A New Framework of Optimizing Keyword Weights in Text Categorization and Record Querying

by

Harsh Singhal

A Thesis submitted to the

Graduate School-New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master Of Science

Graduate Program in

Industrial and Systems Engineering

written under the direction of

Dr. Wanpracha A. Chaovalitwongse

and approved by

_____

_____

_____

New Brunswick, New Jersey

May, 2008

Abstract of the Thesis

A New Framework of Optimizing Keyword Weights

in Text Categorization And Record Querying

By Harsh Singhal

Thesis Director:
Dr. Wanpracha A. Chaovalitwongse

In text mining research, the Vector Space Model (VSM) has been commonly used to represent text documents as a vector where each component is associated with a particular word in the documents. Assigning appropriate keyword weights in VSM has been critical in Information Retrieval (IR) and Text Categorization (TC).

Traditionally keyword weighting processes are unsupervised; that is, the knowledge of document's category is not leveraged to label the documents. Typically, each keyword weight is assigned using the term frequency – inverse document frequency (TFIDF) measure. Although the TFIDF measure has been proven effective in several text mining problems, it might not give the optimal classification power for IR and TC. In this thesis, we propose a new optimization framework to find the best keyword weights based on the proposed inter-class and intra-class similarity concept.

The optimal keyword weight can be viewed as the feature space projection where documents from the same category are best clustered together and separated from other categories. Subsequently, the category average (centroid) classification is employed to categorize text documents. The proposed approach is tested on two practical

applications: record query and text categorization. The record query application is slightly different from traditional IR problems as the goal is to find correlated (duplicate and master) text records. This problem was initiated by a telecommunication company where service engineers attempt to look for associations of the current defect problem in previously recorded problems in the database. Extensive experiments demonstrate that the proposed framework significantly improves the classification accuracy and provides balanced performance as measured on all text categories when compared to the standard TFIDF search. The text categorization application is tested on the Reuters news data set which is a gold-standard benchmark data set. The results show that our framework improves performance for the two applications considered, namely Information Retrieval and Text Categorization.

Acknowledgements

The research documented in this thesis was made possible by Dr. Wanpracha A. Chaovalitwongse who labored incessantly to work with me in explaining and detailing the proposed framework and the primary focus of this work. It is a matter of honor for me to have associated with Dr. W. A. Chaovalitwongse, my advisor and mentor for the learning opportunity and an experience which I will cherish for the rest of my life.

I would like to express my gratitude to Dr. Hoang Pham who has constantly guided this work in adopting directions that have proved conducive to the betterment and improvement of our research.

Liang Zhe and Andrew Rodriguez graduate students and my colleagues in the Industrial and Systems Engineering Department, Rutgers University who worked towards providing critical guidance and the necessary data to accomplish some of the tasks which this work drew upon in its nascent stages.

The Industrial and Systems Engineering Department at Rutgers University has provided me with a life experience that has molded my academic, social and personal character to greater levels of credibility and improvement. Dr. Susan Albin has over the many semesters provided me with great advice peppered with intense enthusiasm which always melted any trace of doubt in my mind about the uncertainties of the future.

I have drawn inspiration from the distinguished faculty members all of whom have pioneered incessantly to create an academic institution which attracts the best minds from all over the. I came looking for academic excellence and I found more than I could have ever asked for. Dr. E. Elsayed, Dr. D. W. Coit, Dr. T. Boucher, Dr. M. Gursoy, Dr.

T. Ozel and Dr. M. Tortorella have in their own individual ways satisfied my intellectual curiosity for which I will be ever grateful.

Ms. Ielmini, Ms. Pirrello and Mr. Lippencot have made trips to the department office and the shop floor pleasant with their ever smiling faces and constant co-operation in providing every form of assistance sometimes repeatedly and with great cheer.

Satish E. Viswanath, a friend indeed for having proof-read my thesis multiple times and making important corrections in the midst of a busy academic schedule while pursuing his PhD degree in the Biomedical Engineering Department, Rutgers University.

Dr. N.V.R Naidu, Professor and Head of Department, Department of Industrial Engineering and Management, M.S. Ramaiah Institute of Technology, Bangalore my alma mater where I had the privilege to pursue my undergraduate degree. Dr. Naidu embodied commitment most of all to the welfare of his students and displayed great mastery of the knowledge he tirelessly instilled in them. He is the inspiration for me having decided to pursue higher education.

It is a monumental task to express gratitude to every entity I have come across in my life here at Rutgers. I am grateful to my spiritual guide and my personal God Shri P. Rajagopalachari for all that has been showered upon me which has brought me to this stage in my journey.

Dedication

This Thesis is dedicated to my parents and my family who have always supported me in my endeavors.

Table of Contents

Lists of Tables

List of Illustrations

Chapter 1: Introduction

Data Mining and knowledge discovery in databases have attracted a significant quantum of research, industry and media attention in the recent past. Across a wide variety of domains data are collected and accumulated so much that the pace of such activities has gradually increased over time. It has become an urgent need to generate innovative computational theories and tools that provide an efficient extraction of relevant information from data. These theories and tools are part of the emerging domain of knowledge discovery in databases (KDD).

Generally, the KDD field seeks to develop methods and techniques for making sense of data. Interpreting low-level data (typically available in large volumes) to other forms which may be conducive to brevity, quantitative manipulation and descriptiveness, is a basic problem in a KDD process. The traditional method of converting data into knowledge relies on manual analysis and interpretation. For example, geologists may sift through remotely sensed images of planetary relief features and make speculations about the possibility of finding natural resources in certain geographical locations on the planet. In many other fields such as marketing, finance, health care and retail, the conventional approach to data analysis relies on one or more analysts becoming closely familiar with the data and creating a functional interface between the data and the users of the processed data. Such manual analysis is soon becoming inefficient due to the exponential growth in the volumes of data collected which are required to be analyzed. Databases are increasing in size in two primary ways: (1) the number of records or objects in the database and (2) the number of fields or attributes of an object. In medical diagnostic applications, the number of fields can easily be of the order of 10^5 or more, akin to the

possible number of genotypic and phenotypic traits found. Manual human analysis thus becomes inefficient when faced with such enormity of scale. Computational resources have provided effective means of gathering large amounts of data to seek meaningful patterns and informative and functionally advantageous structures from such massive amounts of data.

1.1 The Knowledge Discovery Process

At its core, the knowledge discovery process is the set of data mining activities used to extract and verify patterns in data. As described by Brachman and Anand, 1996, knowledge discovery takes place in a number of stages:

- Getting to know the data and the task: A significant stage where sometimes the data is extracted from multiple sources.
- Acquisition: Bringing the data into a pertinent environment for investigative analysis.
- Integration and checking: Confirming the expected form and broad contents of the data and integrating the data into the tools as needed.
- Data cleaning: Looking for glaring flaws in the data, their removal along with the removal of glaring flaws and insignificant outliers.
- Model and hypothesis development: Simple exploration of data and the derivation of new data attributes when needed. Appropriate model selection and hypothesis to test.
- Data mining: Application of the core discovery procedures to reveal patterns and new knowledge or to verify pre-developed hypothesis.

- Testing and verification: Assessing the discovered knowledge, including testing predictive models on test sets and analyzing segmentation.
- Interpretation and use: Integrating with existing domain knowledge, which may confirm, deny, or challenge the newly discovered information.

1.2 Applications of Data Mining

In considering the application of data mining, distinctions are not drawn based on the popularity of application areas and those mentioned below are successful domains in which data mining has been applied.

Astronomy is a prominent area of application in science. Notably, success was achieved by SKICAT, which provides astronomers with the capability of performing image analysis, classification, and cataloging of sky objects from sky-survey images [Weir *et al.*, 1995].

In marketing the market-basket analysis system described by Agrawal *et al.*, 1996, finds patterns such as, “If customer purchased A, he/she is likely to buy B and Z because A, B and Z are more often purchased together than other combination of products”. The revelation of such patterns is advantageous to retailers. Many financial analysis tools apply predictive modeling concepts (For example, regression, neural networks and decision tree induction [Spangler *et al.*, 1999]) for such activities as the creation and optimization of portfolios and trading models. Such applications are usually held confidential by their users and developers to maintain monetary advantages. For example, LBS Capital Management a fund-management firm, uses expert systems, neural networks and genetic algorithms to manage portfolios worth $600 million has known to

outperform the overall stock market [Hall *et al.,* 1996]. Carlberg & Associates have developed a neural network model for prediction the Standard & Poor's 500 Index [S&P 500] using interest rates, earnings, dividends, the dollar index and oil prices. The model was surprisingly successful, explaining 96% of the variation in the Index from 1986 to 1995.

1.3 Text Data Mining

Text exhibits a varied range of information in a form that is difficult to analyze using automated procedures. Text Data Mining or simply Text Mining is an extension of data mining, which is also known as knowledge discovery in databases, as mentioned earlier. Hearst, 1999, draws a distinction between text data mining and information retrieval (IR). The goal of information retrieval is to help users find documents that satisfy their information needs [Yates and Neto, 1999]. As described by Hearst, 1999, the problem of information retrieval is to retrieve the desired information that co-occurs with other information objects. A user maybe currently interested in the NASDAQ [NASDAQ COMP] and not in the Hang Seng [Hang Seng Index] which does not invalidate all descriptions of the Hang Seng as immaterial. The problem is retrieval of what is currently of interest to the user. As pointed out by Hearst, 1999, the purpose of data mining is to uncover or derive new information from data, pattern discovery and the separation of signal from noise. The process of retrieving a document that contains the information a user is in need of, does not imply a new discovery, and thus may not be a strictly data mining process. Similarly the process of categorization of a document to one of a pre-

selected/known list of categories may not be considered data mining in the strict application of the term.

Hearst, 1999, further discusses the results of various text processing activities that may yield tools to aid and improve the information retrieval process. Cutting *et al.,* 1992, discuss a method of clustering text collections giving a subject/theme overview. Query expansion is a tool to improve the effectiveness of the user query in creating a more effective information bridge to the text collection in retrieving relevant information. Work done by Peat and Willett, 1991, Xu and Croft, 1996, provide automatic term associations that are used in query expansion frameworks.

As described by Hearst, 1999, text categorization (TC) is a simplification of the specific content of a document into one (or more) of a set of pre-defined labels. Though this may not lead to the discovery of new information, this provides a compact summary of the document's overall information. As mentioned by Hearst, 1999, there is research in the domain of text categorization which conforms to the strict definition of data mining, and seeks the discovery of new trends and patterns for general purpose consumption. Feldman and Dagan, 1995, use text category labels to find 'unknown patterns' among text documents, pertaining to the Reuters newswire collection. The definitive idea is a comparison of category distributions in subsets of document collections. For example, the distribution of commodities in country C1 is compared against those of country C2 in trying to find interesting and hitherto unknown patterns.

The DARPA (Defense Advanced Research Projects Agency) Topic Detection and Tracking [DARPA TDT] initiative has an interesting task of On-line New Event Detection, whose incoming data stream are news stories in chronological order, the

output being binary (yes/no) for each story, made at the time of story arrival, reflecting the story being the first reference to a newly occurring event [Allan *et al.*, 1998]. Work carried out by Swanson and Smalheiser, 1994, in deriving new information from text collections. It was correctly hypothesized that experts read only a small subset of what is published in their domains and are often unaware of related developments in peripheral, yet germane fields. The possibility offered by text mining in revealing to experts information about their respective fields by mining for new and related insights from other fields was seen when the hypothesis proposed by Swanson and Smalheiser, 1994, was validated by Ramadan *et al.*, 1989, which supported the claim of the magnesium-migraine hypothesis formed due to text mining work in medical literature.

Efforts undertaken by Narin *et al.*, 1997, showed that the technology industry relies heavily on government funded research undertakings. Another noteworthy application of text mining is the LINDI (Linking Information for Novel Discovery and Insight) initiative which investigates the possibility of researchers using large text collections to reveal important information and the software systems to aid such a process. A very important and critical problem in molecular biology is the automation of the discovery process in finding the functions of newly sequenced genes [Walker *et al.*, 1998]. Human genome researchers carry out experiments to analyze the co-expression of thousands of novel and known genes at the same time. This massive collection of genetic information provides the possibility of finding novel genes which are interesting and which maybe co-expressed with already understood genes involved in disease causing processes. Text mining would be employed in this case to explore the biomedical literature in order to formulate possible hypotheses about such interesting genes. The

LINDI project allows the users to create and reuse sequential queries using simple graphical information when investigating large textual document collections. In the gene investigation example this would allow the user to specify a query sequence to be executed repeatedly on various co-expressed genes. Another system that provides such functionality is the Visage Interface [Derthick *et al.*, 1997].

Systems have been built which have interactive user interfaces to create a seamless link between the user and the advantages offered by text mining tasks. Chilibot [Chen and Sharp, 2004] is a specialized software for the PubMed literature database [PubMed] and rapidly identifies relationships between genes, proteins and other user-specified keywords.

IHOP (Information Hyperlinked Over Proteins) [IHOP, 2004] is a system that links sentences and abstracts with genes and proteins and creates a navigable resource which combines the advantages of a web based system in providing efficient scientific literature research. Text mining has been employed to predict the response of financial and other equity markets in response to the news. Most of the models discussed by Mittermayer and Knolmayer, 2006, forecast price trends and stock prices, exchange rates and equity indices in particular. Some of the models use features that are manually selected by domain experts to restrict the lexicon. The restrictions that are imposed on the use of the lexicon and its members is also encountered in the Information Retrieval task to which our proposed framework is applied. This restriction is advantageous since feature reduction no longer remains a primary concern and the intended user of the system provides the feature set which is most appropriate to the queries that the intended user will use eventually.

1.4 Scope and Contribution of the Thesis

Text data mining provides a broad framework and environment to improve information retrieval and text categorization tasks among other text mining tasks. This thesis documents the research that was carried out resulting in an optimization framework for the improvement of information retrieval and text categorization tasks.

The Vector Space Model (VSM) is the underlying document representational format used in our work. Figure 1.1 illustrates the Vector Space Model representation of text documents. The document-term vectors may represent the presence of absence of the term in the document by binary weights where 1 indicates the term's presence and 0 indicates the term's absence. The document-term vectors may also be constructed of real valued term weights where such values will indicate the term's frequency in the document in conjunction with the term's frequency across the collection of documents being considered. Finding inherent categories and improving document-category relationships in a text collection is the purpose of our work.

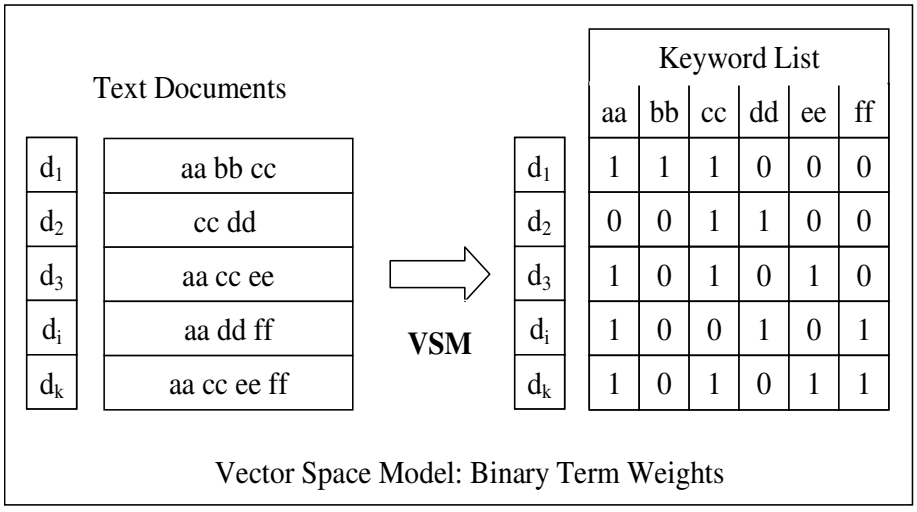

Figure 1.1: Vector Space Model (VSM) representation

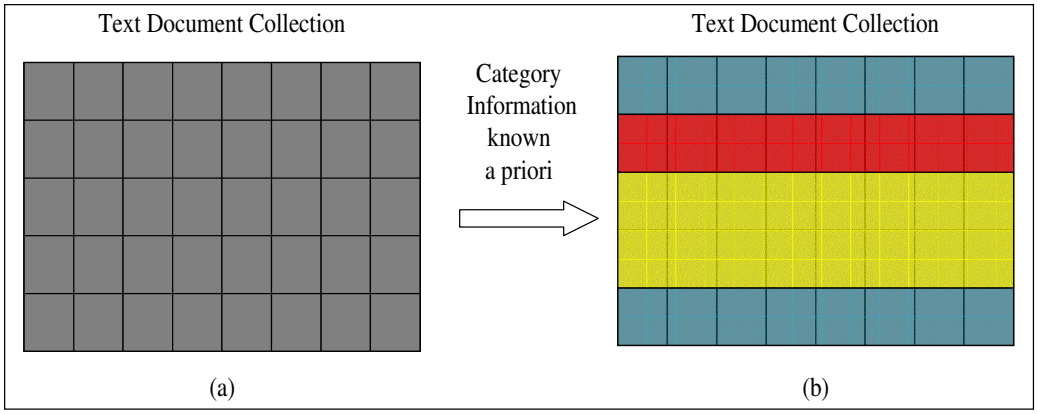

Figure 1.2: Known category information used for grouping documents

Figure 1.2(a) is the resultant VSM representation of a text collection where each row is a document belonging to the collection and each column is a term belonging to the term list. In Figure 1.2(b) documents in the VSM are grouped together based on additional document information. This additional information can be category labels of documents or the similarity between documents which creates a group of similar documents. Documents belonging to the same group are indicated by the different rows which have the same color.

Figure 1.3 summarizes the purpose of our work in establishing a new framework which makes the following contributions.

1) Document-Category relationship where the object is represented in terms of the numerical relation of its features to the chosen category centroid.
2) Leveraging the Object-Category relationship to create an optimization model which estimates term weights used to modify the term weights of the document collection thus improving retrieval relevancy to queries (IR) and rate of accurate classification of text documents (TC).

Considering information retrieval and text categorization independently of data mining tenets is useful in developing a framework whose sole purpose in not new information discovery but improvement of prevalent methods.

