Automatic Identification of user goals in web search based on classification of click-through results

Thesis submitted for the Award of the Degree of

Masters of Technology in Computer Science and Engineering

by

Amar Kumar Dani

(03CS3014)

Under the guidance of

Prof. Chittaranjan Mandal & Prof. Pabitra Mitra

" YOGA KARMASU KAUSALAM "

Department of Computer Science & Engineering

Indian Institute of Technology

Kharagpur-721302, India

May, 2008

Certificate

This is to certify that the report entitled '**Automatic Identification of user goals in web search based on classification of click-through results**' submitted by Mr. **Amar Kumar Dani** to the Department of Computer Science & Engineering, Indian Institute of Technology, Kharagpur in partial fulfillment of the requirement for the degree of **Master of Technology** during the academic **year 2007-2008** is a record of authentic work carried by him under my supervision and guidance.

Prof. Chittaranjan Mandal

Dept. of Computer Science and Engineering

Indian Institute of Technology

Kharagpur 721302, INDIA

May, 2008

Prof. Pabitra Mitra

Dept. of Computer Science and Engineering

Indian Institute of Technology

Kharagpur 721302, INDIA

May, 2008

Table of Contents

Abstract

The Web is a huge resource for people who use search engines to search for specific pages related to their specific needs. As a result, search engines are continuously striving to improve their ranking algorithms to efficiently fulfill end users' search needs. While such algorithms are effective in handling large volumes of web documents and queries, an understanding of web queries remains quite primitive. In recent years, extensive study has been performed to characterize how users seek information on the web. Such studies focus on how users modify queries and what are the possible user goals in web search. This project is inspired by a study about identification of user goals in web search carried out by Broder which described how the goal behind a web query can be classified into three categories: Navigational searches are those which are intended to find a specific web site that the user has in mind; informational searches are intended to find information about a topic; transactional searches are intended to perform some web-mediated activity. The objective of this work is to identify automatically if the user query has a predictable goal and if it does have a unique goal, what it really is. The results are very promising. The identification of user goals can ultimately be used to achieve efficient and effective ranking of search engine results. The design of a Search Engine based on user goals is also presented in the work.

Chapter 1

Introduction

Given the impact of search engines on the Web users' experience, improving the quality of search results has become the holy grail of search engine operators. As part of this endeavor, there has been a recent interest in identifying the “goal" of a user during a search, so that the identified goal can be used to improve page ranking as well as the presentation of the search results.

If we imagine seeing the world from the perspective of a search engine, our only view of user behavior would be the stream of queries users produce. Search engine designers often adopt this perspective, studying these query streams and trying to optimize the engines based on such factors as the length of a typical query. Yet this same perspective has prevented us from looking beyond the query, as to *why* the users are performing their searches in the first place.

The “why” of user search behavior is actually essential to satisfying the end user’s information need. After all, users don’t sit down at their computer and say to themselves, “I think I’ll do some searches.” Searching is merely a means to an end – a way to satisfy an underlying goal that the user is trying to achieve. By “underlying goal,” we mean how the user might answer the question “why are you performing that search?” That goal may be to gain information about some topic, to buy some gift from an online shop or to navigate to the homepage of some website.

What difference would it make if the search engine knew the user’s goal? At the very least, the engine might provide a user experience tailored toward that goal. For example, the display of relevant advertising might be welcome in a shopping context, but unwelcome in a research context. The underlying relevance-ranking algorithms that determine which results are presented to users might differ depending on the search goal. For example, if the user’s intentions are identified to be transactional, a results page representing transactional features could be ranked higher than an informational page which in case a results page representing informational goal would be ranked higher.

1.1 Classical Information Retrieval vs Web Information Retrieval

Classic IR (information retrieval) is inherently predicated on users searching for information, the so called "information need". But the need behind a web search is often not informational - it might be navigational (give me the url of the site I want to reach) or transactional (show me sites where I can perform a certain transaction, e.g. shop, download a file, or find a resource). A central tenet of classical information retrieval is that the user is driven by an information need. Schneiderman, Byrd, and Croft define information need as "the perceived need for information that leads to someone using an information retrieval system in the first place." But the intent behind a web search is often not informational. In fact, informational queries constitute less than **50%** of web searches.

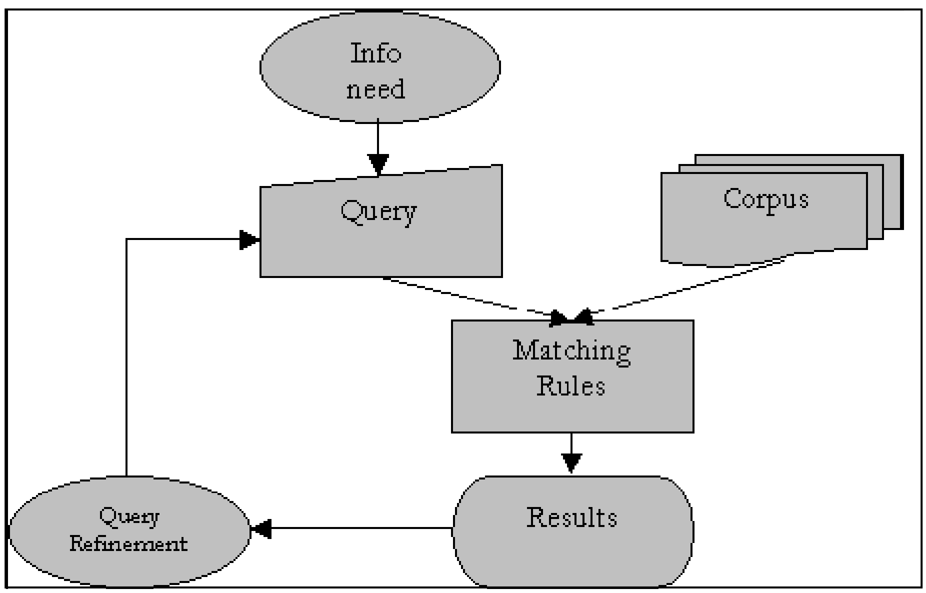

Figure 1: Classical Information Retrieval System

Figure 1 shows a classic IR system. Essentially, a user, driven by an information need, constructs a query in some query language. The query is submitted to a system that selects from a collection of documents (corpus), those documents that match the query as indicated by certain matching rules. A query refinement process might be used to create new queries and/or to refine the results or to provide the user with new reformulations of the query.

Since in the web context the human-computer interaction factors and the cognitive aspects play a significant role, it is useful to detail this model further as in Figure 2.

