceedings of the thirteenth ACM international conference on Information and knowledge management, 342-349, 2004.

[WN13] Whetzel, P.L., and NCBO Team: NCBO Technology: Powering semantically aware applications. J Biomed Semantics, 4(Suppl 1): S8, 2013.

[WPW95] Wiener, E.D., Pedersen, J.O., and Weigend, A.S.: A neural network approach to topic spotting. In Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval (Las Vegas, NV, 1995), 317–332, 1995.

[YL99] Yang, Y. and Liu, X.: A re-examination of text categorization methods. In Proceedings of SIGIR-99, 22nd ACM International Conference on

Research and Development in Information Retrieval (Berkeley, CA, 1999), 42–49, 1999.

[YP97] Yang, Y. and Pedersen, J.O.: A comparative study on feature selection in text categorization. In Proceedings of ICML-97, 14th International Conference on Machine Learning (Nashville, TN, 1997), 412–420, 1997.

[YY01] Yang, Y.: A study on thresholding strategies for text categorization. Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval, eds. W.B. Croft, D.J. Harper, D.H. Kraft & J. Zobel, ACM Press, New York, US: New Orleans, US, 137–145, 2001.

[ZGW05] Zhang Y., Gong, L., Wang, Y.: Chinese word sense disambiguation using HowNet. Lecture Notes in Computer Science, 3610/2005, 925–932, 2005.

[ZXJ07] Zhu, X. J.: Semi-supervised learning literature survey. http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html, 2007.

Acknowledgment

Stefania Dessì gratefully acknowledges Sardinia Regional Government for the financial support of her PhD scholarship (P.O.R. Sardegna F.S.E. Operational Programme of the Autonomous Region of Sardinia, European Social Fund 2007-2013 - Axis IV Human Resources, Objective l.3, Line of Activity l.3.1.)

La presente tesi è stata prodotta durante la frequenza del corso di dottorato in Informatica dell'Università degli Studi di Cagliari, a.a. 2013/2014 - XXVII ciclo, con il supporto di una borsa di studio finanziata con le risorse del P.O.R. SARDEGNA F.S.E. 2007-2013 - Obiettivo competitività regionale e occupazione, Asse IV Capitale umano, Linea di Attività l.3.1 "Finanziamento di corsi di dottorato finalizzati alla formazione di capitale umano altamente specializzato, in particolare per i settori dell'ICT, delle nanotecnologie e delle biotecnologie, dell'energia e dello sviluppo sostenibile, dell'agroalimentare e dei materiali tradizionali.