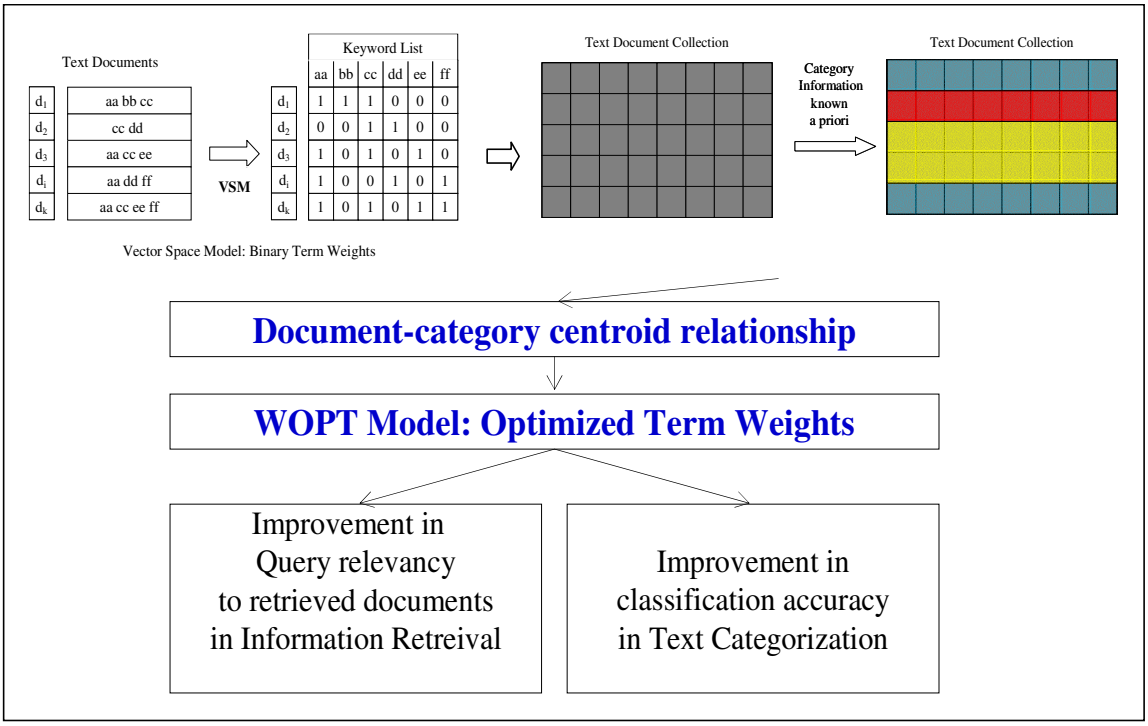

Figure 1.3: Summary of proposed framework.

The work documented in this thesis contributes towards improvements in Information Retrieval and Text Categorization systems on various levels of functionality. Representing documents in relation to the category centroids creates a combined identification of the individual document's feature set with the feature set of the category as a whole. An explicit representation of the category information for every individual document allows the proposed framework in estimating term weights that validate comparisons made between document-category relationships. This process weights optimal term values to provide improved retrieval and classification results. IR results are relatively dependent on similarity measures used to compare document-term vectors. The choice of similarity measure is highly correlated to the document collection, document integrity, document representation model and the requirements of the end user. The proposed framework shows improved performance based on criteria set by the end user of the IR system developed.

Text Categorization depends heavily on such application parameters including but not limited to:

1) Text document representation,
2) Choice of Text Classifiers,
3) Categorization/classification performance metric.

This work uses the VSM model for Text representation, but employs the category-centroid vector as the choice of classifier for the categorization application. This classifier quantifies the documents belonging to a particular category in terms of the feature weights across all the documents in that category. Unclassified documents are compared to these category-centroids to decide their assigned categories. The classification accuracy is computed by determining the proportion of documents from the test set that are accurately classified. As mentioned in this chapter, information retrieval and text categorization may not always yield new patterns or information, but could provide tools to enhance prevalent activities in their respective domains.

Our work focuses on the proposal of a generalized framework to improve such text mining tasks as retrieval and classification of text documents by leveraging additional information and estimating optimal modifiers to re-calculate document term weights.

Chapter 2: Literature Review

This chapter discusses research and related undertakings which have inspired our work and have a bearing to the functional aspects of our proposal. Categorization and retrieval of documents are text mining activities which to a certain extent are inter-dependent as far as underlying application methodologies are concerned and developments in either field have been successfully used in the other.

A concise definition of information retrieval is given by Lancaster, 1968: "An information retrieval system does not inform (i.e., change the knowledge of) the user on the subject of his inquiry. It merely informs on the existence (or non-existence) and whereabouts of documents relating to his request". It is evident from this description that the evaluation of IR systems in the 'real world' is based on 'user satisfaction' and can be quantified by the cost the user is willing to incur for its use. Experimental IR systems, such as our framework are evaluated using specific metrics of evaluation that are developed in keeping with the user's preferences, and may differ for other IR systems.

The effectiveness of an IR system has been traditionally measured in terms of recall (the proportion of all relevant documents a system is able to retrieve) and precision (the proportion of retrieved documents that are relevant). In converting text information to a format easily manipulated by an automatic retrieval or classification system, we use the vector space model due to Salton *et al.*, 1975. The vector space model computes a measure of similarity by defining a vector that represents each document, and a vector that represents the query. As discussed by Grossman and Frieder, 1998, the vector space model represents the idea that an approximate meaning of the document is conveyed by the words used. The representation of a document by a vector makes it possible to

compare documents with queries to determine mutual similarity of their content. Based on the similarity between objects, we may then create groups of similar objects. Measures of similarity are designed to quantify the similarity between objects (document vectors) and the possibility of selecting groups of objects wherein the objects within a group are more like the other objects in the group, than any object outside the considered object's parent group.

To quantify similarity and rank documents in a retrieval process, web data mining has provided many advances and innovations to provide improved Information Retrieval and Text Categorization. The web is a massive storehouse of data that can be manipulated in a number of different ways, and structured to suit mining algorithms to create and reveal useful information. The web provides data, rich in multi-media content and massively textual at the same time. In order to mine multi-media data, it would be necessary to develop similarity measures that draw heavily upon image/pattern recognition and cognitive machine learning, where audio-visual content of a non-textual nature would be presented for automated retrieval and semantic classification. Promising developments in multi-media web mining have been achieved by Hauptmann, 1999. It is indeed difficult to approach the development of a unified framework which tries to represent, solve and learn from multi-media data, as correctly claimed by Uthurusamy, 1996, and Shapiro *et al.*, 1996. Multi-media data have an exponentially higher quantum of attributes when compared to pure textual data.

The similarity measure increases as the number or proportion of attributes increases. In the vector space model, a document-vector consists of vector elements akin to words (terms/features/attributes). The vector space model involves creating a vector

which represents the terms in the document and the choice of a similarity measure between vectors. Traditionally, the similarity between vectors is determined by measuring the size of the angle between them. This angle is determined by the inner product, and any monotonic function of the angle will suffice. This similarity co-efficient can be computed in a number of ways, but the inner product is used generally. Summarily, a document and a query (other document) are similar to the degree that their associated vectors point in the same direction. As mentioned earlier, in the vector space model, documents are represented as vectors of binary or real valued term weights. The relevance of a document to a query (similarity between documents) is determined in terms of proximity of similarity measures such as a distance metric or angular similarity measures [Jones and Furnas, 1987].

Kuhns, 1964, describes various experiments with similarity measures. Typically a query representation is matched with document representations, and those documents residing within a particular similarity threshold are retrieved. As mentioned by Dubin, 1996, these similarity measures have two attractive features:

1) Similarity measures relieve a user of the responsibility of exploring various term combinations. The user simply enters terms, possibly with weights reflecting relative importance. Documents can be matched directly against other documents known to be relevant.
2) Similarity measures take advantage of richer representations (For example, term weights based on occurrence frequencies).

Dubin, 1996, also describes the broad classes into which similarity measures can be grouped. The four types of measures are:

1) Angular measures (For example, Cosine measure),

2) Distance measures (For example, Euclidean distance),

3) Association coefficients (For example, Jaccard's coefficient),

4) Probabilistic measures.

In this work association coefficients are employed when we consider binary term weights for document-term vectors. For binary document-term vectors the similarity measure is the count of the number of common keywords between compared document-term vectors. In case of real valued term weights for document-term vectors, the vector inner product is used to determine the similarity between document-term vectors.

This work primarily tries to improve group-membership integrity described subsequently. Localized collections of documents are formed when there is a semantic identity that the documents may possess which allows for divisions in the form of clusters and groups. In the IR task that has been considered in this work, we form clusters of documents and enhance the cluster integrity by the proposed optimization framework.

The learning or training process is supervised in that it learns from the training clusters/groups, and by considering the integrity of group memberships in the light of document-term vectors that are part of the group and their term weights (binary or real valued), the optimization framework determines term weights from the training set of document-term vectors which will enhance or 'tighten' the clusters/groups and cause a distinct separation between clusters.

Every document-term vector may be considered a point in *n* space, where *n* is the cardinality of the set of features. When document identification labels are used, such as category information of documents, their thematic labels and the like, it becomes possible to create clusters of documents based on their identification labels. These clusters may overlap with other clusters and may not provide the degree of separation between clusters that is desired. The optimization framework proposed learns from the clusters in the training sample and the document database to re-assess the document term weights to provide better object-cluster relationships and grouping of document-term vectors. These term weights, estimated by the optimization framework is also used to scale the predetermined term weights in order to improve cluster integrity.

The IR task that has been considered here is a very specific undertaking for an IT firm in California, USA. This work also considers the application of the proposed optimization framework to text categorization, by using the Reuters News Corporation dataset.

2.1 Cluster Hypothesis

Categories or groups are formed by a collection of objects exhibiting a common attribute or a set of common attributes. Such commonalities are sometimes evident as in the case of document category labels, and may not be very evident as in the case of the distribution of specific terms/keywords in documents belonging to various categories. The clusters/categories that are considered in this work, merely recognizes existing groups based on document category labels provided *a priori*.

In trying to recognize common attributes which would allow efficient group formations of objects, work done by Brin and Page, 1998, has resulted in what is now unarguably the most efficient and accurate search system for the World Wide Web, the Google Search system. Google provides an efficient and state of the art methodology to bring web content together on being prompted by the user with a set of search terms. These search terms, or query string is compared with the documents (web-pages/web logs/bulletin boards) available on the Internet, and the results thus generated have been found to be more query-relevant to a large number of users than those results provided by any other web search system available today. This reflects the robust and efficient process of finding commonalities between web content in a way that provides high relevancy and similarity to user query and user provided information, which is a matter of continuous improvement in any Information Retrieval process.

It is evident that the rate of content acquisition on the World Wide Web is near-exponential and such information must be efficiently and speedily indexed to be available to retrieval systems as quickly as they are added to the collective web repository. This is an application area for developing and improving accurate classification and highly relevant retrieval systems.

As mentioned earlier, considering documents as points in vector space gives rise to what is called the Cluster Hypothesis [Salton, 1989]. The hypothesis claims that the retrieval in the neighborhood (measured by similarity) of a known relevant document increases the chances of retrieval of other relevant documents. Salton, 1989, describes the cluster hypothesis by stating that it is valid “when associations between documents convey information about the joint relevance of documents to queries in a collection”.

This means that if a document is relevant to a certain query, other relevant documents will be found in close proximity to it in the document space.

We consider the document proximity space to be the collection of documents that are identified by their category information, or a binding relationship between the documents that puts them in a singular group. This additional information is exploited when we consider the retrieval and categorization tasks. Category information makes this a supervised learning optimization framework. We merely seek to determine document-term weights which will create tighter, more consistent clusters and increase the separation between clusters, by re-locating objects in the document-vector space with greater conformity to clusters that are formed based on their category information. This will be achieved by representing document term-vectors in terms of their membership with their parent category and to other chosen categories.

Rijsbergen, 1975, summarizes the discussion on the use of clustering in IR and lists the conditions that must be kept in mind when choosing a clustering method:

1) The method creates clusters which will not be altered significantly when objects are incorporated in the future.
2) The technique must be immune to slight changes in object representation, and not affect clustering significantly as a result of these changes.
3) The technique remains independent of the initial ordering of the objects.

Rijsbergen, 1975, points out two distinct approaches to clustering:

1) The clustering is dependent on the similarity measure of the objects being clustered.
2) The cluster technique is consequent to the object's description.

The object's (document-term vector) description is used in the optimization framework and the learning process employs category information for each object to describe the object's relationship to the centroids of chosen categories. Subsequently the objects are re-distributed in the document space by estimating term weights which are used as optimal modifiers for terms in the document vectors.

Let us consider document-term vectors that have their term weights, defined in three different ways:

1) Binary term weights where term weights can be either 1 or 0, where 1 represents the presence of the term in the document and 0 represents its absence.
2) Real valued TFIDF (Term Frequency Inverse Document Frequency) weights, where term weights are real valued, and represent by their magnitude, term frequency in the document in conjunction with term frequency across documents in the document collection.
3) Real valued global weights, where term weights are real valued and global, in that, the weights for every term is identical across all document-term vectors, in the document collection. The proposed optimization framework estimates such global term weights or what we call the estimated optimal modifiers.

The optimization framework proposed is applicable to document-term vectors whose term weights correspond to the types enumerated above, to create clusters with greater inter-cluster distance and smaller intra-cluster distances.

2.2 Cluster Representation

The proposed framework details the representation of the object in relation to it's parent category and to the other non-parent categories. The proposed mathematical programming model will estimate the term weights for the object and satisfy those constraints which seek to increase the similarity of the object with its parent category and reduce the similarity of the object with other categories.

We identify the similarity measures employed in this study for three different types of weights that are encountered for document-term vectors:

1) For binary term weights, the similarity measure (S_{Bin}), will be a count of common terms for the documents compared. For example, d_i =< 1 0 0 0 1 1 1 0> and d_k =< 1 0 0 0 1 1 0 1> are two document-term vectors. The similarity measure is the number of common keywords defined as:

$$S_{Bin}(d_i, d_k) = \mid j \mid, j \in \{J\},$$

where $\mid j \mid$ is the cardinality of the set of common keywords/terms between documents d_i and d_k and $\{J\}$ is the keyword list.

2) For real-valued, TFIDF term weights, we consider the inner product. For example d_i =< $w_{i,1}$ $w_{i,2}$ .. $w_{i,j-2}$ $w_{i,j-1}$ $w_{i,j}$ > and d_k =< $w_{k,1}$ $w_{k,2}$ .. $w_{k,j-2}$ $w_{k,j-1}$ $w_{k,j}$ > are two document term vectors with real valued local weights. The similarity measure, $S_{\Re}$ in this case is defined as:

$$S_{\Re}(d_i, d_k) = \sum_{j \in \{d_i \cap d_k\}} w_{i,j} \cdot w_{k,j},$$

where j is the set of common features/terms for the two documents compared.

3) For real-valued, global term weights, we consider sum of weights. For example

$d_i =< w_1\ w_2\ ..\ w_{j-2}\ w_{j-1}\ w_j >$ and $d_k =< w_1\ w_2\ ..\ w_{j-2}\ w_{j-1}\ w_j >$ are two document term vectors with real valued global weights. The similarity measure, $S_{\Re}$ in this case is:

$$S_{\Re}(d_i, d_k) = \sum_{j \in \{d_i \cap d_k\}} w_j ,$$

where j is the set of common terms for the documents d_i and d_k.

In their definitive work on similarity measures and their implications in document comparisons Zobel and Moffat, 1998 categorically claim that no similarity measure was found that could be considered a good performer under all circumstances in its application to query processing or text categorization. Minor variations such as different logarithmic base values caused pronounced changes in the performance of the similarity measure. Zobel and Moffat, 1998 propose a measure that is required to be customized to each individual query, which seems highly implausible to implement in practical systems. In the determination of a suitable similarity measure, such factors as type of data, type of query, the individual query itself, the evaluation metric and the conditions that the 'answer' must meet, play a decisive role. This causes great variability rendering the attainment of a single measure which would perform better than two thirds of what is known in the literature, impossible.

The implications of this claim made by Zobel and Moffat, 1998 are that different similarity measures could result in different sets of results may be obtained which may influence the performance measures used in the retrieval and classification tasks. It is hence re-iterated that what is proposed here is a generalized framework to estimate term weights or optimal term weight modifiers in improving retrieval and classification of text

documents. The choice of similarity measures, classifiers, initial term-weighting schemes and the analysis data set can be chosen based on user preferences.

2.3 Object-Category Relationships

Rijsbergen, 1975, discusses the cluster representative, which has been called cluster profile, classification vector, or centroid. The category classifier is the cluster representative or category centroid vector. It is the vector that represents the collection or overall category weight (total/average) of the terms in the documents that belong to that cluster. When a new object needs to be assigned to a certain cluster, its similarity measure is determined (For example, dot product) with the centroids of the existing clusters and the object is assigned to the cluster with which it shares the highest similarity. Rijsbergen, 1975, states it must be near to every object in the cluster in some average sense which is why the term centroid is used. The category is formed from those documents which have common identifying information that is known *a priori*, which puts them together in a group/category. It is true that most documents will subscribe to multiple category identities, but this work considers those objects in the training set that belong only to a single category.

Documents in category C_b $\forall i \in \{C_b\}$			
Terms→ *Documents↓*	*1*	*j..*	*|J|=3*
d_1	$w_{1,1}$	$w_{1,j}$	$w_{1,\|J\|}$
d_2	$w_{2,1}$	$w_{2,j}$	$w_{2,\|J\|}$
d_3	$w_{3,1}$	$w_{3,j}$	$w_{3,\|J\|}$
d_4	$w_{4,1}$	$w_{4,j}$	$w_{4,\|J\|}$
.	.	.	.
d_i	$w_{i,1}$	$w_{i,j}$	$w_{i,\|J\|}$
Category centroid classifier, $\tilde{C}_b$	$\sum_{i\in\{C_b\}} w_{i,1}$	$\sum_{i\in\{C_b\}} w_{i,j}$	$\sum_{i\in\{C_b\}} w_{i,\|J\|}$

Table 2.1: Centroid Classifier

Figure 2.1 describes the re-representation of documents in the document space in relation with the centroid of the document's parent category which is defined as the document-intra category relationship. This is illustrated in Figure 2.1 where documents when re-represented in relation to the centroid move closer to the centroid.