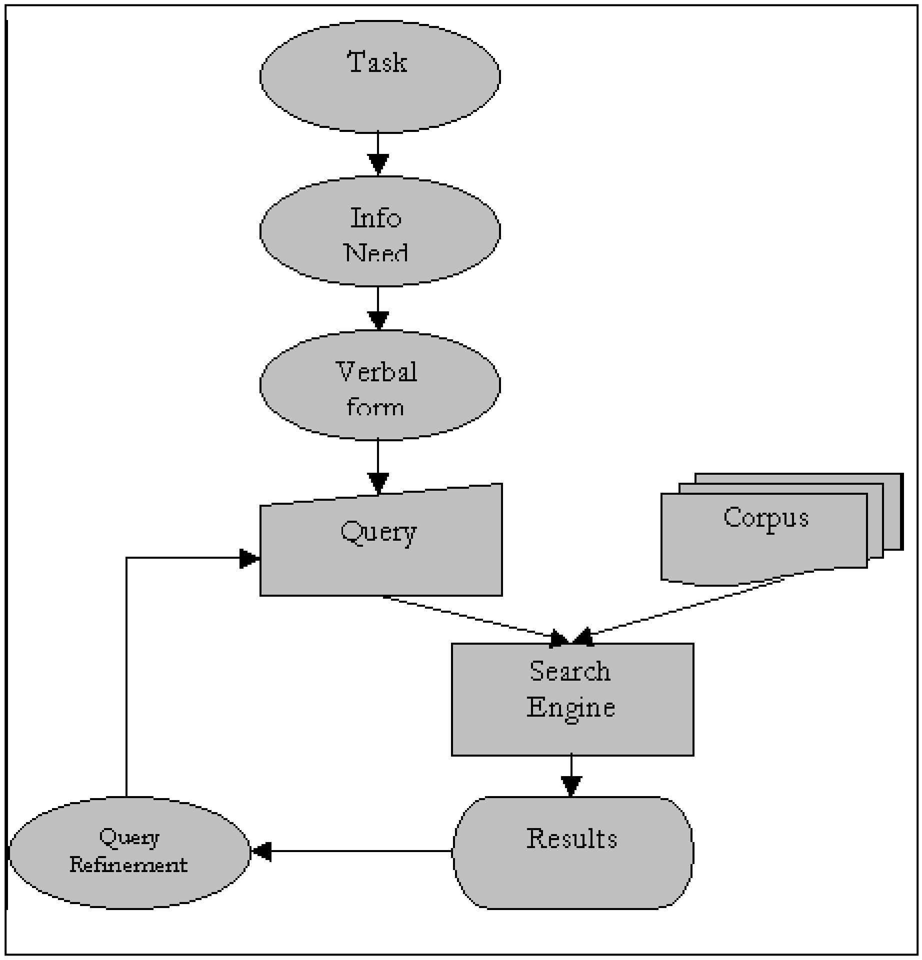

Figure 2: Web Information Retrieval System

Thus we recognize that the information need is associated with some task. This need is verbalized (usually mentally, not loud) and translated into a query posed to a search engine.

Results have confirmed that the common web search user differs significantly from the user model conceived by the traditional IR community. This is stated in the analysis carried out by Jansen and Pooch where the authors compare traditional IR with Web Searching and conclude that the "web is a unique searching environment that necessitates further and independent study". In a comparison between the two IR categories, Jansen and Pooch found out that while the mean length of a traditional IR query is between **6 and 9 terms**, the mean of a web search query is about **2 terms**. This "unique search environment" represents the recent interest in complex subject of understanding the user goals when submitting a query to a search engine. Web Search users tend to make use of short queries to represent their needs, implying that a search engine must make use of other features and algorithms that enhance the relevancy of the search results.

1.2 A taxonomy of web searches

In the web context the "need behind the query" is often not informational in nature. Broder classified web queries according to their intent into 3 classes:

1. Navigational. The immediate intent is to reach a particular site.

2. Informational. The intent is to acquire some information assumed to be present on one or more web pages.

3. Transactional. The intent is to perform some web-mediated activity.

Navigational Queries

The purpose of such queries is to reach a particular site that the user has in mind, either because they visited it in the past or because they assume that such a site exists. Some examples are

• Greyhound Bus. Probable target http://www.greyhound.com

• compaq. Probable target: http://www.compaq.com.

• national car rental. Probable target http://www.nationalcar.com

• american airlines home. Probable target http://www.aa.com

• Google. Probable target http://google.com

• Yahoo. Probable target http://yahoo.com

This type of search is sometimes referred as "known item" search in classical IR. Navigational queries have usually only one "correct" result.

Informational Queries

The purpose of such queries is to find information assumed to be available on the web in a static form. No further interaction is predicted, except reading. By static form we mean that the target document is not created in response to the user query. Informational queries are closest to classic IR queries. What is different on the web is that many informational queries are extremely wide, for instance cars or San Francisco, while some are narrow, for instance normocytic anemia, Scoville heat units. Informational pages are characterized by lot of textual information which is meant to be read by the user. Examples: bird flu, kidney stones, pregnancy, etc.

Transactional Queries

The purpose of such queries is to reach a site where further interaction will happen. This interaction constitutes the transaction defining these queries. We define a transactional page as one where a user can perform some transaction where a transaction is constituted by being able to place an order for some product or to be able to download a file or get to the resource indicated by the query term. Examples:

• Resource finding: dictionary, thesaurus, myspace layouts, funny pictures

• Commercial Transaction: engagement rings, buy cars

• Download file: msn messenger, download Netscape browser

1.3 Literature Survey

Based on the taxonomy presented by Broder, Kang and Kim proposed an automatic query goal identification scheme to distinguish between Navigational and Information queries. They divided a set of web WT10g into 2 sets, DBTopic and DBHome, and based on these sets they extracted features such as the distribution of terms in a query, the mutual information between the query terms, the usage rate of query terms as anchor texts and POS information. However, the authors concluded that there is a significant inadequacy in the proposed approach for classifying queries.

Lee et al. built upon this work and substantiated the idea that the process of automatic query-goal identification is a feasible objective in Web IR. In an initial analysis following a human survey they demonstrate how more than half the queries have a predictable goal (the intention is not ambiguous) and that around 80% of those with an unpredictable goal are either software or person names. Their work also introduced two new features for automatic classification: click distribution and anchor link distribution which yielded an accuracy of 90% for query classification between navigational and informational query classes. Both features are modeled using statistical distributions from past user interaction based on the intuition that if a particular hyperlink shows authoritativeness in terms of a given query, the most probable intention is navigational.

Both Broder and Rose and Levinson observe that the “need” behind considerable amount of queries is transactional. Kang proposes a scheme that serves transactional queries postulating that hyperlinks are a good indicator in classifying queries and collecting relevant pages for transactional queries. The author suggests that by observing the actions related to a hyperlink, cue expressions related to transactional queries can be extracted from tagged anchor texts and titles. These actions are determined by observing the link types of the hyperlinks extracted from relevant web documents.

A frequent occurrence of music, text, application and service link types suggest that the intention of the query is transactional. In a separate study, Li et al. propose a mechanism for identifying transactional queries by building a transactional annotator from a corpus collected from the web that is capable of highly specific labeling of many distinct transaction types. The authors suggest that transactional features engineering, hand crafted regular expressions and an index of terms are suitable and robust for identifying transactional terms within a web document. The process relies on regular expressions that identify the existence of transactional patterns and a dictionary of negative patterns that evaluates the presence of any negative terms collected by the object identifier.