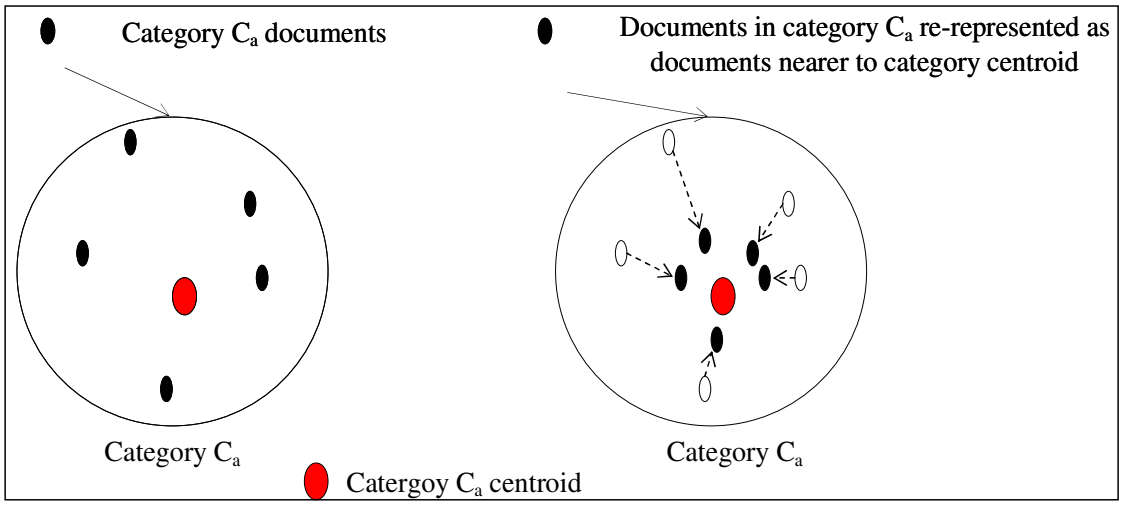

Figure 2.1: Documents in C_a shown nearer to category centroid.

This relationship is defined and used in this framework in the form of logical and model constraints to re-calculate the term weights which will locate the documents close to the category centroid.

Three ideas are derived from the observations which are as follows.

1) The document is represented by a vector of term averages where the average value indicates the document's term proximity to the category centroid.
2) The classifier is the cluster representative, or the cluster/category centroid.
3) The document-category relationship vector can also establish the document's proximity to its non-parent categories.

It is important to understand that this study does not propose a strict clustering procedure. Category information is used to identify existing groups of documents, and by

utilizing object-cluster representations, cluster centroids and the object-cluster similarity and dissimilarity representations, global term weights/optimal term modifiers are estimated, which by modifying document-term weights would redistribute the objects in the document space to form consistent and distinct clusters.

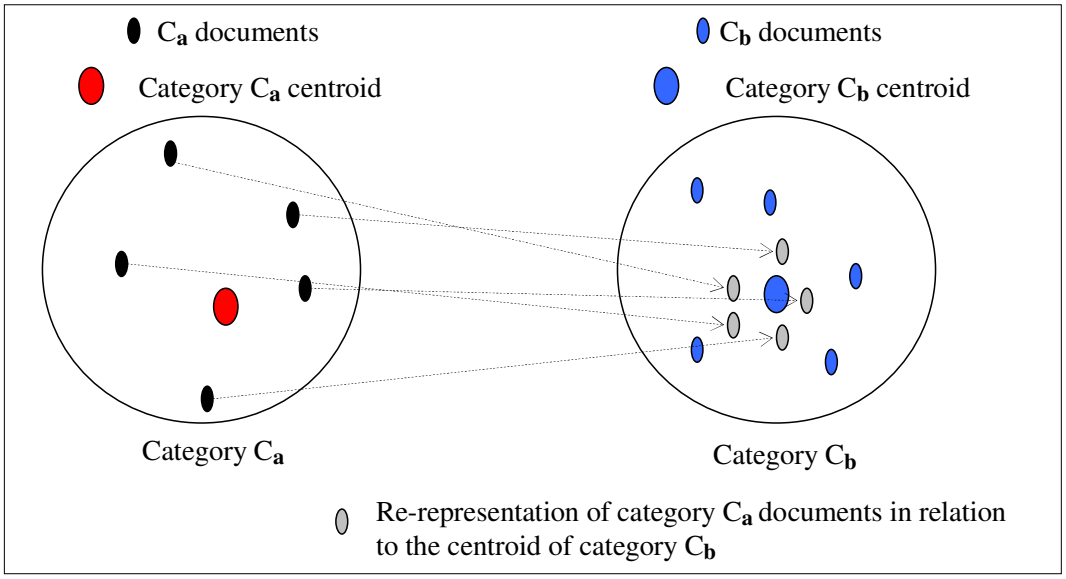

Figure 2.2: Documents in C_a shown nearer to category C_b centroid.

Figure 2.2 illustrates the representation of documents belonging to category C_a in relation with the centroid of category C_b to construct the document-inter category relationship. The proposed framework estimates the term weights by comparing the document intra-category and document inter-category relationships. The optimization model is constructed to estimate term weights such that the re-representation of documents closer to the centroid of their parent category is made attractive and suitably constrained as compare to the re-representation of the documents closer to the centroid of a non-parent category.

Figure 2.3 illustrates the un-optimized scenario where classification accuracy may suffer and documents may be erroneously classified to other categories more often than

preferred. Figure 2.3 shows the classification of documents in category C_a to categories C_a and C_b. The number of documents that are in-accurately classified is to be reduced, which the proposed framework achieves.

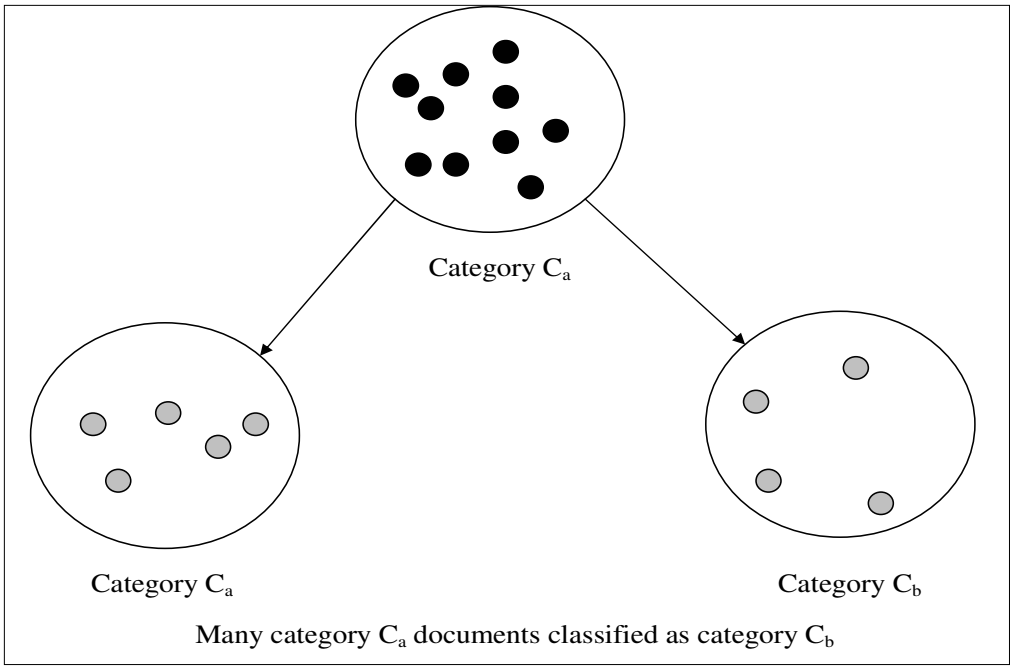

Figure 2.3: Un-optimized categorization of documents.

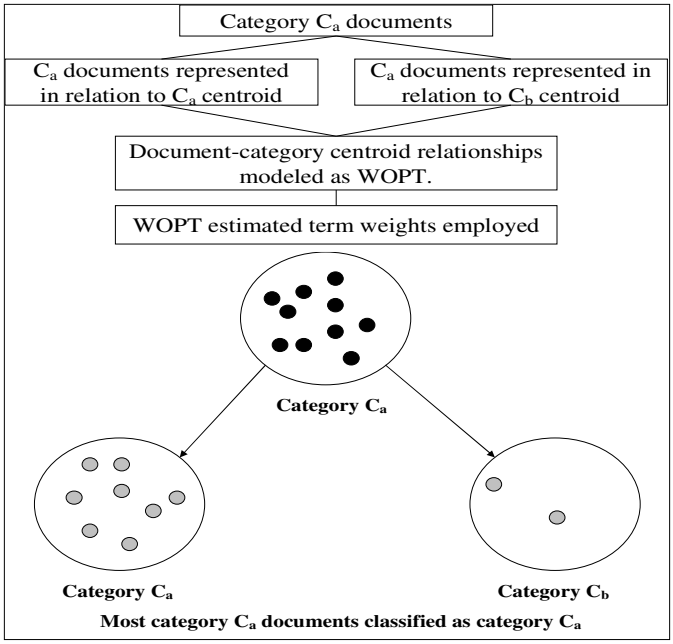

Figure 2.4: Proposed framework to improve retrieval and classification

Figures 2.2, 2.3 and 2.4 illustrate that the proposed framework utilizes information of a document vector on two functional levels.

1) Document's position in *n* space,
2) Document's category identity.

The document's position is fixed by its associated term vector. The category identity of a document is considered additional information. Other information which would group documents together is important and used in the proposed framework. It will be shown that term weights which are binary or real-valued in nature provide the same semantic information and differentiating capability when employed in the proposed optimization framework. This proves that term frequency in individual documents and term frequency across a document collection may not be the most effective means of discerning documents as being dissimilar to other documents. It is suggested that documents need to be differentiated utilizing their term weights in close conjunction with their category information. We also define the utilization of centroids of categories to which an object does not belong, in order to create logical constraints which give rise to model constraints that are used to estimate optimal term weights. Cluster-object relationships are central to the framework that is proposed. The optimization model incorporates document-category relationships as constraints which when satisfied will yield term weights or optimal modifiers to improve retrieval and categorization task. Every object exists in *n* dimension document space, *n* being the collective number of terms that are considered for the document collection.

Representative vectors for documents are shown in Table 2.2. These vectors will provide information of the document on an individual and category-centric basis.

Document Information	Representative Vector
Document	d
Document belonging to category C_a	d_i where i indexed on members of the set $\{C_a\}$ comprising of documents belonging to category C_a and $i = 1..\|C_a\|$
Document's term identification vector	$\vec{d}_i = < t_2\ t_{23}\ t_{142} \ldots t_{j_i} >$ where j_i is the set of terms that are present in document d_i
Document's term-weight vector	$\hat{d}_i = < w_{i,2}\ w_{i,23}\ w_{i,142} \ldots w_{i,j_i} >$ where $w_{i,j_i} \in \{0,1,\Re\}$ are term weights (binary or real valued).
Document category relationship vector: Intra-category.	$\tilde{d}_i = < w_{i,2}\ w_{i,23}\ w_{i,142} \ldots w_{i,j_i} >$ is the vector quantifying relationship of d_i with centroid of parent category, C_a
Document category relationship vector: Inter-category.	$\tilde{d}_{i,C_b} = < w_{i,2}\ w_{i,23}\ w_{i,142} \ldots w_{i,j_i} >$ is the vector quantifying relationship of d_i with centroid of category C_b.

Table 2.2: Document's representative vectors

Constructing the document intra-category relationship represented by vector $\tilde{d}_i$ we first construct the document intra-category centroid defined in Table 2.3.

Category C_c *documents* $\forall k \in \{C_c\}$			
Terms→ *Documents*↓	*1*	*j*	*J*
d_1	$w_{1,1}$	$w_{1,j}$	$w_{1,J}$
d_2	$w_{2,1}$	$w_{2,j}$	$w_{2,J}$
d_3	$w_{3,1}$	$w_{3,j}$	$w_{3,J}$
d_k	$w_{4,1}$	$w_{4,j}$	$w_{4,J}$
.	.	.	.
$d_{\|C_b\|}$	$w_{i,1}$	$w_{i,j}$	$w_{i,J}$
Document intra-category centroid for document $d_m \in C_c$ $\tilde{C}_c - \hat{d}_m$	$\sum_{\{C_c\}-m} w_{k,1}$	$\sum_{\{C_c\}-m} w_{k,j}$	$\sum_{\{C_c\}-m} w_{k,J}$

Table 2.3: Document intra-category centroid

Table 2.3 shows the preliminary construction of the document intra-category centroid for a member document d_m belonging to category C_c. It is seen in the construction of the document intra-category centroid for the document $d_m \in C_c$ the

components of the centroid are the sum of corresponding term weights of all documents in category C_c not considering document $d_m \in C_c$. The vector $\tilde{d}_m$ which is the intra-category relationship for document $d_m \in C_c$ is defined as:

$$\tilde{d}_m = \frac{\tilde{C}_c - \hat{d}_m}{|C_c| - 1} \cdot \vec{d}_m .$$

Each component of $\tilde{d}_m$ is the product of the term identifier and a numeric quantity that quantifies the term's relationship to the centroid of C_c.

In considering the inter-category relationship of any document it is important to indicate which non-parent category the document is being compared with. The inter-category relationship for document $d_m \in C_c$ to the non-parent category C_b is represented as $\tilde{d}_{m,C_b}$ and can be constructed as:

$$\tilde{d}_{m,C_b} = \frac{\tilde{C}_b}{|C_b|} \cdot \vec{d}_m .$$

Each component of $\tilde{d}_{m,C_b}$ is the product of the term identifier and a numeric quantity that quantifies the term's relationship to the centroid C_b.

The inter- and intra-category relationships for every document as constructed above, provides the category information for every document in relation to its parent and non-parent categories. This forms the proposed representation of a document in the form of a vector with the use of additional information which in this case is the document's category information. These relationships are modeled into the WOPT model framework giving rise to model constraints which reflect the logical constraints in the form of inter-cluster and intra-cluster relationships just described.

2.4 Term Weighting Paradigms

TFIDF (Term Frequency Inverse Document Frequency) is a term weighting measure that uses certain assumptions and statistical concepts in the estimation of real valued weights for terms in a document as a way of discerning their importance and semantic purport. TFIDF subscribes to the idea that a term that occurs infrequently should be given a higher weight than a term that occurs frequently. Weights are assigned to terms based on the frequency of their occurrence in the document in which it was contained. In the 1970s, it was found that relevance rankings improved if collection-wide weights were included. Small document collections were used to conduct the experiments, it was nevertheless concluded that, "in so far as anything can be called a solid result in information retrieval research, this is the one" [Robertson and Jones, 1976].

TFIDF weighing schemes are unsupervised for they do not consider category information or query relevancy. TFIDF weighting schemes have found considerable attention and many formulae exist which arrive at TFIDF weights. The word frequency in the document is proportional to the word's relevance to the document (Term Frequency). Salton and Buckley, 1998 carried out extensive experiments to improve on the basic combinations of TFIDF weights. Their estimator for term *j* in document *i* with good performance characteristics is:

$$w_{ij} = \frac{(\log TF_{ij} + 1.0) * IDF_j}{\sum_{j=1}^{t} [(\log TF_{ij} + 1.0) * IDF_j]^2},$$

where TF_{ij} is the number of occurrences of term T_j in document d_i (Term Frequency) and DF_j is the number of documents that contain T_j. The Inverse Document Frequency is $IDF_j = \log(\frac{d}{DF_j})$ where *d* is the total number of documents in the collection.

It was further concluded by Salton and Buckley, 1998 that a single matching term with a high frequency would skew the effect of remaining matches between a query and a document, which necessitated the need for normalization. The normalization measure $\log(TF)+1$ reduces the range of term frequencies and the skewness encountered.

Normalization removes the bias a longer document may have over a shorter one. This belief has been challenged in the work done by Singhal, 1997, where experiments prove that most documents judged to be relevant were found to be longer documents. The obvious conclusion is that a longer document simply has more opportunity to have some components that are relevant to a given query. Singhal, 1997, proposes a correction factor based on experiments which compare the likelihood of relevance to the likelihood of retrieval for documents which were already known to be relevant to a set of queries. The experiments showed that there must be a document length for which the probability of retrieval equals the probability of relevance. Based on these findings Singhal, 1997, proposed a correction factor which does fairly well for short and moderately long documents but extremely long documents tend be more favored than those without any normalization. Thus normalization is unnecessary since the choice of similarity measure affects retrieval and classification performance more significantly than the implementation (or the absence of) a normalization scheme. In supervised term weighting research Debole and Sebastiani, 2003, have described various measures of term selection and term weighting. It is interesting that our framework by consequence does term selection or feature reduction. This is achieved since the optimization framework does not estimate weights for all the terms in the WOPT model. Only a subset of terms in the

WOPT model are given non-zero values. The terms that are zero valued are dealt with a method which will be discussed towards the end of this chapter.

In the IR task that is considered in this work it has been shown that categories can be formed by grouping documents based on the similarity scores between them. The choice of similarity measure influences which documents will be considered similar to other documents. This will create groups with elements in the group being different for different choices of similarity measure. When retrieving documents from a database when prompted with a query the choice of similarity measure will also determine the documents retrieved. This creates a scenario where we are to reckon with changing category document relationships which may not be conducive to the training process from which the proposed optimization model is derived. This is the reason we use a similarity measure in the IR task which is best suitable to the functional aspects of the document collection and which satisfies the IR system user's preferences.

Supervised term weighting is directly influenced by category information. Such category information is utilized in many ways [Debole and Sebastiani, 2003] to weight terms in documents belonging to these categories, such that terms are weighted reflecting their semantic importance in differentiating between documents belonging to dissimilar categories. Supervised term weighting thus has a direct bearing on the IDF (Inverse Document Frequency) measure of a term. A term's document frequency measure will be affected under the dynamic assessment of the elements of the document collection when considering category information. Category information creates localized document collections which we identify as document groups. Categories can thus be formed based

on a variety of attributes chosen one at a time or many together which consequently will create new group boundaries and alter document-category relationships.

In determining the document-frequency of a term in a document belonging to a particular category work done by Soucy and Mineau, 2005, bears similarity to our current work. They estimate term weights based on the frequency of a term as it appears in documents belonging to various categories. The statistical range of the proportional frequency of occurrence of the term across categories is also considered. This makes it possible to estimate the range of the proportional frequency of a term in multiple categories and thus creating the possibility of choosing the proportion value which best quantifies the term's relation to a certain category.

Their method is described as:

Let x_t be the number of documents containing the term *t* in the text collection and *n*, the size of the text collection. The proportion of the documents containing the term is:

$$\tilde{p} = \frac{x_t + 0.5 z_{\alpha/2}^2}{n + z_{\alpha/2}^2},$$

where statistical values are appropriately assumed based on the size of the collection. They construct the above mentioned proportion for each category $\tilde{p}_+$ for the documents that belong to the category. $\tilde{p}_-$ is the proportion for the documents that do not belong to the category. MinPos is the lower range of the confidence interval for $\tilde{p}_+$ and MaxNeg is the higher range for $\tilde{p}_-$.

The relative proportion of MinPosRelFreq is defined as:

$$\text{MinPosRelFreq} = \frac{\text{MinPos}}{\text{MinPos+MaxNeg}}.$$

The strength of the term *t* in category + is defined as:

$$str_{t,+} = \begin{cases} \log_2(2.\text{MinPosRelFreq}) \text{ if MinPos} > \text{MaxNeg}, \\ 0 \text{ otherwise.} \end{cases}$$

Global term weights are obtained which are used in the IDF component of the TFIDF term weight.The global weight is obtained as:

$$\text{maxstr}(t) = \left(\max_{c \in Categories} (str_{t,c}) \right)^2.$$

Their proposed term weight is defined as:

$$ConfWeight_{t,d} = \log(tf_{t,d} + 1)\text{maxstr}(t).$$

We have used the *ConfWeight* term weighting scheme to modify the term weights for the TFIDF weighted document term vectors in the TC task and have improved classification results by implementing our proposed framework.

We have used the category centroid classifier for the different term-weighting schemes with which document-term vectors are constructed. Soucy and Mineau, 2005, propose supervised term weighting schemes based on statistical analysis of document distribution in categories and use support vector machines [Joachims, 1998] and k-NN (k-Nearest Neighbour) [Ham, 1999] techniques of classification. Our work compares the performance of the term weighting scheme proposed by Soucy and Mineau, 2005, when used with the centroid classifier before and after the implementation of our proposed WOPT model. The classifier used in the categorization task can also be considered as the

nearest neighbor classifier. When a test document is encountered it is classified to the category whose centroid it is nearest to.

Work done by Jin *et al.*, 2005, along similar directions in the weighting of terms to increase similarity between document vectors utilizes a category vector which represents the category membership of the document. They propose a quadratic programming framework which seeks to iterate over pairs of documents in trying to estimate term weights which will improve the similarity measure of the pair under consideration concurrently weighting the terms in the category vector to improve the similarity of the document category vectors also. This approach becomes infeasible in two situations. One situation is when the feature set is large and creates scalability issues, which is avoided by estimating feature weights only for a sample of features and executing this scheme iteratively. When the document collection is very large pair-wise document-term vector comparisons to estimate term weights imposes great demands on computational resources.

Our work differs with the model proposed by Jin *et al.*, 2005, since it considers documents which identify with only one category label. Moreover, we do not concern ourselves with feature reduction since feature reduction is a natural consequence of the WOPT model estimating certain terms to have zero values. We consider the estimation of term weights which identify documents more closely with their categories and are not directly concerned with the similarity of documents with other documents in the category or text collection. WOPT estimated global feature weights (optimal modifiers) are used to re-calculate term weights to improve classification accuracy and retrieval relevancy. In most text mining applications the reduction in feature space is desired to provide a

manageable scale of implementation and retain only informative or semantically important and highly relevant features. The framework proposed in this work results in the mathematical programming model assigning non-zero weights to a subset of the features present in the model. The features (terms) that are assigned zero weights can be considered to be those features which were not included in the process of feature selection and weight estimation. In our analysis when the zero-valued terms were ignored during the IR task and TC task we encountered a slight decrease in the performance metrics as opposed to when the zero-valued terms were manually weighted after the WOPT model run. The weight that was assigned to the zero valued terms was the minimum of the non-zero weight values estimated by the optimization model. The inspiration for this approach was obtained from Good, 1953. The terms that are zero-valued by the optimization model are not ignored but given the minimum weight. What this provides is an opportunity to create weighing measures that deal with the zero-valued terms of the solved optimization model in order to improve performance measures.

In summarizing, the proposed framework utilizes the document-term vectors with the weight values in creating numerical relationships between documents and category centroids. This provides the necessary constraints for the WOPT optimization model. The logical comparisons made between document-category relationships gives rise to model constraints. The model when solved to optimality estimates term weights which improve classification accuracy and retrieval relevancy.

Chapter 3: Information Retrieval Models and Applications

We shall now present the proposed framework and the optimization model WOPT for an IR task. This task was commissioned by a large Information Technology firm in California (USA).

3.1 Preliminary Approach

The dataset obtained from the company was a collection of semi-structured text documents. These documents were problems encountered by customers using products and services provided by the company. When customers would encounter problems that needed diagnosis and troubleshooting, they would contact the company's customer-care center. Technicians would then provide solutions to the customers and document the exact nature of their problem, problem summary, initial diagnosis and final solution statement in the form of natural language text. The partial structuring was provided by specific fields under which the problem was described by mentioning the component, product specifications and other user specified data.

The company held these documents in their database and would recall documents on receiving customer-complaints which referred closely to the documents already recorded and maintained in the database. This would provide the technician with information that was employed when solving similar problems in the past allowing the troubleshooting of recurring problems at a faster pace. To retrieve relevant documents the company needed the creation of an application which would convert these text documents into an appropriate format, conducive to manipulation by IR practices and methodologies. A small subset of the company's database was provided to us as a test

dataset. We were also provided with additional information which was used to identify clusters and groups to be suitably used in WOPT.

3.2 Document-Category Relationships

The text documents were mined for keywords and once the keyword collection was created and pruned the documents were transformed into document-term vectors. The company determined empirically pairs of documents that addressed the same type of user problem though their term vectors were not identical. These pairs were called *'Duplicate-Master'* pairs of documents. All *'Duplicate'* documents were isolated and collected in the duplicate document collection. The *'Master'* documents were held together with those documents that were not identified as duplicates. This created two collections of documents:

1) *'Duplicate'* documents collection called *CollDup*,
2) Collection of documents that are *'Master'* documents along with all those documents that are not *'Duplicate'* documents called *CollOther*.

The document-term vectors were created by mining each document and referring the word encountered against a list of keywords determined by the company's technical personnel. After all documents were converted to term-vectors we determined the recall proportion. The recall proportion for this task is described as follows:

Consider document dup_i, belonging to *CollDup* and dup_i is queried on *CollOther*. The similarity measure employed provides the similarity of dup_i with all documents in *CollOther*. Based on the similarity measure, the documents in *CollOther* are ranked in order of relevancy with dup_i. We would like to know at what position in the ranked

documents retrieved form *CollOther* when queried with dup_i does the master document of dup_i (known as mas_i) appear in 100 most relevant documents, or top 100 relevant documents. mas_i may appear anywhere in the top 10, top 20, top 30,……top 100.

On querying *CollOther* with every document in *CollDup* we find the proportion of the documents in *CollDup* which see their master document in the top *n* where $n \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$.

To increase the proportion of documents in *CollDup* that see their master document in the top *n* documents when queried on *CollOther* is the purpose of WOPT as applied to this IR task.

3.3 Category Identification

Figure 3.1 represents the categories that are formed or the document pairs of duplicate-master documents creating the category $dupmas_i$ which will form the basis for the intra-category relationship for document dup_i.

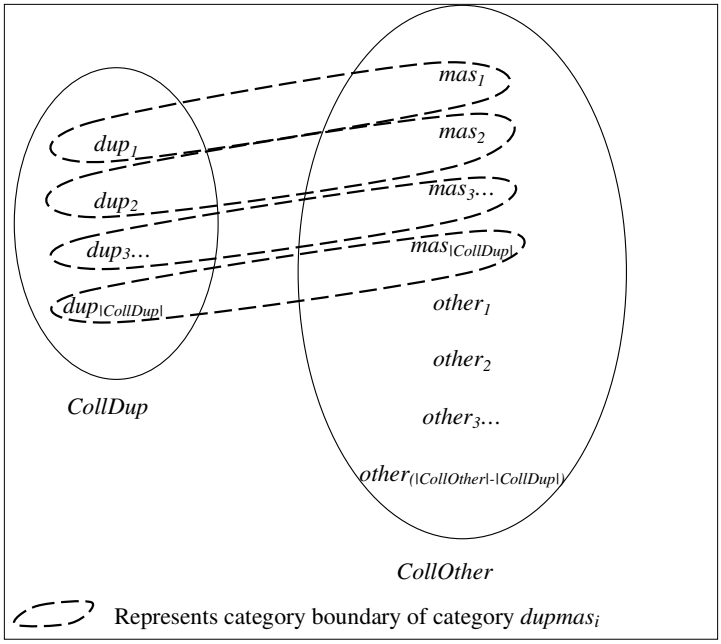

Figure 3.1: Category formed of document pairs of duplicate-master documents.

The groups that are shown in Figure 3.1 are the document pairs of the duplicate document *i* belonging to the *CollDup* collection and its master document belonging to the *CollOther* document collection.

As seen in Figure 3.2 we identify document groups formed of 100 most relevant documents of dup_i from the *CollOther* document collection called $CollOther(dup_i)$. Checking to see the position of mas_i in $CollOther(dup_i)$ for all *i* (all documents in *CollDup*) we obtained recall proportion results for our recall metric. As explained earlier we consider proportion of documents in *CollDup*, which see their master document in the top *n* relevant documents when queried on *CollOther*. The document group formed which is called category $CollOther(dup_i)$ forms the basis of defining the inter-category relationship for document dup_i.

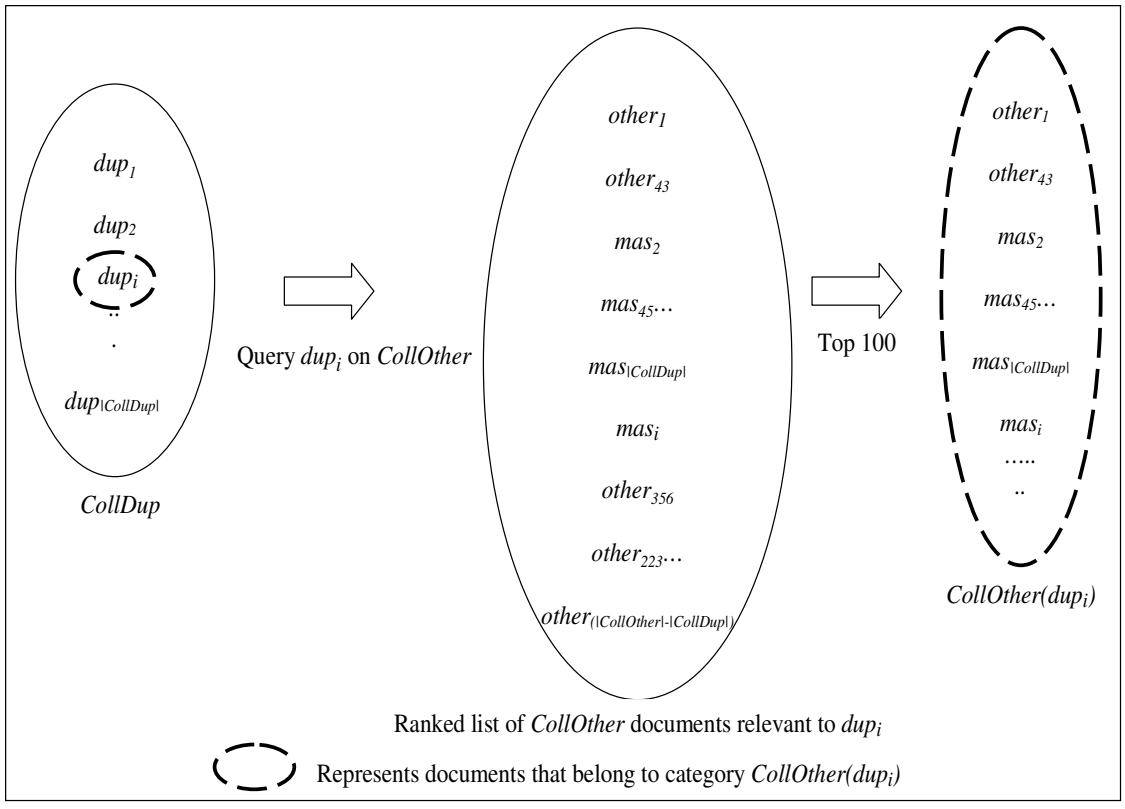

Figure 3.2: Group representation of dup_i and its 100 most relevant documents.

The recall proportions were recorded and will be discussed in the results section of this chapter. The results obtained are from two document-vector representations. One representation of document vectors has binary term weights and the other real valued term weights. The results obtained without the implementation of the WOPT optimization model are called *Pre-WOPT(Bin)* and *Pre-WOPT(TFIDF)*.

The category representations that have been discussed in this chapter will be utilized when we construct the WOPT model which will be described in subsequent sections. The implementation of the WOPT model will yield term weights, which will be used to modify the document-vectors of *CollOther* and *CollDup* and re-call performance values re-determined and compared to the Pre-WOPT implementation. The document category relationships will be defined which will then be utilized in the optimization model, WOPT.

For every document member of the *CollDup* collection we define two categories to which their relationship will be described. The category formed by each *CollDup* member and its master document is defined as the category illustrated by Figure 3.1 and called $dupmas_i$ consisting of duplicate document dup_i and its master document mas_i. The basis of the intra-category measure for dup_i is the category $dupmas_i$.

The category formed by each *CollDup* member and its 100 most relevant documents from *CollOther* collection illustrated in Figure 3.2 is called $CollOther(dup_i)$. $CollOther(dup_i)$ will form the category from which the inter-category relationship of dup_i will be defined.

For every dup_i, belonging to *CollDup*, we have the intra-category centroid vector defined as:

Category C_i*:* $(dupmas_i)$			
Terms→ *Documents*↓	*1*	*j*	$\|J\|=3$
dup_i	$w_{1,1}$	$w_{1,j}$	$w_{1,\|j\|}$
mas_i	$w_{2,1}$	$w_{2,j}$	$w_{2,\|j\|}$
Category centroid $\tilde{C}_i$	$w_{11}+w_{21}$	$w_{1j}+w_{2j}$	$w_{1\|J\|}+w_{2\|J\|}$

Table 3.1 Intra-category centroid

For every duplicate document three associated document vectors that provide information about the various aspects of the document have been defined earlier and will be repeated here with specific application to this task.

Document Information	Representative Vector
Duplicate document	dup
Duplicate document belonging to category $dupmas_i$	dup_i where i indexed on members of the set { $dupmas_i$ } comprising of documents belonging to category $dupmas_i$ and $i = 1..\|\ dupmas_i\ \|$
Duplicate document's term identification vector	$\vec{dup}_i = < t_2\ t_{23}\ t_{142}\ ...\ t_{j_i} >$ where j_i is the set of terms that are present in document dup_i
Duplicate document's term weight vector	$\hat{dup}_i = < w_{i,2}\ w_{i,23}\ w_{i,142}\ ...\ w_{i,j_i} >$ where $w_{i,j_i} \in \{0,1,\Re\}$ are term weights (binary or real valued).
Duplicate document category relationship vector: Intra-category.	$\tilde{dup}_i = < w_{i,2}\ w_{i,23}\ w_{i,142}\ ...\ w_{i,j_i} >$ is the vector quantifying relationship of dup_i with centroid of parent category $dupmas_i$ which is also category C_i.
Document category relationship vector: Inter-category.	$\tilde{dup}_{i,M_i} = < w_{i,2}\ w_{i,23}\ w_{i,142}\ ...\ w_{i,j_i} >$ is the vector quantifying relationship of dup_i with centroid of category M_i which is category $CollOther(dup_i)$.

Table 3.2: Duplicate document representative vectors.

The duplicate document intra-category centroid relationship isdefined as:

$$\tilde{dup}_i = \frac{\tilde{C}_i - \hat{dup}_i}{|C_i| - 1} \cdot \vec{dup}_i,$$

where $\tilde{dup}_i$ is of the form, $< \alpha_1 t_1 \ \alpha_2 t_2 \ \alpha_3 t_3 \ .. \ \alpha_j t_j >$ and the α_j values are the numerical coefficients of the terms, t_j belonging to duplicate document dup_i that indicate the average relationship of the terms t_j to the centroid of category C_i.

For every *CollDup* document dup_i we compute the inter-category relationship by first describing the inter-category as the collection of the hundred most relevant documents in the *CollOther* collection when *CollOther* is queried with duplicate document vector dup_i. In considering these 100 most relevant documents we discard the master document related to dup_i if the master document (mas_i) is retrieved within the 100 most relevant documents from *CollOther.* This category of documents is called the *CollOther(dup$_i$)* or category M_i.

Documents in category *CollOther(dup$_i$)*: M_i $k = 1..\|M_i\|$			
Terms→ *Ranked documents*↓	*1*	*j*	*|J|=3*
$d_{[1]}$	$w_{[1],1}$	$w_{[1,]j}$	$w_{[1],\|j\|}$
$d_{[2]}$	$w_{[2,]1}$	$w_{[2,]j}$	$w_{[2],\|j\|}$
$d_{[3]}$	$w_{[3],1}$	$w_{[3,]j}$	$w_{[3],\|j\|}$
$d_{[k]}$	$w_{[k],1}$	$w_{[k,]j}$	$w_{[k],\|j\|}$
.	.	.	.
$d_{[100]}$	$w_{[100],1}$	$w_{[100],j}$	$w_{[100],\|j\|}$
Category centroid: $\tilde{M}_i$	$\sum_{k\in\{M_i\}} w_{k,1}$	$\sum_{k\in\{M_i\}} w_{k,j}$	$\sum_{k\in\{M_i\}} w_{k,\|J\|}$

Table 3.3: Inter-category centroid.

Table 3.3 defines the construction of the category centroid for category M_i which forms the basis to define the inter-category relationship for document dup_i.

The relationship between d_i and the centroid of M_i is defined as:

$$\tilde{dup}_{i,M_i} = \frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i ,$$

where $\tilde{dup}_{i,M_i}$ is of the form $< \beta_1 t_1 \ \beta_2 t_2 \ \beta_3 t_3 \ .. \ \beta_j t_j >$ and β_j is the category-term-average relationship for term t_j to the term t_j in the centroid vector of category M_i . $|M_i|$ is the size of the category *CollOther(dup$_i$).* We may vary the cardinality of the set *CollOther(dup$_i$)* but for our analysis, we consider 100 an appropriate number of documents.

3.4 WOPT Model Implementation

The optimization model will be discussed for the IR task implementation. As evident from the earlier chapters we try to estimate term weights, by creating document-category relationships and constructing a model to represent these relationships in a way that is conducive to the application of the proposed optimization framework.

The document-term vectors in the *CollDup* collection are represented in the form of their inter- and intra-category relationships. These relationships are then modeled in the form of the WOPT model we propose.

CollDup documents $i = 1..\|CollDup\|$	Intra-Category Relationship with Category($dupmas_i$): C_i	Inter-Category Relationship with category($CollOther(dup_i)$): M_i	Binary Variable for WOPT model
dup_i	$\tilde{dup}_i = \frac{\tilde{C}_i - \hat{dup}_i}{\|C_i\| - 1} \cdot \vec{dup}_i$	$\tilde{dup}_{i,M_i} = \frac{\tilde{M}_i}{\|M_i\|} \cdot \vec{dup}_i$	y_i

Table 3.4 Intra- and Inter-category relationship.