1.4 Motivation

Identifying the end user goal in web search can be utilized for improving the search engine results presentation in a big way. This has already been utilized in the Yahoo mindset search engine which estimates the commercial intent of the user and presents the results along with a metric estimating the commercial content of a web site. The user goals can be utilized to improve web search in the following ways:

Optimization of Relevance Ranking Algorithm

The user goal can be incorporated into the relevance ranking algorithm to reorder the ranking of search engine results. The most relevant result should be presented to the user as the first result such that the user does not have to scroll down to view the relevant result. If the end goal of the user is identified to be navigational, then only one result best matches with his goal whereas if the end goal is identified to be informational or transactional, other methods can then be employed to identify the most relevant page. These methods could include page rank algorithm used by Google or can also take the click-through results into consideration which indicates which pages have received considerable amount of clicks for a query. Further, for ambiguous queries for which the end goal cannot be determined uniquely, the top results can include top results from each class so that the goal of each user can be fulfilled.

Clustering of Search Engine results

The search engine results can be presented as clusters of informational, navigational or transactional with each cluster including the top pages for each class. The search engine clusty clusters the results into various classes but the clustering is unsupervised and not into known classes. Clustering the results into these three classes and then hierarchically into smaller clusters within each higher level class can lead to better organization of search engine results and meet the requirements of all users of search engine.

Display of Advertisements

The display of advertisements is relevant only if the end user has a transactional goal. Further the relevant advertisements can be determined in case of informational goal by identifying what informational is being sought by the user. For example, if the end user is seeking information on cars, ads relevant to cars can be displayed. In case of navigational queries, the display of ads becomes irrelevant. In this way, the search engine results page can be optimized.

Display of text snippets for search engine results

The display of text snippets can be targeted based on the end users goal. If the end users goal is navigational, the display of text snippet becomes irrelevant. Further for a particular site, a different snippet must be displayed for the case if the end goal is informational and a different snippet must be displayed if the end goal is transactional. For example, for the query 'cars', if the goal is identified to be informational, the most relevant information on the site related to cars must be displayed. But for the query 'buy engagement rings', the relevant text on the site would be the cost information and the specifications of engagement rings which should be displayed as the text snippet.

1.5 Objective

All the approaches to identification of user goals in web search mentioned above have not taken all the three classes of goals into consideration. Lee et al classifies the queries into navigational and informational whereas others provide features useful for navigational and transactional query classification. But, we have seen that most researchers agree on the existence of three fold user intent in web searches as proposed by Broader: navigational, informational and transactional. To be able to utilize any information regarding user intent, any search engine must be able to detect and distinguish between the three classes of user intention. Further, the query intention identification system must be able to clearly distinguish between the ambiguous queries for which the intention is not clearly identified and the unambiguous queries where the intention is clearly identified. This work focuses on automatically identifying whether the query has a predictable goal and if so, detect the goal of the query.

1.6 Experimental Setup and Approach

Our query intention classifier takes the past user click behavior into account to classify the intention of a query entered into a search engine. The click through data of a search engine consists of the query and the url of the result clicked at by the user who issued the query. The approach is based on the intuition that user's goal for a given query may be learned from how users in the past have interacted with the returned results for this query. To classify the intent of query, the click-through pages of the query are classified as navigational, informational or transactional page. Then the dominating class is identified to determine the class of the query. Figure 3 shows the various steps involved in the query classification process.

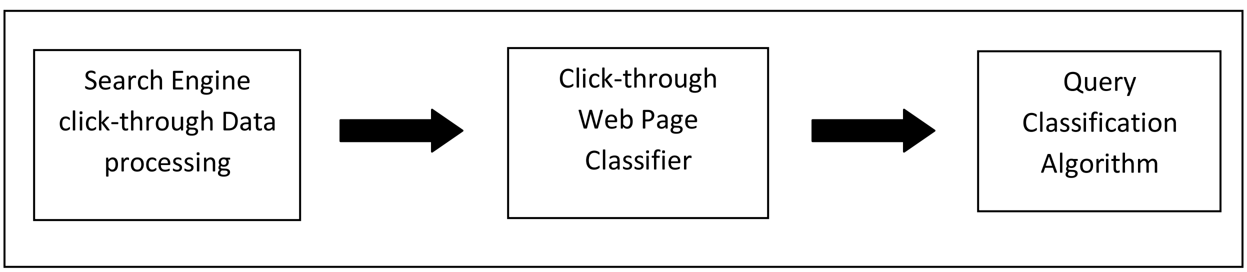

Figure 3: Steps in Query Classification

The 1st step involves first getting the click-through data of search engine for experimental purposes and then to process it to sort the data in order of the number of clicks each query has received, extract the test set of queries, and expand the domain name of click-through via the Yahoo search API by simulating a virtual user.

The 2nd step involves building the three way web page classifier for classifying the click-through url into either navigational, transactional or informational. The corpus is first built by manually classifying a number of pages belonging to each class and then extracting several relevant features to distinguish between the classes, and finally identifying the appropriate machine learning algorithm to achieve the highest 10-fold cross validation accuracy.

The final step includes the query classification algorithm to classify the query into either ambiguous (if the query does not have a predictable goal) or classifying the query into one of the above mentioned classes. The results of automatic classification are then compared with the benchmark set of queries consisting of **65 queries** classified by a user survey involving **30** users.

Chapter 2

Search Engine Click-Through Data processing

In order to build the classifier and to carry out the experiments, the click-through data of a search engine was to be obtained. AOL had released its log of search data to the public in August 2006 which has been used in our experiments. The data has to be preprocessed to extract the queries to be used in our experiments. In this chapter, the AOL data and the data processing steps are described.

2.1 AOL Search Engine click-through data

In order to manually classify the queries, we use the click-through data of AOL search engine. This data is taken from an AOL log of search data released to the public in August 2006. This includes around 36 million search queries from circa 658,000 of AOL's users taken from the period between March 01 2006 and May 31 2006. Each line of data includes an anonymous ID, the actual query, the date and time the query was submitted, the page rank and the domain portion of the URL as the click-through results. The query issued by the user is case shifted with most punctuation removed. The data represents one of two types of events. The first is a query that was not followed by the user. The other is a click-through URL returned by the search engine for that particular query.

2.2 Data processing

Figure 4 shows the various steps involved in processing the AOL search engine data before extracting the queries for classification experiments.

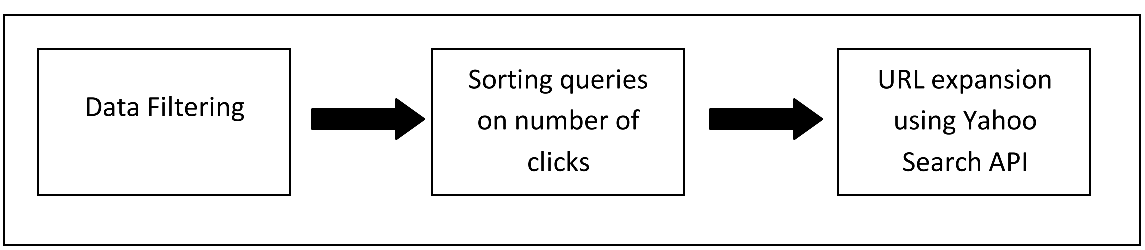

Figure 4: Steps in Data Processing

Data Filtering

The AOL search engine data is filtered to remove extraneous information for the experiments. From the data, the required information is extracted. The data includes the id of the user issuing the query. For each query, if a query is issued by the same user at different times, it is taken to be a duplicate and counted to be one click. Such duplicates are removed for each query and then the total clicks for each query is summed up. The time stamp and the id of the user issuing the query are removed from the data as they are inconsequential in our experiment.