In the earlier sections inter- and intra-category relationships have been defined for each document belonging to the *CollDup* collection. The document-category relationships as shown in Table 3.4 are constructed for all document belonging to the *CollDup* document collection.

As mentioned earlier, $\tilde{dup}_i$ is of the form $< \alpha_1 t_1 \ \alpha_2 t_2 \ \alpha_3 t_3 \ .. \ \alpha_j t_j >$ and $\tilde{dup}_{i,M_i}$ is of the form $< \beta_1 t_1 \ \beta_2 t_2 \ \beta_3 t_3 \ .. \ \beta_j t_j >$ where the α_j and β_j values are the numerical coefficients of the terms t_j in the document dup_i that quantify the term's relation to the centroid of the category considered. The optimization model WOPT estimates the values for $t_j, \forall j \in \{J\}$ which will be employed as term weights or optimal modifiers for the document-term vectors in the documents belonging to the *CollDup* and *CollOther* document collections.

The binary decision variables defined in Table 3.4 above are employed in WOPT to satisfy the logical constraint described as:

$$\tilde{dup}_i = \frac{\tilde{C}_i - \hat{dup}_i}{|C_i| - 1} \cdot \vec{dup}_i \geq \tilde{dup}_{i,M_i} = \frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i$$

which is satisfied if the binary variable y_i is 1 and is not satisfied when y_i is 0. These binary variables are incorporated in the objective function thus creating a pseudo-objective function which when maximized seeks to satisfy as many logical constraints as there are in the WOPT model. It is thus necessary to force the binary decision variables in WOPT to be 1. In order to achieve the validation of the logical constraint we need to construct two model constraints which on being met will validate the logical constraints as constructed above by forcing the associated binary decision variable to 1.The WOPT model is constructed as a Mixed Integer Linear Programming model.

The model constraints which are formed to satisfy a single logical constraint are constructed as:

$$\frac{\tilde{C}_i - \hat{dup}_i}{|C_i| - 1} \cdot \vec{dup}_i - \lambda \left(\frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i \right) \leq y_i, i \in \{1..|CollDup|\}$$

$$\lambda \left(\frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i \right) - \frac{\tilde{C}_i - \hat{dup}_i}{|C_i| - 1} \cdot \vec{dup}_i \leq 1 - y_i, i \in \{1..|CollDup|\}$$

These model constraints are constructed for every document in the *CollDup* collection so that for every duplicate document the logical constraint formed would be to retrieve the associated master document when that particular duplicate document is queried on the *CollOther* collection. λ indicates a tolerance value which can be user controlled to improve the intra-category relationship λ more than the inter-category relationship for the duplicate document concerned.

λ is the WOPT model parameter and is varied empirically to assess the influence on the estimated term weights.

The objective function for the WOPT model is:

$$Maximize\ Z = \sum_{i=1}^{|CollDup|} y_i$$

Trivial solutions are avoided by adding the constraint:

$$\sum_{j \in \{J\}} t_j = \eta,$$

where η is a numerical value which can be chosen based on the scaling factor used in the model. $\{J\}$ is the collection of keywords found across the documents in the *CollDup*

collection or can be considered as the set of terms that are present in the WOPT model. All term weights estimated by WOPT are positive values.

The WOPT model is defined as:

$$Maximize\ Z\ =\ \sum_{i=1}^{|CollDup|} y_i$$

subject to the constraints:

$$\frac{\tilde{C}_i - \hat{dup}_i}{|C_i|-1} \cdot \vec{dup}_i - \lambda\left(\frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i\right) \leq y_i, \forall i \in \{1..|CollDup|\}$$

$$\lambda\left(\frac{\tilde{M}_i}{|M_i|} \cdot \vec{dup}_i\right) - \frac{\tilde{C}_i - \hat{dup}_i}{|C_i|-1} \cdot \vec{dup}_i \leq 1 - y_i, \forall i \in \{1..|CollDup|\}$$

$$\sum_{j \in \{J\}} t_j = \eta$$

where $y_i \in \{0,1\}, \forall i = 1..|CollDup|$ are binary decision variables and $t_j \geq 0, \forall j \in \{J\}$.

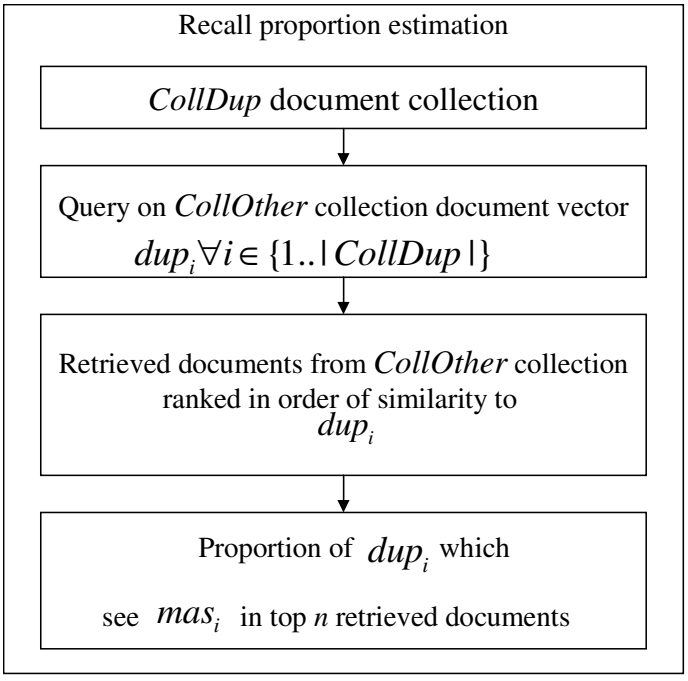

Figure 3.3: Recall proportion estimation

The WOPT estimated term weights are used to modify the term weights in the *CollDup* and *CollOther* document collections after which the recall proportions were estimated again. Figure 3.3 illustrates how the documents from *CollDup* are queried on the documents of *CollOther*. When determining similarity (after obtaining the WOPT estimated weights) between two documents we use the similarity measure defined as:

$$S_{\Re}(d_i, d_k) = \sum_{j \in \{d_i \cap d_k\}} w_j ,$$

where $j \in \{d_i \cap d_k\}$ is the set of common terms when comparing documents d_i and d_k.

Based on this similarity measure between document dup_i and the documents in *CollOther* collection documents in *CollOther* are ranked in order of similarity with document dup_i. It is then determined where in the hundred most relevant (top hundred ranked) documents *CollOther(* dup_i *)* does document mas_i (dup_i's master document) appear as illustrated in Figure 3.3. This procedure is repeated for all documents in *CollDup* and the number of documents in *CollDup* which see their master document in the top *n* where $n = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$ are counted. This information is used to compute the recall proportion after the implementation of WOPT as it was done before WOPT was used to improve recall proportions.

3.5 Pre-WOPT Model Results

For the IR task we consider the results obtained for the recall proportions based on the term weights which have not been modified as a result of the WOPT-IR implementation. The documents that were provided were divided into the *CollDup* and

CollOther collections and were scanned to create the document term vectors where the term weights were of two kinds:

1) Binary term weights where 1 signifies the presence of the term and 0 signifies its absence.
2) Real valued weights which were computed based on the term frequency in the document being scanned and the term frequency across the collection of documents.

The recall proportions were estimated for the IR task for two pre-WOPT model implementations:

1) Document term vectors consisting of binary term weights, Pre-WOPT(Bin).
2) Document term vectors consisting of real-valued term weights, Pre-WOPT(TFIDF).

The method of calculation of the recall proportions has been described in earlier sections. The results obtained for the recall proportions for Pre-WOPT(Bin) and Pre-WOPT(TFIDF) are as follows:

Table 3.5 below represents the recall proportions for the Pre-WOPT(Bin) model

Query Size, n	*Recall percentage as a proportion of CollDup documents*
10	*21.51*
20	*25.81*
30	*28.65*
40	*31.03*
50	*33.49*
60	*35.02*
70	*36.64*
80	*37.79*
90	*39.17*
100	*39.86*

Table 3.5: Recall proportion for Pre-WOPT(Bin)

Table 3.4 represents the percentage of *CollDup* documents when queried on *CollOther* see their master document in the retrieved document set of size *n*.

The real-valued term weights were computed using the TFIDF term weighting scheme. It was observed that the frequency of occurrence of keywords in the documents was to a great extent, unity. The terms in the keyword list were highly specialized terms selected by the company to form the feature set, and it was not suprising to see that the term frequency was one, for a majority of terms in a large proportion of the documents.

The Document Frequency is the frequency of appearance of the term across the documents in the *CollDup* and *CollOther* collection combined. DF_{t_j} is the document frequency for term t_j, where $j \in \{J\}$. DF_{t_j} is the number of documents in the combined collection of *CollDup* and *CollOther* that contain term t_j.The document frequency weight for each term is then computed as:

$$DFW_{t_j} = \log \frac{|J|}{DF_{t_j}},$$

where $|J|$ is the size of the keyword list. To implement Pre-WOPT(TFIDF) we multiply the binary term weights of the document vectors in the IR collection with the DFW_{t_j} term weights computed as shown above. The document-term vectors now have term weights which are real valued and reflect the Term Frequency and the Inverse Document Frequency of the terms.

The recall proportions are estimated once again using the modified document-term vectors. The similarity measure employed is the vector dot product. The recall proportion values are the percentage of *CollDup* documents which see their master document when queried on the *CollOther* document collection. The recall percentage

values are the proportion of *Duplicate* documents from the *CollDup* document collection for which their associated *Master* document is retrieved in the n retrieved documents when queried on the *CollOther* collection.

The results for the recall proportions for the Pre-WOPT(TFIDF) model is:

Query Size *n*	Recall percentage as a proportion of *CollDup* documents
10	29.80
20	38.02
30	42.70
40	46.08
50	48.31
60	50.31
70	52.15
80	53.30
90	54.53
100	55.38

Table 3.6 Recall proportion for Pre-WOPT(TFIDF)

Figure 3.4 illustrates the increase in recall proportions for different query sizes when comparing Pre-WOPT(Bin) and Pre-WOPT(TFIDF). The Pre-WOPT(TIDF) model performs better since the term weights quantify the collection wide importance of the terms in the document. The binary term weights do not capture the term's importance in the document collection. For the term's importance in the individual document the binary term weight considers it to be as important as the other terms in the document vector.

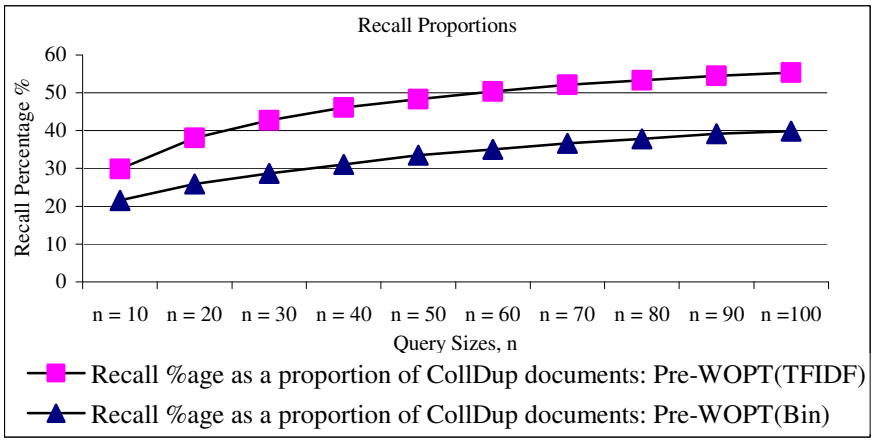

Figure 3.4: Graphical comparison of Pre-WOPT(Bin) and Pre-WOPT(TFIDF)

3.6 WOPT Model Results

The WOPT model is implemented using the document vectors with binary term weights. As mentioned in the earlier chapter the object-category relationships are established in terms of inter- and intra-category relationships. These relationships are then modeled in the form the WOPT framework. The WOPT model is solved to optimality and the term weights are estimated. These term weights are then used as optimal modifiers to modify binary term weight for the document-term vectors in the document collection. This modifies the document-term vectors which have binary term weights and yields real-valued term weights. Consequently the recall proportions are determined. It is to be noted, that the IR WOPT model incorporates a model parameter λ whose value is varied to yield term weights which are different for different values of λ. For every set of term weights estimated, the recall proportions are determined.

Query Size, *n*	Recall percentage as a proportion of *CollDup* documents							
	$\lambda = 1$	$\lambda = 2$	$\lambda = 3$	$\lambda = 4$	$\lambda = 5$	$\lambda = 6$	$\lambda = 7$	$\lambda = 10$
10	15.59	25.57	28.73	31.87	25.26	25.96	33.94	36.64
20	18.66	33.10	37.25	39.63	32.79	33.10	41.16	43.63
30	21.35	37.27	40.11	45.36	38.45	39.12	46.29	46.72
40	25.72	40.47	45.85	49.00	44.01	44.16	50.38	51.46
50	28.26	44.32	49.92	52.15	48.23	48.23	53.45	54.30
60	30.41	47.31	51.69	53.84	50.92	51.38	55.45	55.68
70	32.56	49.23	54.38	55.45	53.69	54.14	57.22	57.22
80	34.63	51.61	56.83	56.45	55.37	55.91	58.98	58.37
90	37.25	53.92	58.75	57.68	56.83	56.91	59.98	59.61
100	39.17	55.99	60.06	59.06	57.83	58.29	60.82	60.37

Table 3.7: WOPT model results for different model parameter values.

Table 3.7 presents the recall proportions for different query sizes and for different model parameters λ. Figure 3.5 provides a graphical comparison of the recall proportions when the WOPT model is solved with different values of the model parameter resulting in different values for the optimal term weights estimated.

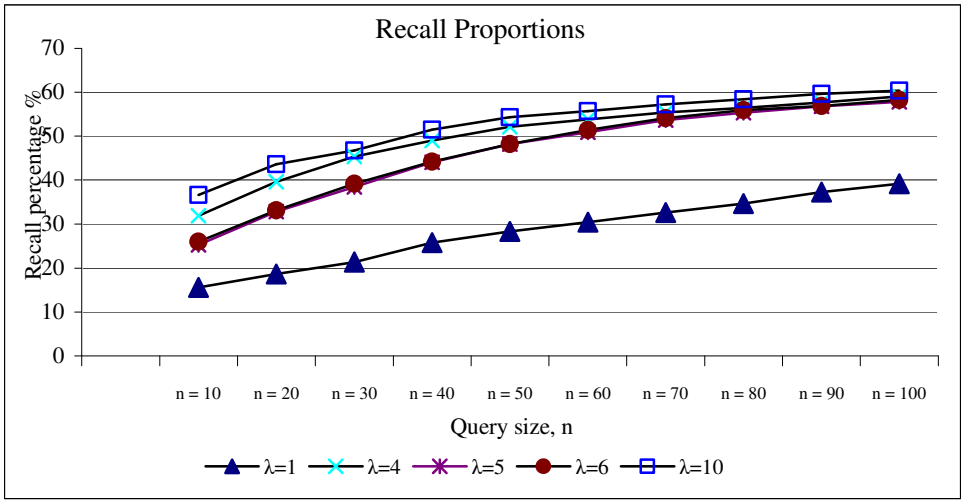

Figure 3.5: Graphical comparison of WOPT results for different parameter values

The WOPT model estimates term weights which when used to modify term weights in the document-term vectors (binary term weights) across *CollDup* and *CollOther* yields a higher recall proportion as illustrated by Figure 3.5.

Figure 3.6 illustrates the improvement in recall proportions observed when WOPT is implemented to obtain global term weights/optimal modifiers compared to when WOPT is not implemented to modify the binary document-term vectors.

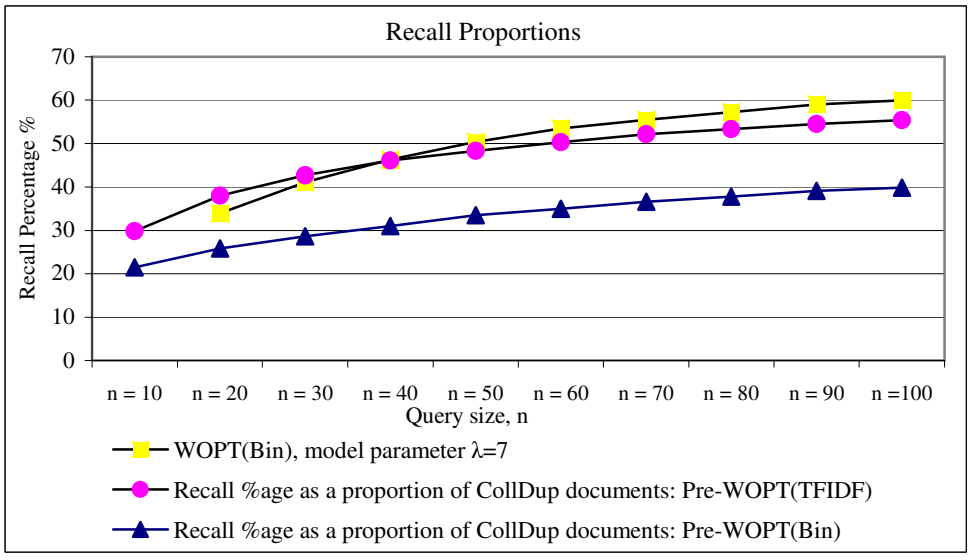

Figure 3.6: Graphical comparison of Pre-WOPT and WOPT results

The results indicate an improvement in the percentage of documents from the *CollDup* collection for which the respective master document is retrieved in the 100 most relevant documents from *CollOther* when that particular duplicate document is queried on *CollOther*.

It is important to retrieve the master document since the master document addresses similar user issues as the associated duplicate document does. This type of information is used in creating categories of documents and improving document-category membership by estimating optimal term weights using the proposed WOPT framework. This lends an intelligent retrieval mechanism to the Information Retrieval system designed because the framework allows for creation of document groups which may have similarities that may not be apparent quantitatively but such similarities can be created by the user based on empirical and qualitative conditions. Groups are also formed based on the similarity of documents with the query and as we have seen above, the group of documents most relevant to the duplicate document is influenced by the similarity measure employed. The proposed framework can be used to model document-category relationships by employing other similarity measure functions as well.

The determination of the category centroid may also be based upon other considerations such as centroid vectors whose components possess only those terms that are found in 90% of the documents belonging to that category.

The proposed framework in its implementation in this IR task shows the possibility of creating empirically influenced document groups and preserving the membership of the documents to the choice of group as decided by the user of the intended IR system.

Chapter 4: Text Categorization Models and Applications

For the Text Categorization task (TC) we consider the Reuters Corpus Volume (RCV1-v2) where 'v2' indicated the corrected data. The RCV1-v2 is an archive of over 800,000 manually categorized newswire stories made available by Reuters Ltd., for research purposes. The documents we have made use of are available in the document vector format as distributed by Lewis et al, 2004 with TFIDF weights pre-computed [Salton and Buckley, 1998].

4.1 Preliminary Approach

The documents are distributed in the form of document collections divided into documents sets which can be used for training and testing purposes. This division of documents has a chronological basis and we have preserved this division when choosing documents for our training and test purposes. The training collection called ' *train'* and the ' *test-0'* test collection were used for our analysis. We consider only those documents in the training and test collections which have only one category label as illustrated by Figure 4.1 so as to learn the document-category membership to a greater degree than from those documents that belong to multiple categories concurrently.

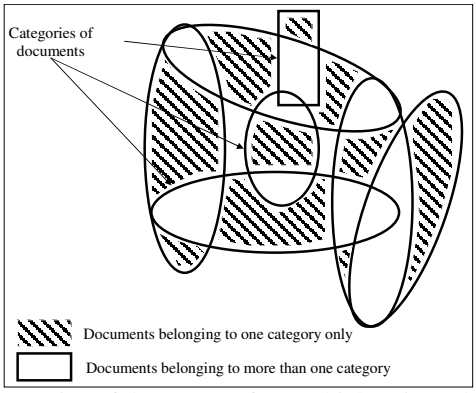

Figure 4.1: Categories of documents from which unique documents are preserved.

The category labels considered were the categories that represented the document's subject or theme. The categories considered were the '*Topics'* categories and are 103 in number. The topics were coded and captured the major subjects of a news story. They were organized in four hierarchical groups:

1) CCAT (Corporate/Industrial),
2) ECAT (Economics),
3) GCAT (Government/Social)
4) MCAT (Markets).

Each group of categories is further divided and sub-divided to form more specific topic labels pertaining to the broad category as illustrated in Figure 4.2. It was decided to prune the document collections (training and test collections) to include only those documents that had only one topic label. In order to make implementation of the proposed framework with the RCV1-v2 data simpler in terms of scalability we considered those labels which contained approximately 250 documents to create the training set. In the training set we thus had a total of 999 documents across the four chosen topics labels:

1) C24: Capacity/Facilities
2) C33: Contracts/Orders
3) C41: Management
4) E51: Trade/Reserves

The hierarchical layout of the topic category labels is not of much importance in our analysis since we have considered topic labels independently of their parent category or the categories into which they can be further divided.

Categories from GCAT and MCAT are not chosen because their sub-categories either contained a large number of documents (> 250) or a very small number of (<50) when the document groups were pruned.

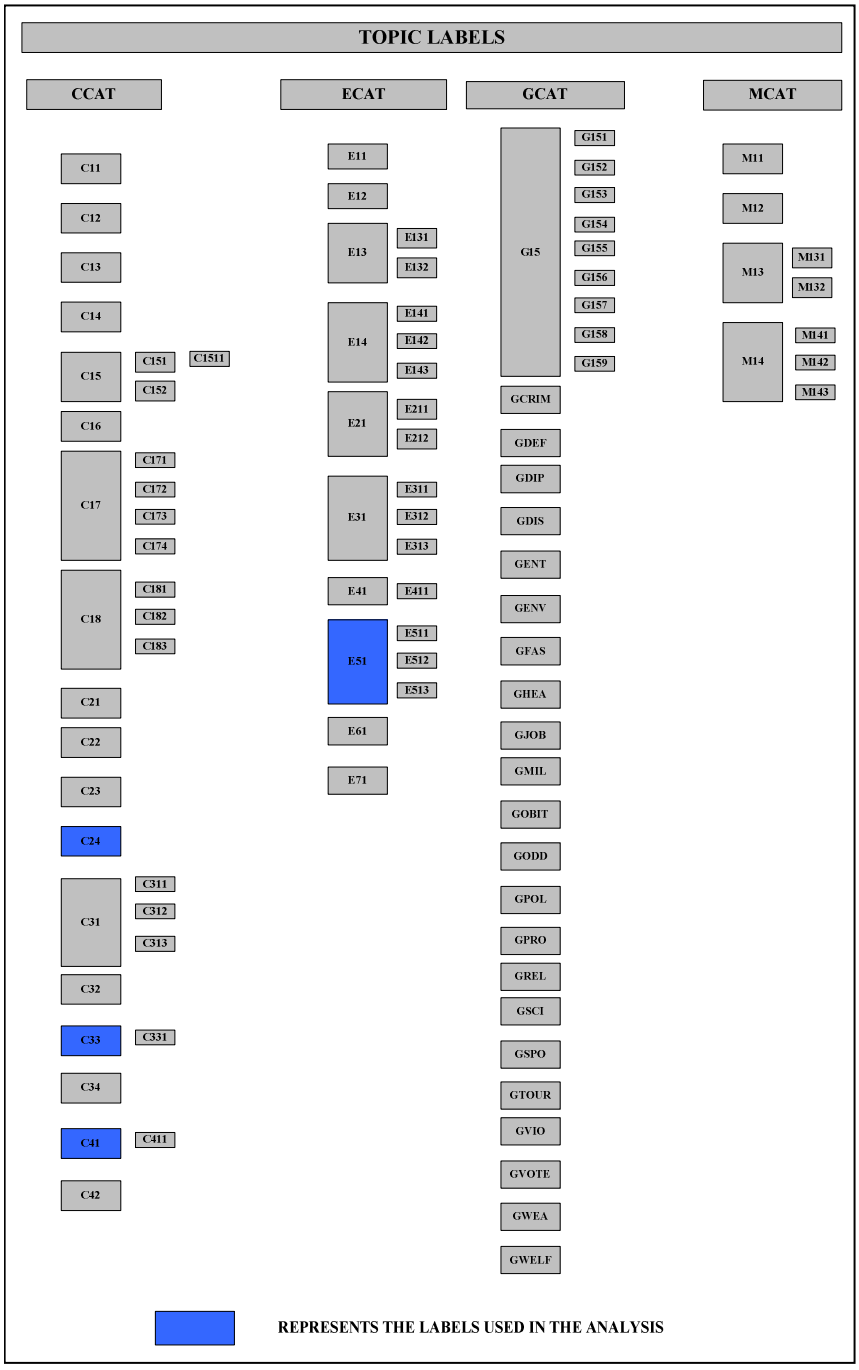

Figure 4.2: Topic category label codes and their hierarchy.

The classifier is constructed for every document category considered in the analysis. The classifier for every category is the classifiers representative vector or the centroid vector as defined in Table 4.1.

Documents in category $C_c, \forall i \in \{C_c\}$			
Terms→ *Documents*↓	*1*	*j*	|J|=3
d_1	$w_{1,1}$	$w_{1,j}$	$w_{1,\|J\|}$
d_2	$w_{2,1}$	$w_{2,j}$	$w_{2,\|J\|}$
d_3	$w_{3,1}$	$w_{3,j}$	$w_{3,\|J\|}$
d_4	$w_{4,1}$	$w_{4,j}$	$w_{4,\|J\|}$
.	.	.	.
d_i	$w_{i,1}$	$w_{i,j}$	$w_{i,\|J\|}$
Category classifier: $\tilde{C}_c$	$\sum_{\forall i \in \{C_c\}} w_{i,1}$	$\sum_{\forall i \in \{C_c\}} w_{i,j}$	$\sum_{\forall i \in \{C_c\}} w_{i,\|J\|}$

Table 4.1: Category classifier definition

The classifier is constructed for every category and is used to categorize documents from the test set.

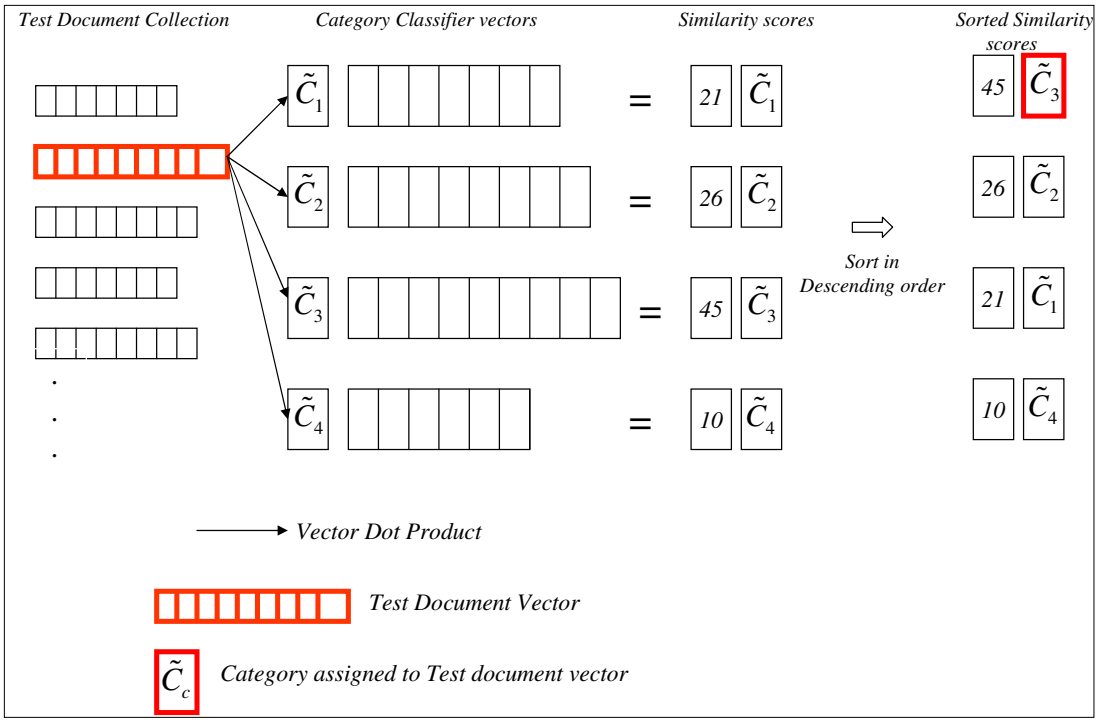

Figure 4.3: Classification of document vectors from test collection

As shown in Figure 4.3 the category classifiers are derived from the category documents in the training collection. Each classifier as constructed in table 4.1 can be constructed from document vectors that have binary, TFIDF or *ConfWeight* term weights. The RCV1-v2 distribution consists of TFIDF [Salton and Buckley, 1998] weighted vectors. We also consider binary and *ConfWeight* [Soucy and Mineau, 2005] term weighted vectors for our analysis.

Subsequently results will be compared for classification accuracy based on the classification method shown in Figure 4.3. The classification accuracy is measured for each of the four categories. From the test document collection the classification accuracy for category C_a as an example is defined as:

$$\frac{Test\ set\ documents\ classified\ as\ C_a}{Total\ number\ of\ \ test\ set\ documents\ truly\ belonging\ to\ C_a}.$$

4.2 Document Category Relationships

Every document in the training set is assigned a representative vector to indicate document information on the individual document level and the relationship the document shares with the centroid of its parent category and with the centroid of other non-parent categories. Table 4.2 lists the vectors that will be defined and constructed for every document in the training set such that it will be possible to make logical comparisons between the document's relationship with the centroid of its parent category and the centroid of its non-parent category. These logical comparisons give rise to WOPT model constraints to estimate optimal term weights which preserve document-category memberships to improve classification accuracy results when classifying documents belonging to the test set.

Document Information	Representative Vector
Document	d
Document belonging to category C_1	d_i where i is indexed on members of the set $\{ C_1 \}$ comprising of documents belonging to category C_1 and $i = 1..\| C_1 \|$
Document's term identification vector	$\vec{d}_i = < t_2 \ t_{23} \ t_{142} \ ... \ t_{j_i} >$ where j_i is the set of terms that are present in document dup_i
Document's term weight vector	$\hat{d}_i = < w_{i,2} \ w_{i,23} \ w_{i,142} \ ... \ w_{i,j_i} >$ where $w_{i,j_i} \in \{0,1,\Re\}$ are term weights (binary or real valued).
Document category relationship vector: Intra-category.	$\tilde{d}_i = < w_{i,2} \ w_{i,23} \ w_{i,142} \ ... \ w_{i,j_i} >$ is the vector quantifying relationship of d_i with centroid of its parent category C_1.
Document category relationship vector: Inter-category.	$\tilde{d}_{i,C_2} = < w_{i,2} \ w_{i,23} \ w_{i,142} \ ... \ w_{i,j_i} >$ is the vector quantifying relationship of d_i with centroid of non-parent category C_2.

Table 4.2: Representative vectors for each document in the training set.

In defining the intra- and inter-category relationship for each document in the training set it is important to define the category centroid which will form the basis for the document-category relationship. A category's centroid is defined as:

Documents in category $C_1, \forall i \in \{C_1\}$			
Terms→ Documents↓	*1*	*j*	*|J|=3*
d_1	w_{11}	w_{1j}	$w_{1\|J\|}$
d_2	w_{21}	w_{2j}	$w_{2\|J\|}$
d_3	w_{31}	w_{3j}	$w_{3\|J\|}$
d_k	w_{41}	w_{4j}	$w_{4\|J\|}$
.	.	.	.
d_i	w_{i1}	w_{ij}	$w_{i\|J\|}$
C_1 *category centroid,* $\tilde{C}_1$	$\sum_{\forall i \in \{C_1\}} w_{i,1}$	$\sum_{\forall i \in \{C_1\}} w_{i,j}$	$\sum_{\forall i \in \{C_1\}} w_{i,\|J\|}$

Table 4.3: Category centroid definition

When a document belonging to category C_1 (for example document d_i) is represented in terms of a document-category relationship with the centroid of its parent category (C_1 in this case) we obtain the document intra-category relationship defined as:

$$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i,$$

where $\tilde{d}_i$ is of the form, $< \alpha_1 t_1 \ \alpha_2 t_2 \ \alpha_3 t_3 \ .. \ \alpha_j t_j >$ and the α_j values are the numerical coefficients of the terms t_j in the document $d_i \in C_1$ that quantifies the relationship of the terms in document $d_i \in C_1$ with the centroid of category C_1.

Since we have considered four categories for the categorization task each document is compared to the centroid of its other three non-parent categories. That is to say that every document can be described as a vector whose components quantify the term's relationship with the centroid of the non-parent category of the document. When defining the vector which compares the document to the centroid of another non-parent category, we obtain the document's inter-category relationship. For example, document $d_i \in C_1$ is compared to the centroid of non-parent category C_2 which results in the inter-category relationship as defined as:

$$\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i,$$

where $\tilde{d}_{i,C_2}$ is of the form $< \beta_1 t_1 \ \beta_2 t_2 \ \beta_3 t_3 \ .. \ \beta_j t_j >$ and the β_j values quantifies the terms relationship to the centroid of category C_2. In the TC task, every object is thus represented by its relationship to the centroid of its parent category and to the centroids of its non-parent categories.

These relationships are utilized in the optimization model to estimate the values of t_j, which are the global term weights/optimal modifiers to modify term weights of test document vectors to obtain improved classification rates.

4.3 WOPT Model Implementation

In the TC task, we consider the intra-category and inter-category relationship for every document across the four categories considered in the training set. The document-category relationships have been defined and constructed earlier and will now be represented in terms of logical constraints to be employed in the construction of the WOPT optimization model. Considering the four categories in the training sample, as C_1, C_2, C_3, C_4, we show the object-category relationships for objects in all four categories.

Documents belonging to C_1 $i = 1..\|C_1\|$ $d_i \in \{C_1\}$	Intra-category relationship: parent category, C_1	Inter-category relationship: $C_1' = \{C_2, C_3, C_4\}$ $\tilde{d}_{i,C_1'} = \frac{\tilde{C}_1'}{\|C_1'\|} \cdot \vec{d}_i$		Binary variable for WOPT model
d_i	$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\|C_1\| - 1} \cdot \vec{d}_i$	C_2	$\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{\|C_2\|} \cdot \vec{d}_i$	$y_{i,2}$
		C_3	$\tilde{d}_{i,C_3} = \frac{\tilde{C}_3}{\|C_3\|} \cdot \vec{d}_i$	$y_{i,3}$
		C_4	$\tilde{d}_{i,C_4} = \frac{\tilde{C}_4}{\|C_4\|} \cdot \vec{d}_i$	$y_{i,4}$

Table 4.4: C_1 objects intra- and inter-category relationships.

Logical constraints are created for the documents belonging to category C_1 by comparing the intra- and inter-category relationships defined above. The logical constraint that are created indicate the preservation of the document-category memberships by estimating term weights which will make documents similar to the centroid of their parent category and make the documents dissimilar to the centroids of other non-parent categories. Such a logical constraint for document belonging to category C_1 and compared to the centroid of document C_2 is defined as:

$$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i \;\geq\; \tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i$$

Table 4.3 also shows the binary variables that are used in the WOPT model framework. These binary variables are used to force the above logical constraint to be valid in the WOPT model. The binary variable assumes a value 1 when the above logical constraint is validated by the optimization model and assumes a value 0 when the WOPT model is unable to validate the logical constraint defined above. The above logical constraint must be modeled to suit a mathematical programming framework such as the one proposed, WOPT. To obtain model constraints derived to reflect the logical constraint, each logical constraint is divided into two model constraints which employ the binary variable and the model parameter λ. These model constraints are constructed to create a Mixed Integer Linear Programming model which will utilize the binary decision variables mentioned in Table 4.3 to create an objective function which when maximized will seek to assign value 1 to as many binary decision variables as possible thus validating as many logical constraints as allowed by the estimated term weights.

Thus for every document category logical constraint which compare document category similarity two model constraints are derived. For example the logical constraint $\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1} \cdot \vec{d}_i \geq \tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i$ will give rise to the two WOPT model constraint defined as:

$$\left(\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1} \cdot \vec{d}_i\right) - \left[\lambda + \left(\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}\right)\right] \leq y_{i,2}$$

$$\left[\lambda + \left(\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}\right)\right] - \left(\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1} \cdot \vec{d}_i\right) \leq 1 - y_{i,2}$$

These WOPT model constraints are constructed to force the associated binary variable $y_{i,2}$ to assume the value 1 which would result in the satisfaction of the inequality represented by $\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1} \cdot \vec{d}_i \geq \tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i$. The model parameter λ is a numerical quantity which can be controlled by the user based on the model estimated term weights and the resultant classification accuracy. We construct logical constraints for all documents in the training set and compare each documents intra-category relationship with the inter-category relationship of the document to the centroids of all the other non-parent categories in the training set.

For example the intra-category relationship of each document in category C_1 will be compared to the centroids of other non-parent categories in the form of the document's inter-category relationship. The comparisons shown in Table 4.4 are in the form of logical inequalities that need to be satisfied by modeling these inequalities in the form of WOPT model constraints with the associated binary decision variables in the WOPT optimization model.