Data Sorting

After filtering the data, the data is stored in different files based on the different alphabets. The queries are then sorted based on the number of clicks received for each alphabet separately. So for each alphabet, we have a sorted list of queries based on the number of clicks received. From this list of sorted queries, the queries to be used for testing would be extracted. So now the data is in the following format:

mortgage calculator http://realestate.yahoo.com 1 742

mortgage calculator http://www.calculators4mortgages.com 3 742

mortgage calculator http://www.mortgagecalc.com 3 742

mortgage calculator http://www.fanniemae.com 4 742

mortgage calculator http://www.interestratecalculator.com 1 742

mortgage calculator http://www.bankrate.com 119 742

mortgage calculator http://mortgage-calculators.org 1 742

mortgage calculator http://mortgage-x.com 1 742

The first term denotes the query keyword, the second term the domain name of click-through, the third the number of clicks by different users for this url-query pair and the last term denotes the total number of clicks for this query.

URL expansion via Yahoo Search API

The AOL search engine click-through data includes only the domain name of click-through url. But for our experimentation purposes, we needed the exact url of the click-through. To facilitate this, we used the Yahoo search engine API to simulate a virtual user firing the queries into Yahoo search engine. The AOL search engine data is of the year 2006. Hence after that many of the sites have become extinct. Search is done using the Yahoo API and the keyword fired for searching includes the query along with the domain name of the click-through. The top 50 results are extracted and the first url whose domain matches with the domain of the click-through is taken to be the expanded url for the given query-click through pair.

Using the AOL data and using the Yahoo search engine for expansion is not detrimental for our experiment since it is like simulating a virtual user firing the queries. For several queries, it was manually observed that the ordering of results for the query domain pair for the AOL search engine was similar to that of the Yahoo search engine. Hence it can be assumed that the user who fired this query and visited a particular site, would have visited this particular page of the site. So the url obtained by url expansion would actually be similar to the url that the user might have actually clicked.

After expansion, the data is in the following format. Some urls that do not match with any of the results returned by the Yahoo API are denoted by DNM (did not match).

mortgage calculator http://realestate.yahoo.com/calculators/payment.html 1 742

mortgage calculator http://www.calculators4mortgages.com/ 3 742

mortgage calculator http://www.mortgagecalc.com/ 3 742

mortgage calculator http://www.fanniemae.com/homebuyers/homepath/index.jhtml 4 742

mortgage calculator DNM 1 742

mortgage calculator http://www.bankrate.com/brm/mortgage-calculator.asp 119 742

Chapter 3

Questionnaire Design and User Survey

In this chapter, we present the description of our human subject study, in which we try to (1) evaluate how many queries have clearly predictable goals and (2) build a benchmark query set against which we can evaluate our automatic identification mechanisms. Our benchmark set consists of **65 queries** selected carefully from the AOL search engine click-through data. To study whether the goals of these queries are predictable regardless of individual users, **30** graduate students were asked to indicate their most probable goal if they issued each query.

3.1 Selection of queries for manual classification

For creating the benchmark set of queries for testing the results of automatic classification, queries with sufficient number of clicks are selected from each of the alphabet sets. 300 is taken to be the threshold for defining sufficient number of clicks. It is difficult to determine what threshold to select for defining sufficient number of clicks. It can be selected by manually classifying a set of queries and then comparing with the automated classification results and comparing with the manual set till the set appears to be matching. But to get such an incremental data for a set of queries, one would need real time access to the click-through data of a search engine which was not feasible in this project. Hence, we take a decent estimate of 300 which gives good results.

After creating a set of queries having a decent number of clicks, the final set of queries for the questionnaire are selected. For the queries we have 6 classes for classification: navigational, informational, transactional and ambiguous with ambiguity of 3 forms: navigational/informational, navigational/transactional and transactional/informational. Our proposed algorithm should be able to distinguish automatically between ambiguous and non ambiguous queries and should be able to detect the type of ambiguity of the query if the query is ambiguous. So ideally, the test set of queries chosen should have representation across all the classes. But it is not possible to identify ambiguous queries across all the ambiguity classes because it is very subjective. So we try to take equal number of queries which seemed to belong to informational/navigational/transactional classes and a few queries that seemed to be ambiguous. The query set included software names and names of people which was reported to be ambiguous by an early study by Lee, Liu and Cho.

3.2 Questionnaire Design

A good design of the survey questionnaire is crucial in collecting reliable results from our user study. In the following, we describe the exact questions that we used in our survey and how our questionnaire has been refined to our final form through multiple revisions. In our initial design stage, we first evaluated whether it is appropriate to directly use the navigational-informational-transactional taxonomy in our questionnaire. For this purpose, we interacted with two participants, first educating them with the taxonomy, and then asking them to classify the 65 queries as either navigational or informational or transactional. Afterwards we interviewed each of them to gather descriptive intentions for some representative queries, and further compared such descriptive intentions with the final navigational/informational/transactional choices. From this comparison we realized that even if two participants had exactly the same descriptive intention, they might end up casting that intention into different navigational-informational-transactional choices.

This confusion was mainly due to the two potential criteria that they could use to classify the user goal. For example, a user might search a person's name in order to reach not only that person's homepage, but also some other related sites, such as news articles about the person. In this scenario, the people who used the first criterion (do you have a particular webpage in mind?) classifed the intention as navigational, because they perceived a particular Webpage (the person's homepage) and reaching that page was part of the goal. On the other hand, the people who used the second criterion (do you intend to visit multiple pages?) classified it as informational because their goal was to gather information from multiple sites including the person's homepage.

Realizing this potential ambiguity and the randomness in the user classification, we decided to ask our subjects to indicate their descriptive intentions directly. Based on their descriptive intentions, we then classify the goal of the queries ourselves. In particular, we decided to present the following three choices to our participants:

Choice 1: You already have a website in your mind (one particular website only) and your intention is to reach that website with the help of the search engine

Choice 2: Your aim is to obtain information on the “query term”

Choice 3: Your aim is to **buy** / **download** or **obtain** the resource implied by the “query term”

The users are also provided a few sample classifications so that they can get a feel of how to classify the given queries. The sample classifications have no relation with the 65 given queries and would create no bias in the end user classification. The sample classified queries given are:

1. Lycos : 1
2. Hair styles : 2
3. Funny videos : 3
4. Myspace backgrounds : 3
5. Guitar Tabs : 2
6. New York Lottery : 1

Note that under both the choices, Choice 1 is clearly navigational because the user intends to visit a single website that he has in mind. Similarly, Choice 2 is clearly informational because the user intends to explore multiple websites and no website is pre-assumed to be the single correct answer and the user is interested in getting information on the query term. Similarly, choice 3 is clearly transactional because the user is interested in undertaking some web-mediated transaction.