The logical inequalities are constructed to provide a description of the similarity measure which computes the quantitative similarity value between document vectors. These logical constraints quantify the similarity and when the inequality that they describe is satisfied within the model framework the term weights that satisfy the constraint will result in the similarity measure values indicating the increase in similarity between earlier dissimilar documents.

$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{\lvert C_2 \rvert} \cdot \vec{d}_i \right)$
$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_3} = \frac{\tilde{C}_3}{\lvert C_3 \rvert} \cdot \vec{d}_i \right)$
$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_4} = \frac{\tilde{C}_4}{\lvert C_4 \rvert} \cdot \vec{d}_i \right)$

Table 4.5: Logical constraints for $d_i \in C_1, \forall i \in \{1,..,\lvert C_1 \rvert\}$

Logical constraints	WOPT model constraints
$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_2} = \frac{\tilde{C}_2}{\lvert C_2 \rvert} \cdot \vec{d}_i \right)$	$\frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i - \left[\lambda + \left(\frac{\tilde{C}_2}{\lvert C_2 \rvert} \cdot \vec{d}_i \right) \right] \leq y_{i,2}$ $\left[\lambda + \left(\frac{\tilde{C}_2}{\lvert C_2 \rvert} \cdot \vec{d}_i \right) \right] - \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \leq 1 - y_{i,2}$
$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_3} = \frac{\tilde{C}_3}{\lvert C_3 \rvert} \cdot \vec{d}_i \right)$	$\frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i - \left[\lambda + \left(\frac{\tilde{C}_3}{\lvert C_3 \rvert} \cdot \vec{d}_i \right) \right] \leq y_{i,3}$ $\left[\lambda + \left(\frac{\tilde{C}_3}{\lvert C_3 \rvert} \cdot \vec{d}_i \right) \right] - \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \leq 1 - y_{i,3}$
$\tilde{d}_i = \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \;\geq\; \lambda + \left(\tilde{d}_{i,C_4} = \frac{\tilde{C}_4}{\lvert C_4 \rvert} \cdot \vec{d}_i \right)$	$\frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i - \left[\lambda + \left(\frac{\tilde{C}_4}{\lvert C_4 \rvert} \cdot \vec{d}_i \right) \right] \leq y_{i,4}$ $\left[\lambda + \left(\frac{\tilde{C}_4}{\lvert C_4 \rvert} \cdot \vec{d}_i \right) \right] - \frac{\tilde{C}_1 - \hat{d}_i}{\lvert C_1 \rvert - 1} \cdot \vec{d}_i \leq 1 - y_{i,4}$

Table 4.6: WOPT Model constraints from Logical constraints

As described earlier each logical constraint must be modeled in the form of WOPT model constraints. We explicitly describe the WOPT model constraints derived from the logical constraints mentioned in Table 4.4 in Table 4.5. As shown for documents belonging to C_1 the representative vectors are constructed and defined for documents belonging to category C_2 and C_3 as shown in Table 4.7 and Table 4.8.

Documents belonging to C_2 $k = 1..\|C_2\|$ $d_k \in \{C_2\}$	Intra-category relationship: parent category, C_2	Inter-category relationship: C_2' non-parent categories $C_2' = \{C_1, C_3, C_4\}$ $\tilde{d}_{k,C_2'} = \frac{\tilde{C}_2'}{\|C_2'\|} \cdot \vec{d}_k$		Binary variable for WOPT model
d_k	$\tilde{d}_k = \frac{\tilde{C}_2 - \hat{d}_k}{\|C_2\| - 1} \cdot \vec{d}_k$	C_1	$\tilde{d}_{k,C_1} = \frac{\tilde{C}_1}{\|C_1\|} \cdot \vec{d}_k$	$y_{k,1}$
		C_3	$\tilde{d}_{k,C_3} = \frac{\tilde{C}_3}{\|C_3\|} \cdot \vec{d}_k$	$y_{k,3}$
		C_4	$\tilde{d}_{k,C_4} = \frac{\tilde{C}_4}{\|C_4\|} \cdot \vec{d}_k$	$y_{k,4}$

Table 4.7: C_2 document intra- and inter-category relationships.

Documents belonging to C_3 $l = 1..\|C_3\|$ $d_l \in \{C_3\}$	Intra-Category Relationship: Parent category, C_3	Inter-Category Relationship: C_3' $C_3' = \{C_1, C_2, C_4\}$ $\tilde{d}_{l,C_3'} = \frac{\tilde{C}_3'}{\|C_3'\|} \cdot \vec{d}_l$		Binary variable for WOPT model
d_l	$\tilde{d}_l = \frac{\tilde{C}_3 - \hat{d}_l}{\|C_3\| - 1} \cdot \vec{d}_l$	C_1	$\tilde{d}_{l,C_1} = \frac{\tilde{C}_1}{\|C_1\|} \cdot \vec{d}_l$	$y_{l,1}$
		C_2	$\tilde{d}_{l,C_2} = \frac{\tilde{C}_2}{\|C_2\|} \cdot \vec{d}_l$	$y_{l,2}$
		C_4	$\tilde{d}_{l,C_4} = \frac{\tilde{C}_4}{\|C_4\|} \cdot \vec{d}_l$	$y_{l,4}$

Table 4.8: C_3 document intra- and inter-category relationships.

Table 4.9 defines the representative vectors for all documents in the training set belonging to category C_4.

Documents belonging to C_4 $p = 1..\|C_4\|$ $d_p \in \{C_4\}$	Intra-Category Relationship: Parent category, C_4	Inter-Category Relationship: C_4' $C_4' = \{C_1, C_2, C_3\}$ $\tilde{d}_{p,C_4'} = \frac{\tilde{C}_4'}{\|C_4'\|} \cdot \vec{d}_p$		Binary variable for WOPT model
d_p	$\tilde{d}_p = \frac{\tilde{C}_4 - \hat{d}_p}{\|C_4\| - 1} \cdot \vec{d}_p$	C_1	$\tilde{d}_{p,C_1} = \frac{\tilde{C}_1}{\|C_1\|} \cdot \vec{d}_p$	$y_{p,1}$
		C_2	$\tilde{d}_{p,C_2} = \frac{\tilde{C}_2}{\|C_2\|} \cdot \vec{d}_p$	$y_{p,2}$
		C_3	$\tilde{d}_{p,C_4} = \frac{\tilde{C}_4}{\|C_4\|} \cdot \vec{d}_p$	$y_{p,3}$

Table 4.9: C_4 document intra- and inter-category relationships.

The objective function is constructed to reflect that it is beneficial to have as many binary decision variables assuming a value 1 as there are in the WOPT model. The objective function along with the WOPT model constraints for all documents across all the categories in the training set is as follows.

$$\begin{aligned} \textit{Maximize } Z \;=\; & \sum_{i=1}^{|C_1|} y_{i,2} + \sum_{i=1}^{|C_1|} y_{i,3} + \sum_{i=1}^{|C_1|} y_{i,4} + \sum_{k=1}^{|C_2|} y_{k,1} + \sum_{k=1}^{|C_2|} y_{k,3} + \sum_{k=1}^{|C_2|} y_{k,4} \\ & + \sum_{l=1}^{|C_3|} y_{l,1} + \sum_{l=1}^{|C_3|} y_{l,2} + \sum_{l=1}^{|C_3|} y_{l,4} + \sum_{p=1}^{|C_4|} y_{p,1} + \sum_{p=1}^{|C_4|} y_{p,2} + \sum_{p=1}^{|C_4|} y_{p,3} \end{aligned}$$

subject to the constraints:

[Constraints for documents $d_i \in \{C_1\}, \forall i \in \{1,..,|C_1|\}$]

$$\frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i \; - \left[\lambda + \left(\frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i \right) \right] \leq y_{i,2}$$

$$\left[\lambda + \left(\frac{\tilde{C}_2}{|C_2|} \cdot \vec{d}_i \right) \right] \; - \frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i \; \leq 1 - y_{i,2}$$

$$\frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i \; - \left[\lambda + \left(\frac{\tilde{C}_3}{|C_3|} \cdot \vec{d}_i \right) \right] \leq y_{i,3}$$

$$\left[\lambda + \left(\frac{\tilde{C}_3}{|C_3|} \cdot \vec{d}_i \right) \right] \; - \frac{\tilde{C}_1 - \hat{d}_i}{|C_1| - 1} \cdot \vec{d}_i \; \leq 1 - y_{i,3}$$

$$\frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1}\cdot\vec{d}_i - \left[\lambda + \left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_i\right)\right] \le y_{i,4}$$

$$\left[\lambda + \left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_i\right)\right] - \frac{\tilde{C}_1 - \hat{d}_i}{|C_1|-1}\cdot\vec{d}_i \le 1 - y_{i,4}$$

[Constraints for documents $d_k \in \{C_2\}, \forall k \in \{1,..,|C_2|\}$]

$$\frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k - \left[\lambda + \left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_k\right)\right] \le y_{k,1}$$

$$\left[\lambda + \left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_k\right)\right] - \frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k \le 1 - y_{k,1}$$

$$\frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k - \left[\lambda + \left(\frac{\tilde{C}_3}{|C_3|}\cdot\vec{d}_k\right)\right] \le y_{k,3}$$

$$\left[\lambda + \left(\frac{\tilde{C}_3}{|C_3|}\cdot\vec{d}_k\right)\right] - \frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k \le 1 - y_{k,3}$$

$$\frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k - \left[\lambda + \left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_k\right)\right] \le y_{k,4}$$

$$\left[\lambda + \left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_k\right)\right] - \frac{\tilde{C}_2 - \hat{d}_k}{|C_2|-1}\cdot\vec{d}_k \le 1 - y_{k,4}$$

[Constraints for documents $d_l \in \{C_3\}, \forall l \in \{1,..,|C_3|\}$]

$$\frac{\tilde{C}_3 - \hat{d}_l}{|C_3|-1}\cdot\vec{d}_l - \left[\lambda + \left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_l\right)\right] \le y_{l,1}$$

$$\left[\lambda + \left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_l\right)\right] - \frac{\tilde{C}_3 - \hat{d}_l}{|C_3|-1}\cdot\vec{d}_l \le 1 - y_{l,1}$$

$$\frac{\tilde{C}_3 - \hat{d}_l}{|C_3|-1}\cdot\vec{d}_l - \left[\lambda + \left(\frac{\tilde{C}_2}{|C_2|}\cdot\vec{d}_l\right)\right] \le y_{l,2}$$

$$\left[\lambda + \left(\frac{\tilde{C}_2}{|C_2|}\cdot\vec{d}_l\right)\right] - \frac{\tilde{C}_3 - \hat{d}_l}{|C_3|-1}\cdot\vec{d}_l \le 1 - y_{l,2}$$

$$\frac{\tilde{C}_3 - \hat{d}_l}{|C_3|-1}\cdot\vec{d}_l - \left[\lambda + \left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_l\right)\right] \le y_{l,4}$$

$$\left[\lambda+\left(\frac{\tilde{C}_4}{|C_4|}\cdot\vec{d}_l\right)\right]-\frac{\tilde{C}_3-\hat{d}_l}{|C_3|-1}\cdot\vec{d}_l \le 1-y_{l,4}$$

[Constraints for document $d_p \in \{C_4\}, \forall p \in \{1,..,|C_4|\}$]

$$\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p -\left[\lambda+\left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_p\right)\right] \le y_{p,1}$$

$$\left[\lambda+\left(\frac{\tilde{C}_1}{|C_1|}\cdot\vec{d}_p\right)\right]-\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p \le 1-y_{p,1}$$

$$\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p -\left[\lambda+\left(\frac{\tilde{C}_2}{|C_2|}\cdot\vec{d}_p\right)\right] \le y_{p,2}$$

$$\left[\lambda+\left(\frac{\tilde{C}_2}{|C_2|}\cdot\vec{d}_p\right)\right]-\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p \le 1-y_{p,2}$$

$$\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p -\left[\lambda+\left(\frac{\tilde{C}_3}{|C_3|}\cdot\vec{d}_p\right)\right] \le y_{p,3}$$

$$\left[\lambda+\left(\frac{\tilde{C}_3}{|C_3|}\cdot\vec{d}_p\right)\right] -\frac{\tilde{C}_4-\hat{d}_p}{|C_4|-1}\cdot\vec{d}_p \le 1-y_{p,3}$$

The solution variables are $t_j \in \{J\}, \forall j \in \{1,..,|J|\}$ and $t_j \ge 0, \forall j$. The importance of the binary decision variables in the WOPT model has been mentioned earlier. The binary decision variables are defined as: $y_{a,b} \in \{0,1\}$,where $a \in \{i,k,l,p\}$ and indicates the index on the documents in categories C_1, C_2, C_3 and C_4. In the binary decision variable $y_{a,b} \in \{0,1\}$ b indicates the category (inter-category relationship) to which the intra-category relationship of the document indicated by *a* is being compared to.

When the WOPT model as described above is solved, trivial solutions may result which can be avoided by adding the constraint defined as:

$$\sum_{j\in\{J\}} t_j = \eta,$$

where η is a numerical quantity which is user defined and influences the magnitude of the WOPT estimated term weights thus affecting the classification accuracy of test documents during classification. It is evident that the solution variables are the terms that are present in the documents comprising the training set and form the keyword list indicated by the set of term $\{J\}$.

When classification of documents from the test set is carried out the category classifiers defined earlier are used. These classifiers are the centroid vectors of each category formed from the training documents in that category. The global term weights estimated by WOPT are used to modify the term weights of the document term vectors in the test set. Once these test document-term vector are modified using the WOPT global weights they are then classified based on the similarity measure with the category centroid classifiers. Needless to say the category centroid classifiers remain unchanged when utilized for classification before and after the implementation of WOPT.

The results are presented for Pre-WOPT (classification based on un-modified test vectors) and WOPT (classification based on WOPT modified test vectors) for three types of document collections described as:

1) Training and test document vectors which have binary term weights.
2) Training and test document vectors which have TFIDF term weights.
3) Training and test document vectors which have *ConfWeight* term weights.

4.4 Results for Pre-WOPT(Bin) and WOPT(Bin)

The document term vectors in this application of our framework in both the test and the training set are binary weighted. The procedure for classification has been

explained earlier and it is to be remembered that the category centroids (category classifiers) are retained as the classifier vectors for the Pre-WOPT and WOPT implementations.

What changes as a result of WOPT model implementation are the term weights for documents in the test collection. Global term weights/optimal modifiers are obtained which are used to modify the term weights in the test documents. These WOPT modified test document vectors are then classified by comparison with the category classifiers. The results for the Pre-WOPT(Bin) model is shown in Table 4.9 below:

Category	Accurately Classified	Total Documents in Category	Accuracy %
E51	1903	1914	99.42
C41	1873	1916	97.75
C24	1645	2072	79.39
C33	1430	1763	81.11
Total	6851	7665	89.42

Table 4.10: Results for Pre-WOPT(Bin).

The classification results obtained after implementing WOPT(Bin) are influenced by the model parameter λ that was used in the model. The model parameter is varied to yield different sets of term weights which are used as optimal modifiers to modify the term weights in the test document vectors. These modified test document vectors are classified once again and the classification accuracy is recorded for every change in the model parameter λ. These classification accuracy values for each category is presented in Table 4.10.

Category	Classification accuracy percentage for various λ values							
	0.001	0.005	0.01	0.05	0.09	0.1	0.5	0.9
E51	89.39	90.75	91.45	91.64	91.89	92.21	92.56	93.07
C41	94.62	95.09	95.59	95.91	96.12	96.23	95.74	95.76
C24	89.19	89.98	90.31	90.69	90.34	89.95	87.35	85.87
C33	82.98	84.71	85.59	86.13	85.92	85.82	83.16	81.50
Total	89.05	90.13	90.74	91.09	91.07	91.05	89.70	89.05

Table 4.11: Results for WOPT(Bin) for different values of model parameter.

The graphical comparison of the classification accuracy values for Pre-WOPT classification of test document vectors and classification of test document vectors with WOPT estimated term modifiers is illustrated in Figure 4.4. WOPT estimated term modifiers classify documents belonging to categories C24 and C33 with higher accuracy compared to when classification is carried out without the use of WOPT estimated term modifiers. For categories E51 and C41 the optimal term modifiers may modify term weights which may result in lower classification accuracy for these categories but an improvement for the other two categories. The implementation of WOPT in this case provides an overall classification accuracy which is as good as that offered by the Pre-WOPT(Bin) model indicating that our framework is successfully applicable to binary term weighted document vectors.

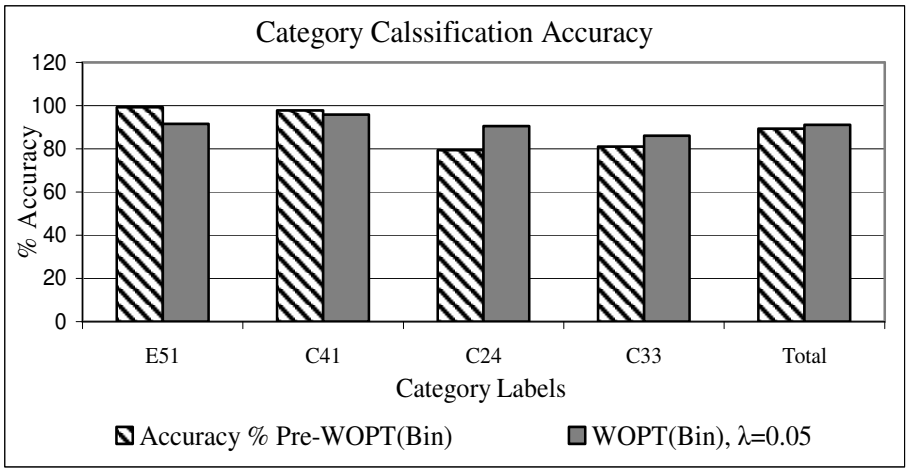

Figure 4.4: Graphical comparison of Pre-WOPT(Bin) and WOPT(Bin).

4.5 Results for Pre-WOPT(TFIDF) and WOPT(TFIDF)

The document term vectors in both the test and the training set are TFIDF weighted. The term weighting scheme used is from Salton and Buckley, 1998. The procedure for classification has been explained earlier and it is to be remembered that the

category centroids which are the chosen classifiers are used to classify the test vectors in the Pre-WOPT and WOPT frameworks. What changes after the WOPT model, are the modification of the test document vector term weights. Global term weights are obtained as a result of the WOPT model and are used to modify the term weights in the test documents. These WOPT modified test document vectors are then classified by comparison with the category classifiers retained from the Pre-WOPT implementation. The results for the Pre-WOPT(TFIDF) model are shown in Table 4.11.

Category	Accurately Classified	Total Documents in Category	Accuracy %
E51	1902	1914	99.37
C41	1869	1916	97.55
C24	1742	2072	84.07
C33	1449	1763	82.19
Total	6962	7665	90.795

Table 4.12: Results for Pre-WOPT(TFIDF)

The classification results obtained after implementing WOPT(TFIDF) are influenced by the model parameter λ that was used in the model. The model parameter is varied to yield different sets of term weights which are used as optimal modifiers to modify the term weights in the test document vectors. These modified test document vectors are classified once again and the classification accuracy is recorded for every change in the model parameter λ. These classification accuracy values for each category is presented in Table 4.12.

Category	Classification accuracy percentage for various λ values							
	0.001	0.005	0.01	0.05	0.09	0.1	0.5	0.9
E51	97.74	90.73	90.93	91.06	91.41	91.36	91.90	91.99
C41	95.56	95.51	95.67	95.49	95.38	95.52	95.46	95.28
C24	88.46	88.15	87.21	87.01	87.38	87.60	87.51	87.36
C33	85.53	85.99	86.14	86.1	86.28	86.53	86.70	86.96
Total	91.82	90.10	89.99	89.92	90.11	90.25	90.39	90.40

Table 4.13: Results for WOPT(TFIDF) for different values of model parameter

The graphical comparison of the classification accuracy values for Pre-WOPT classification of test document vectors and classification of test document vectors with WOPT estimated term modifiers is illustrated in Figure 4.5. WOPT estimated term modifiers classify documents belonging to categories C24 and C33 with higher accuracy compared to when classification is carried out without the use of WOPT estimated term modifiers. For categories E51 and C41 the optimal term modifiers may modify term weights which may result in lower classification accuracy for these categories but an improvement for the other two categories. The implementation of WOPT in this case provides an overall classification accuracy which is as good as that offered by the Pre-WOPT(TFIDF) model indicating that our framework is successfully applicable to TFIDF term weighted document vectors.

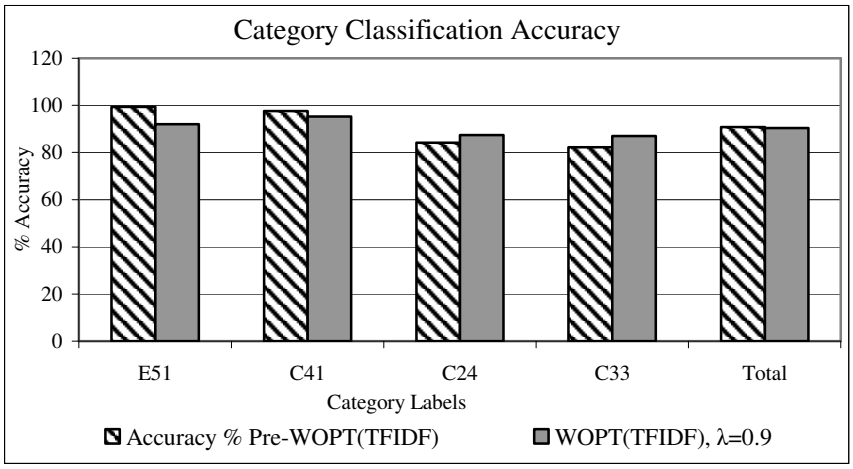

Figure 4.5: Graphical comparison of Pre-WOPT(TFIDF) and WOPT(TFIDF).

4.6 Results for Pre-WOPT(*ConfWeight*) and WOPT(*ConfWeight*)

To implement the WOPT framework for *ConfWeight* document term vectors the TFIDF document term vectors were modified in two phases. The first phase of modification required the TFIDF term weight to be reduced to its TF (Term Frequency)

component. This was done by dividing each term weight by its IDF weight (Inverse Document Frequency) which was provided as part of the RCV1-v2 distribution.

The second phase required the determination of term weights based on the *ConfWeight* measure as described earlier which became the new IDF (Inverse Document Frequency) component which was multiplied with the TF component obtained above. The document vectors so produced are called *ConfWeight* weighted document term vectors. This re-computation of term weights for document-term vectors was carried out for all the documents in the training and test set considered independently. The classifiers were created based on the documents in the categories once the *ConfWeight* measures were implemented. Classification accuracy values were obtained for classification of test documents in the absence of WOPT estimated optimal term modifiers (Pre-WOPT(*ConfWeight*) and with the implementation of the WOPT estimated term modifiers.

Results for Pre-WOPT(*ConfWeight*) are provided in the table below:

Category	Accurately Classified	Total Documents in Category	Accuracy %
E51	1873	1914	97.86
C41	1858	1916	96.97
C24	1293	2072	62.40
C33	1031	1763	58.48
Total	6055	7665	78.93

Table 4.14: Results for Pre-WOPT(*ConfWeight*).

The classification results obtained after implementing WOPT(*ConfWeight*) are influenced by the model parameter λ that was used in the model. The model parameter is varied to yield different sets of term weights which are used as optimal modifiers to modify the term weights in the test document vectors. These modified test document vectors are classified once again and the classification accuracy is recorded for every

change in the model parameter λ. The classification accuracy values for each category is presented in Table 4.14.

Category	Classification accuracy percentage for various λ values							
	0.001	0.005	0.01	0.05	0.09	0.1	0.5	0.9
E51	92.84	93.41	93.66	94.30	93.89	93.82	93.56	93.70
C41	96.86	97.10	97.23	95.91	95.13	95.10	94.28	94.31
C24	88.56	88.56	87.41	81.89	78.16	77.11	66.75	63.95
C33	85.93	86.44	86.34	84.38	81.29	80.69	76.79	73.04
Total	91.05	91.38	91.16	89.12	87.12	86.68	82.85	81.25

Table 4.15: Results for WOPT(*ConfWeight*) for different values of model parameter.

The graphical comparison of the classification accuracy values for Pre-WOPT classification of test document vectors and classification of test document vectors with WOPT estimated term modifiers is illustrated in Figure 4.6. WOPT estimated term modifiers classify documents belonging to categories C41, C24 and C33 with higher accuracy compared to when classification is carried out without the use of WOPT estimated term modifiers. For category E51 the optimal term modifiers may modify term weights which may result in lower classification accuracy for this category but an improvement for the other three categories.

The implementation of WOPT in this case provides an overall classification accuracy which is as higher than what is offered by the Pre-WOPT(*CongWeight*) model indicating that our framework is successfully applicable to *CongWeight* term weighted document vectors. From Figure 4.6 showing the comparisons of the results obtained by implementing the WOPT model we observe that the WOPT implementation provides a successful framework to estimate term weights which when used as an optimal term modification scheme for document classification performs as well as in the case of other term weighting schemes. When using binary document term vectors the WOPT model

estimated term weights cause the classification accuracy to be as good as when classifying TFIDF term weighted document term vectors.

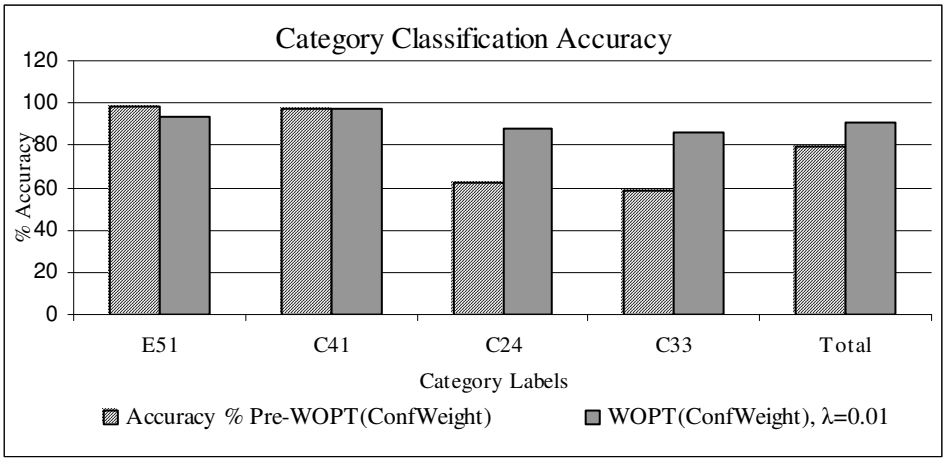

Figure 4.6: Graphical comparison of Pre-WOPT(*ConfWeight*) and WOPT(*ConfWeight*).

This similarity of classification accuracy that can be obtained by using the framework with binary term weighted document vectors reduces the computational expense when determining term frequency or frequency of occurrence of terms in a document. A binary representation can be utilized to provide document-category relationships which modeled in the WOPT framework will yield term weights similar to the TFIDF weighting scheme or the *ConfWeight* term weighing procedure.

The framework proposed provides a successful implementation of representing document-category relationships and the estimation of optimal term weights to preserve the inequalities that can be constructed with these relationships.

Chapter 5: Conclusions and Perspectives

Our framework is multi-functional in representing document-group memberships and in order to preserve such memberships of documents to user-defined categories our framework also estimated optimal term weights towards this end. In the presence of document term vectors that have already been weighted by a weighing scheme such as TFIDF [Salton and Buckley, 1998] and *ConfWeight* [Soucy and Mineau, 2005] WOPT may provide an optimal scaling or term weight modification mechanism to leverage object-category relationships as described in this work.

In the case of binary term weighted document vectors WOPT utilizes object-category relationships to estimate term weights where such weights are used to preserve the groups of documents that has been defined and the logical constraints between documents and group centroids which have been constructed from which the model constraints have been derived.

If WOPT estimated weights can provide comparable performance to TFIDF or *ConfWeight* schemes it would then be un-necessary to count the occurrences of terms in documents to establish term frequencies. This also points to that idea that multiple occurrences of a term in a document may not necessarily better capture the document's intent. Also analyzed is the need for document length normalization which is considered unnecessary when only certain sub-sections of a document are being scanned for keywords and when the keyword set is pre-determined. This is helpful when considering web search systems that scan a webpage for the keywords in certain sections of the page against the list of keywords provided by the user (search string or query) using the search system/engine.

WOPT provides a framework to test the category-relation of documents. In constructing document-category relationships and testing the binary decision variables for their values at the completion of a model run it is possible to choose which category labels a document subscribes to. The decision variables assume a value 1 in the WOPT model to satisfy a logical constraint. These logical constraints may be created to represent the similarity of the document with the centroids of multiple categories in order to choose which category the document is most similar to. This may lead to re-categorization of documents which may have been manually and wrongly categorized by users based on empirical observation of the document. This process of re-classification creates an efficient and robust archiving process. Definitions of categories and also the conditions which need to be met by documents to be members of such categories change as a function of time. In re-classifying documents one is able to observe the changing trends in the meaning of categories and the change in the primary direction of content/subject of the documents that are added to the category. When the documents are checked periodically for their category membership based on updated content it will be possible to create new categories to better separate documents which may have belonged to a category earlier but now would need a separate category of their own since the primary subject of the documents in the category has changed.

WOPT is thus a framework which learns from the known divisions of documents based on category information and can be used to create document-category relationships. It is necessary to create these document-category relationships since they provide constraints for the mathematical programming model in WOPT. Constraints can

be developed differently and derived from other representations of objects in relation to the categories in the document space.

WOPT estimates zero value for certain solution variables (terms/keywords) which indirectly causes reduction of features. The zero valued terms are those terms estimated by the model to be of no semantic importance in deciding the category identity of documents. The zero-valued terms may be as many as the non-zero valued terms and sometimes more. Such a situation is avoided in this framework work by assigning the minimum of the non-zero WOPT estimated term weights to those terms that have been assigned zero valued weights by WOPT. It would be interesting to observe the influence on performance metrics when the zero-valued weights could be given differential values from a distribution with the distribution having a maximum value equal to the minimum of the non-zero weights estimated by the model.

WOPT has been developed as a general framework which has many implications in text/data mining two of which as shown in the work; specifically Information Retrieval and Text Classification. Moreover the Vector Space Model, and other similar data representative models like those employed in Genetic Algorithms and Evolutionary Programming may be employed in a framework similar to that of WOPT.

References

1) Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., and Verkamo, I. Fast Discovery of Association Rules. Advances in Knowledge Discovery and Data Mining, 1996.
2) Allan, J., Carbonell, J., Doddington, G., Yamron, J., and Yang, Y. Topic detection and tracking pilot study: Final report. Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop.
3) Baeza-Yates, R. and Ribeiro-Neto, B. Modern Information Retrieval. Addison-Wesley Longman Publishing Company, 1999.
4) Brachman, R., and Anand, T. The process of knowledge discovery in databases: A human-centered approach. Advances in Knowledge Discovery and Data Mining, 1996.
5) Brin, S., and Page, L. The anatomy of a large-scale hypertextual Web search engine. Seventh International World Wide Web Conference, 1998.
6) Chen, H., and Sharp, B. M. Content-rich biological network constructed by mining PubMed abstracts. BMC Bioinformatics, 2004.
7) Cutting, D. R., Pedersen, J. O., Karger, D., and Tukey, J.W. Scatter/Gather: A cluster-based approach to browsing large document collections. Proceedings of the 15th Annual International ACM/SIGIR Conference, 1992.
8) DARPA TDT. Topic Detection and Tracking: Event-based Information Organization, James Allan. Springer, 2002.
9) Debole, F., and Sebastiani, F. Supervised term weighting for automated text categorization. ACM Symposium on Applied Computing, 2003.
10) Derthick, M., Kolojejchick, J., and Roth, F. An interactive visualization environment for data exploration. Proceedings of the Third Annual Conference on Knowledge Discovery and Data Mining (KDD), 1997.
11) Dubin, D. S. Structure of Document Browsing Spaces. Doctoral Dissertation, School of Information Sciences, University of Pittsburgh, 1996.
12) Feldman, R., and Dagan, I. KDT- knowledge discovery in texts. Proceedings of the First Annual Conference on Knowledge Discovery and Data Mining (KDD), 1995.

13) Good, I. J. The population frequencies of species and the estimation of population parameters. Biometrika, 1953.

14) Grossman, D. A., and Frieder, O. Information Retrieval: Algorithms and Heuristics. Kluwer Academic Publishers, 1998.

15) Hall, J., Mani, G., and Barr, D. Applying computational intelligence to the investment process. Proceedings of Computational Intelligence in Financial Engineering, 1996.

16) Ham, E. H. Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification. PhD thesis, University of Minnesota, 1999.

17) Hang Seng Index. Hang Seng Indexes Company Limited, Hang Seng Bank, Hong Kong, PRC.

18) Hauptmann, A. Integrating and using large databases of text, image, video and audio. IEEE Intelligent Systems, 1999.

19) Hearst, M. A. Untangling Text Data Mining. Proceedings of ACL, 1999.

20) Hoffman, R., and Valencia, A. IHOP: A Gene Network for Navigating the Literature, Nature Genetics, 2004.

21) Jin, R., Chai, J. Y., and Si, L. Learn to Weight Terms in Information Retrieval Using Category Information. International Conference on Machine Learning, 2005.

22) Joachims, T. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. European Conference on Machine Learning, 1998.

23) Jones, W.P., and Furnas, G.W. Pictures of relevance: A geometric analysis of similarity measures. Journal of the American Society for Information Science, 1987.

24) Kuhns, J. L. The continuum of coefficients of association. Statistical Association of Methods for Mechanized Documentation, 1964.

25) Lancaster, F. W. Information Retrieval Systems: Characteristics, Testing and Evaluation. Wiley, New York, 1968.

26) Lewis, D. D., Yang, Y., Rose, T., and Li, F. RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research, 2004.

27) Mittermayer, M., and Knolmayer, G. F. Text Mining Systems for Market Response to News: A Survey. Institute of Information Systems, University of Bern, 2006.

28) Narin, F., Hamilton, K. S., and Olivastro, D. The increasing linkage between technology and public science. Research Policy, 1997.

29) NASDAQ COMP. NASDAQ Composite Index Methodology, National Association of Securities Dealers Automated Quotation System, Financial Industry Regulatory Authority (FINRA), USA.

30) Peat, H. J., and Willett, P. The limitation of term co-occurrence data for query expansion in document retrieval systems. JASIS, 1991.

31) PubMed available via the NCBI Entrez Retrieval System, National Center for Biotechnology Information (NCBI), National Library of Medicine (NLM), U.S. National Institutes of Health (NIH), http://www.pubmed.gov.

32) Ramadan, N. M., Halvorson, H., Vandelinde, A., and Levine, S. R. Low brain magnesium in migraine. Headache, 1989.

33) Rijsbergen, C. J. Information Retrieval, Butterworths, 1975.

34) Robertson, S., and Jones, S. Relevance weighting of search terms. Journal of American Society of Information Science, 1976.

35) S & P 500. Index Mathematics Methodology, Standard and Poor's, New York, 1997.

36) Salton, G., Yang, C., and Wong, A. A vector-space model for automatic indexing. Communications of the ACM, 1975.

37) Salton, G. Automatic Text Processing. Addison-Wesley, 1989.

38) Salton, G., and Buckley, C. Term-weighting approaches in automatic text-retrieval. Information Processing and Managament, 1988.

39) Shapiro, G., Braachman, R., Khabaza, T., Klosgen, W., and Simoudis, E. An overview of issues in developing industrial data mining and knowledge discovery applications. Second Int. Conference on Knowledge Discovery and Data Mining, 1996.

40) Singhal, A. Term Weighting Revisited. PhD thesis, Cornell University, 1997.

41) Soucy, P., and Mineau, G.W. Beyond TFIDF Weighting for Text Categorization in the Vector Space Model, Proceedings of IJCAI, 2005.

42) Spangler, W. E., May, J. H., and Vargas, L. G. Choosing data-mining methods for multiple classification: representational and performance measurement implications for decision support. Journal of Management and Information Systems, 1999.

43) Swanson, D. R., and Smalheiser, N. R. Assessing a gap in the biomedical literature: Magnesium deficiency and neurological disease. Neuroscience Research Communications, 1994.

44) Uthurusamy, R. From data mining to knowledge discovery: Current challenges and future directions. Advances in Knowledge Discovery and Data Mining, 1996.

45) Walker, M.G., Volkmuth, W., Sprinzak, E., Hidgson, D., and Klingler, T. Prostate cancer genes by genome-scale expression analysis. Technical Report (unnumbered).

46) Weir, N., Fayyad, U.M., Djorgovski, S.G., and Roden, J. The SKICAT System for Processing and Analyzing Digital Imaging Sky Surveys. Publications of the Astronomical Society of the Pacific, 1995.

47) Xu, J., and Croft, W.B. Query expansion using local and global document analysis. Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996.

48) Zobel, J., and Moffat, A. Exploring the similarity space. SIGR Forum, 1998.