3.3 Manual Classification Results

After collecting the survey results from 30 users, the queries are classified into the 6 classes based on the belongingness value of the query in each of the classes navigational/transactional/informational. For each query q, values i(q), n(q) and t(q) are defined which denote the percentage of candidates who have indicated the goal of the query to be informational or navigational or transactional respectively. If the difference between the maximum belongingness value and the 2^{nd} maximum belongingness value is greater than .2, then the query is said to have a predictable goal else the query is said to have belongingness in both the classes. The following tables give the belongingness values of the manually classified queries.

Navigational Queries

Query	N(q)	I(q)	T(q)
Hotmail	1.00	0.00	0.00
Google	1.00	0.00	0.00
Espn	1.00	0.00	0.00
Imdb	0.90	0.10	0.00
Honda	0.67	0.23	0.10
Yahoo	1.00	0.00	0.00
Ask	0.80	0.20	0.00
Amazon	0.83	0.00	0.17
Thesaurus	0.67	0.10	0.23
Suzuki	0.67	0.13	0.20
Microsoft	0.80	0.20	0.00
Encyclopedia	0.70	0.07	0.23
Dell	0.77	0.00	0.23
Pogo games	0.70	0.00	0.30
Ebay	0.90	0.00	0.10

Table 1: Manual classification results for Navigational Queries

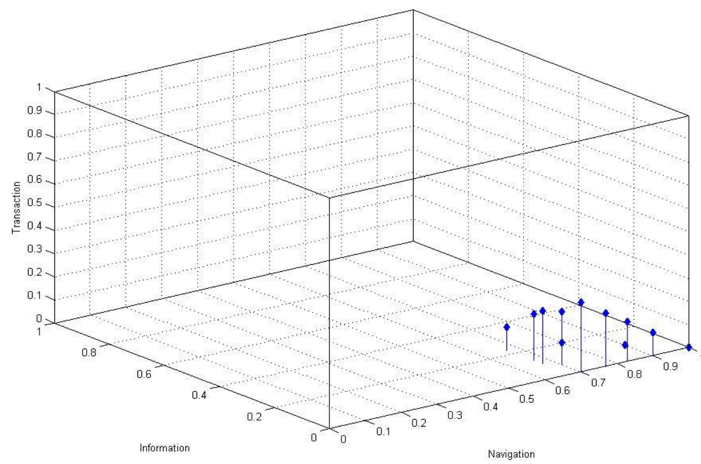

Figure 5: Distribution of navigational queries

Transactional Queries

Query	N(q)	I(q)	T(q)
Mortgage Calculator	0.00	0.20	0.80
Myspace Layouts	0.00	0.23	0.77
Tattoos	0.00	0.33	0.67
Cigarettes	0.00	0.23	0.77
Funny Pictures	0.00	0.20	0.80
Free music downloads	0.00	0.23	0.77
Msn messenger	0.00	0.20	0.80
Free ringtones	0.00	0.03	0.97
Download	0.27	0.00	0.73
Ipod	0.03	0.20	0.77
Screensavers	0.00	0.03	0.97
Netscape	0.23	0.07	0.70

Deal or no deal	0.24	0.13	0.63
Shoes	0.00	0.20	0.80
Airsoft guns	0.00	0.27	0.73
Aol media player	0.13	0.03	0.84
Itunes	0.17	0.07	0.76
Internet explorer	0.20	0.03	0.77
Sudoku	0.07	0.13	0.80

Table 2: Manual classification results for Transactional Queries

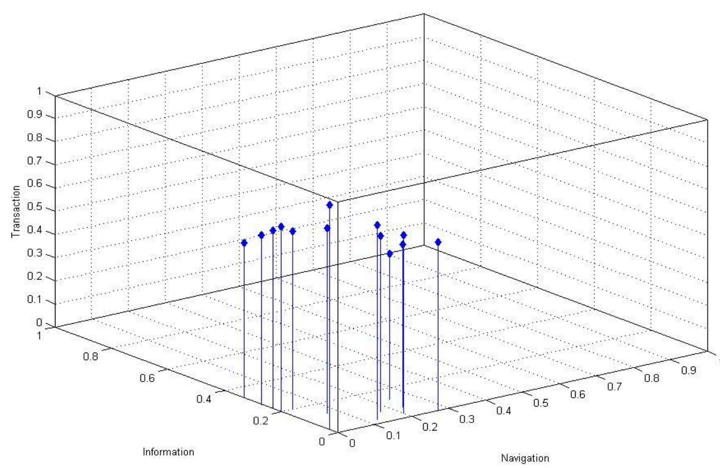

Figure 6: Distribution of transactional queries

Informational Queries

Query	N(q)	I(q)	T(q)
Kidney stones	0.00	1.00	0.00
Bird flu	0.10	0.90	0.00
Employment	0.00	1.00	0.00
Motorcycles	0.00	0.73	0.27
Html	0.00	1.00	0.00

Pregnancy	0.00	1.00	0.00
Snakes	0.00	1.00	0.00
Optical illusions	0.00	0.90	0.10
Exe	0.00	0.73	0.27
Guns	0.00	0.63	0.37
Florida lottery	0.23	0.63	0.14
Airline tickets	0.00	0.63	0.37
Anna benson	0.13	0.6	0.27
Jessica simpson	0.07	0.63	0.30
Paris Hilton	0.10	0.63	0.27
Baby names	0.00	0.70	0.30
Jessica alba	0.03	0.70	0.27
Kelly blue book	0.00	0.63	0.37
Recipes	0.00	0.70	0.30

Table 3: Manual classification results for Informational Queries

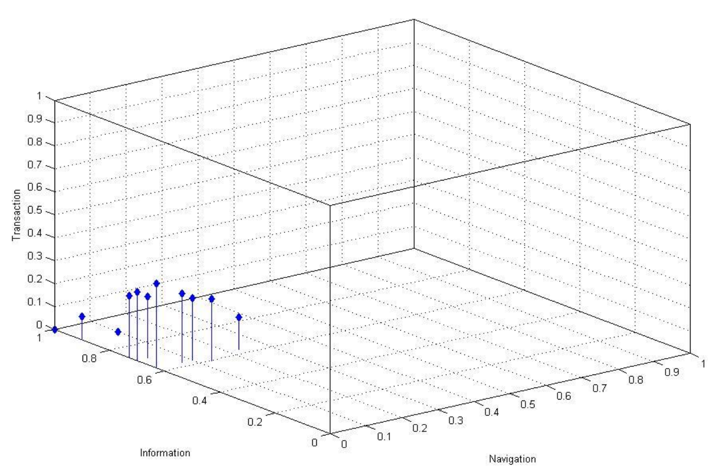

Figure 7: Distribution of informational queries

Informational-Transactional Queries

Query	N(q)	I(q)	T(q)
Furniture	0.00	0.43	0.57
Online games	0.03	0.40	0.57
Costa rica	0.13	0.40	0.47
Britney spears	0.07	0.50	0.43
Shakira	0.13	0.47	0.40
Kelly Clarkson	0.10	0.43	0.47
Reverse lookup	0.00	0.60	0.40
David blaine	0.07	0.50	0.43
Movies	0.13	0.40	0.47
Cars	0.00	0.57	0.43

Table 4: Manual classification results for Informational-Transactional Queries

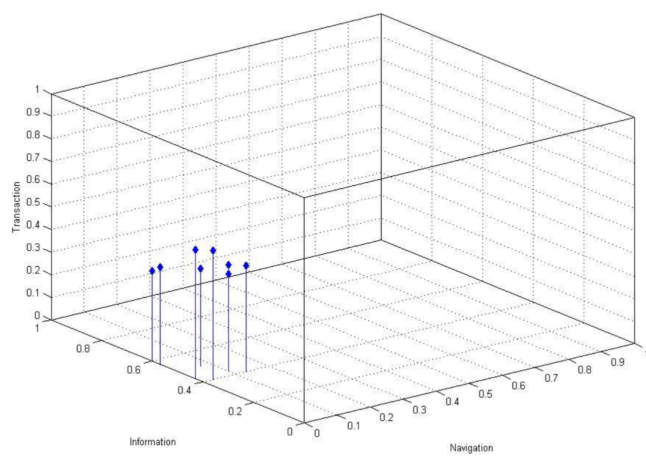

Figure 8: Distribution of informational-transactional queries

Informational-Navigational Queries

Query	N(q)	I(q)	T(q)
Harry Potter	0.43	0.37	0.20

Table 5: Manual classification results for Informational-Navigational Queries

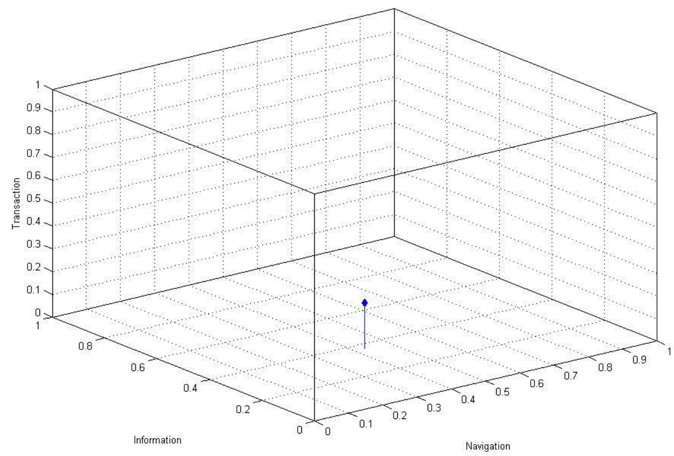

Figure 9: Distribution of informational-navigational queries

Transactional-Navigational Queries

Query	N(q)	I(q)	T(q)
Bible	0.40	0.10	0.50

Table 6: Manual classification results for Transactional-Navigational Queries

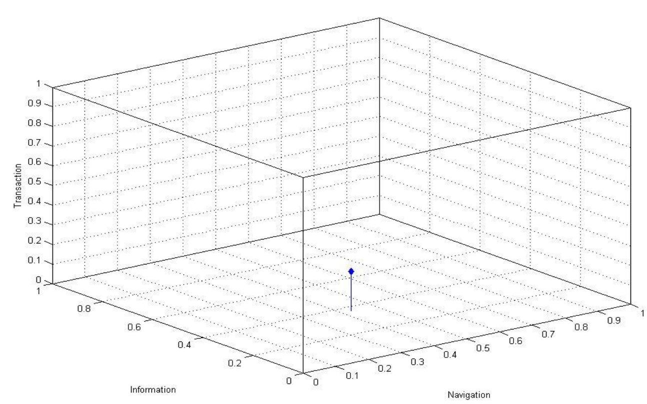

Figure 10: Distribution of navigational-transactional queries

Chapter 4

Web Page Classification

We build a web page classifier which classifies the web page into three classes: navigational, informational and transactional. Features are defined for classifying a web page as navigational or informational or transactional. The web page classifier is the central concept in our query classification algorithm. The features defining a page to be transaction/informational/navigational would ultimately identify a query to be navigational or transactional or informational. Navigational pages are the home pages of web sites and if a person has a navigational intent, he would visit the home page of the web site. So, it is relatively easy to identify whether the visited page is navigational or not. The key to the classification is identifying the features defining transactional and informational pages. The features can be altered based on the final aim of the search engine.

There are two approaches to define transactional/informational pages. One is to define the possible transactions possible like resource finding / download / commercial transactions and then identify the features for each of the type of pages. We observed that if the goal of a user is transactional, he might also visit several navigational pages of sites offering those services. For example for the query dictionary, there were several navigational pages, i.e. home pages of sites which the end user visited. Classifying these pages into transactional would be very difficult and would lead to reduction of accuracy. Hence, to handle such cases the final query classification algorithm was modified.

Another approach which we have also adopted is to define informational pages and transactional pages by the style of presentation. Informational pages have lots of textual material to be read and the amount of text per paragraph also dominates. Further, on a transactional page, the amount of different HTML elements like tables, images, download buttons, etc dominate. We have combined the two approaches to include both transaction identifying features via the bag of words features and identified the HTML elements via HTML features.

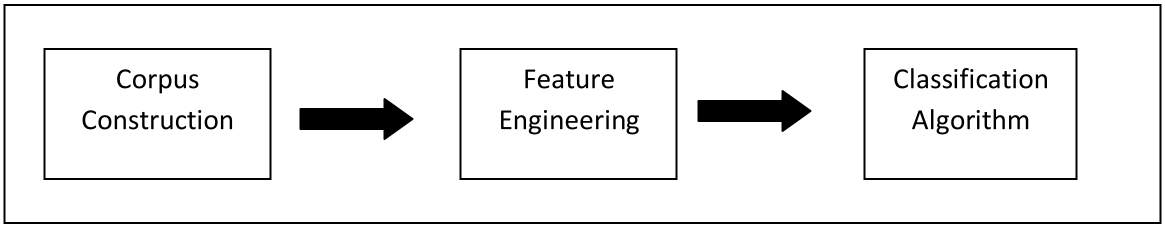

Figure 11: Steps in Web Page Classification

4.1 Class description

The classifier classifies the pages into three classes navigational / informational / transactional each of which are defined by several features identified and extracted from the HTML page, url and query keyword of the query-url pair.

Navigational Class

Navigational pages are the home pages of web sites and if a person has a navigational intent, he would visit the home page of the web site. So, it is relatively easy to identify whether the visited page is navigational or not. It is possible that a person having a transactional goal visits several home pages of different sites. In such a case, it might not be feasible to denote the home page of the site to be navigational. But, classifying the home page of a site as a transactional page when it bears similarity with a navigational page would lead to reduction of accuracy of our classifier. Hence, to overcome such a scenario, the query classification algorithm is altered rather than reduction in accuracy of the classifier. A person having an informational goal is very unlikely to visit the home page of a particular site which is also observed from the AOL search engine click-through data.

Navigational pages also have a very high number of clicks because if the goal of a query is navigational, many people would visit the same site but if the goal is informational or transactional, users would visit different informational/transactional pages because of which the clicks would get distributed. Hence other home pages which do not have a high ratio of clicks relative to the total number of clicks for the query tend to be more transactional in nature. These home pages are classified as navigational pages but the end query classification algorithm is modified to take this into consideration.

Informational Class

The informational class includes pages which contain lots of textual material to be read up. The informative pages are generally not the home pages of sites and have a high url depth. Further, it can be observed that the query keyword occurs more frequently in the latter part of the url not including the domain name. This is also true for the transactional urls but for navigational pages, the query term frequently is the domain name of the web site or it occurs frequently in the domain name of the url. The fact that textual material dominates on informational pages according to our definition, lexical features become essential in distinguishing these pages from the transactional pages which have more of HTML elements dominating relative to the textual material. Lexical features include the count of number of paragraphs, total number of characters occurring in the text, average text length in the paragraphs, etc.

Transactional Class

We define a transactional page as a web page that a user visits to either carry out a commercial transaction or to download something or to find some online resource. Like the informative pages, the transactional pages are generally not the home pages of sites and have a high url depth. Further, it can be observed that the query keyword occurs more frequently in the latter part of the url not including the domain name.

To distinguish the pages defining commercial transaction, we can observe that these pages have very little textual material and common commercial terminology is used like ‘product specification’, ‘hot product’, ‘buy’, ‘sell’, etc. Further these pages have lots of specifications of the product which are also present on the download pages where the software specifications are specified. Hence the bag of words features becomes useful in identifying these pages. For the pages consisting of online resources like dictionary, thesaurus, myspace layouts etc. there are no standard features identifiable other than the fact that more of HTML elements like images, tables, divs dominate on such pages than the textual elements. This is also true for other transactional pages including commercial transaction pages and download pages. Hence, the basic features used for distinguishing transactional pages from informational pages include the lexical features defining the amount of textual material on the HTML page and the HTML features defining the amount of HTML elements relative to the textual material.

4.2 Corpus construction

The corpus consists of the pages of each of the classes used to train the classifier. A good corpus is essential for a good classifier and must encompass all types of pages defining a particular class. The pages for different classes to be used for training are chosen from the AOL search engine click-through expanded urls so that the pages would be representative of the pages that would have to be classified to predict the type of the query.

A total of **322 instances** were manually classified for training the three-fold classifier with **127 navigational** pages, **92 informational** pages and **103 transactional** pages. It was relatively easy to identify navigational pages as the home pages of web sites. The confusion was with classifying a page into transactional or informational page. Initially the corpus was built taking the first approach into consideration where we tried to define the possible transactions possible like resource finding / download / commercial transactions and then identify the features for each of the type of pages. Other pages were classified as informational pages. We saw that it was difficult to identify resource finding pages. Hence we resorted to the second approach whereby the pages which had sufficient textual material as information would be classified as informational pages whereas pages with more transactional features as defined above would be classified as transactional pages.

4.3 Feature Engineering

A total of **152 features** are extracted from the HTML pages by writing a parser of the HTML page and extracting features including HTML, url based features and bag of words features. Then, feature selection algorithm was run to extract the important features. The supervised attribute selection algorithm resulted in **12 best features** whose importance for each class and description is given in the next chapter.

Figure 6 shows the various steps involved in feature extraction from the web page and the url, query keyword and number of clicks given as input to the Html parser and feature extractor written in Python. The various features extracted include the url features, html features and lexical features which are described below. In many cases the Html page is corrupt and has to be cleaned. This is done using the Html tidy software which cleans the html markup wherever possible. After this various features are extracted and written in an arff file which is taken as input file into the weka software which is used to run several classification algorithms.

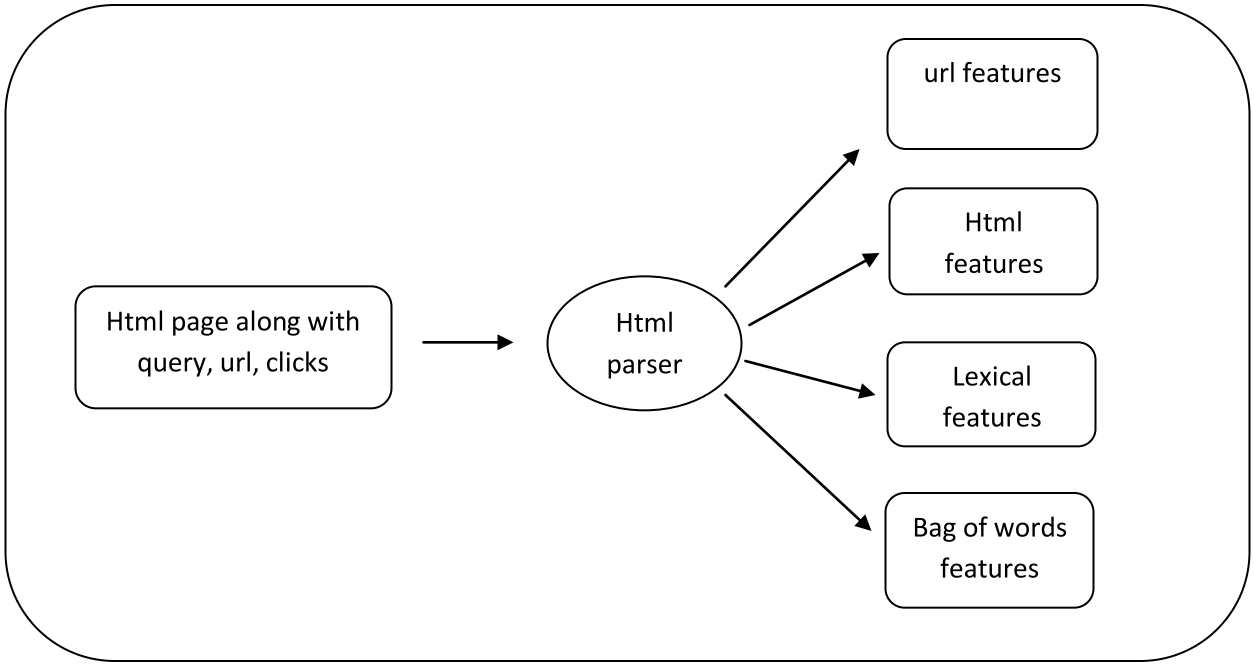

Figure 12: Steps in feature extraction from web page

Url Features

The navigational pages are generally the homepages of web sites and hence have a less depth than other pages of either transactional or navigational pages. The url features used are:

1. url depth
2. length_url
3. Occurrence of query keyword in the domain name
4. Occurrence of query keyword in the latter part of the url
5. Ratio of clicks received for this url to total number of clicks received for the query

HTML Features

The HTML page corresponding to the url is downloaded and saved. The HTML page is parsed and the Title text, anchor text, headings, paragraphs, special texts are stored in different data structures. Several features are used including the frequency and ratio of commonly occurring tags like img, anchor, input boxes, inner hyperlinks(hyperlinks pointing to the same domain), outer hyperlinks(hyperlinks pointing to other domains), table, div, list, form and other commonly occurring html tags.

Lexical Features

The lexical features are based on the fact that for different classes, the lexical features might have distinctive values. The lexical features are specially helpful in distinguishing between the transactional and informational pages. The lexical features used are:

1. chars_per_word
2. sentences_per_p
3. words_per_p
4. sentencess_per_p
5. length_text
6. no_of_words
7. no_of_sentences

Bag of Words Features

This feature is based on the fact that some words are common for specific classes. Occurrence of these words is characteristic for the particular class. These words are selected by manually going through the various pages for the classes. Further the words are weighted differently by its occurrence in meta text, title text, headings, special text, anchor text, alternate text and input text. The bag of words features can be used to identify navigational pages and transactional pages but not informational pages since one cannot identify commonly occurring keywords for all domains of information.

The keywords used in the bag of words feature set include: 'basket', 'buy', 'cart', 'catalogue', 'checkout', 'cost', 'delivery', 'offer', 'order', 'pay', 'price', 'purchase', 'rebate', 'save', 'sell', 'trolley', 'story', 'store', 'shop', 'shipping', 'homepage', 'corporate', 'welcome', 'our', 'my', 'company', 'business', 'products', 'services', 'cost', 'purchase', 'shopping', 'cart', 'now', 'delivery', 'item', 'sale', 'quantity', 'specification', 'dollar', 'customer', 'availability', 'download', 'home page’, ‘products & services', 'online store', 'hot product', 'add to cart', 'shopping cart', 'order now', 'buy now', 'item number', 'product features', 'product details', 'product description', 'product review', 'list price', 'sale price', 'sold out', and 'download now'.

From the above set we see that most of the keywords are to identify the transactional pages whereas a few are to identify navigational pages which include 'homepage', 'corporate', 'welcome', 'our', 'my', 'company', 'business', 'products', and 'services'.

Feature selection Algorithm

After extracting the features and storing the features in an arff file format, the file is opened using the weka tool. The weka tool allows applying several feature selection algorithms which selects the best few features out of the given set of features. This helps to eliminate the features which are not required and selecting the best set of features at the same time. Applying the supervised feature selection algorithm, we get the following 12 best features:

1. ratio_outer_hyperlinks
2. url_depth
3. length_url
4. no_query_first
5. no_query_others
6. ratio_hits
7. length_text
8. no_title
9. no_cost
10. no_rebate
11. no_homepage
12. no_hot_product

4.4 Classification Algorithm and Classification Results

Several classification algorithms including NaiveBayes, J48, Random Forest, SMO and RandomCommittee Meta classifier were experimented with after running the feature selection algorithm. We report the 10 fold cross validation accuracy which is a standard metric used to evaluate the learned classifier. In a 10 fold cross validation evaluation scheme, the training data is divided into 10 sets. The classification model is learned from the first 9 sets and is tested on the 10^{th} set. The process is repeated for all the 10 sets learning on 9 sets and testing on the 10^{th} set. We report the confusion matrix across the three classes and 10 fold cross validation accuracy achieved using all the classification algorithms. We achieve the highest accuracy using the RandomCommitte meta classifier which is finally used to classify the web pages corresponding to the queries used in our experiment.

Naïve Bayes Algorithm

87.8% 10 fold cross validation accuracy is achieved using the Naïve Bayes algorithm.

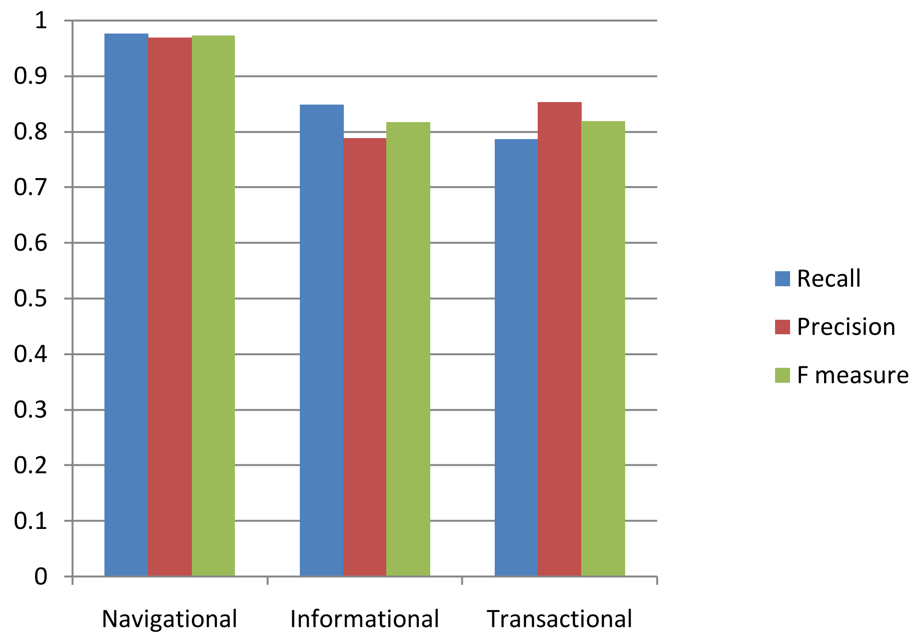

Figure 13: Classification accuracy across classes using Naïve Bayes Algorithm

Navigational	Informational	Transactional	← Classified as
124	1	2	Navigational
2	**78**	12	Informational
2	20	**81**	Transactional

Table 7: Confusion Matrix for using Naïve Bayes Algorithm

J48 Algorithm

87.8% 10 fold cross validation accuracy is achieved using the J48 algorithm.

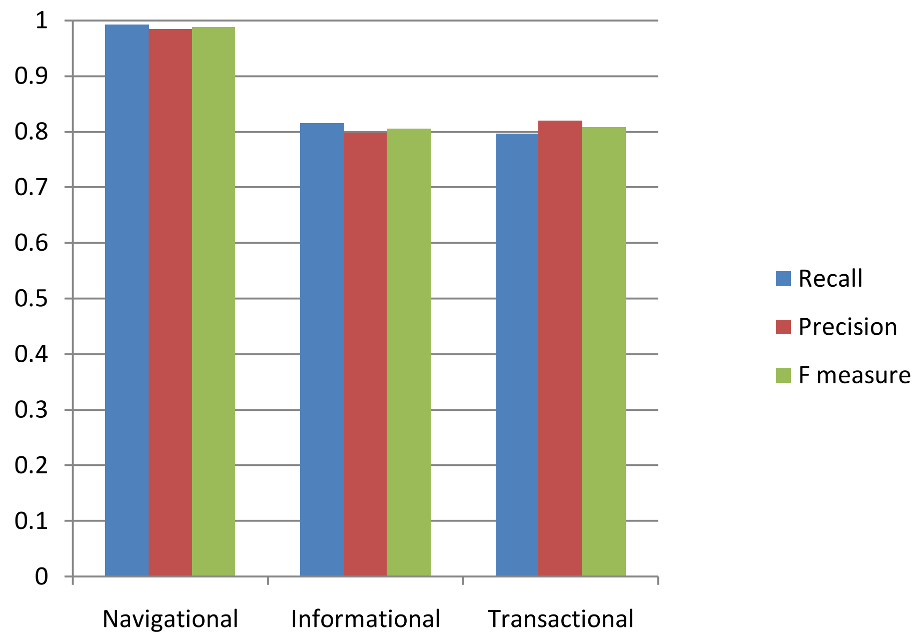

Figure14: Classification accuracy across classes using J48 Algorithm

Navigational	Informational	Transactional	← Classified as
126	0	1	Navigational
0	**75**	17	Informational
2	19	**82**	Transactional

Table 8: Confusion Matrix for using J48 Algorithm

Random Forest Algorithm

90.0621% 10 fold cross validation accuracy is achieved using the Random Forest algorithm.

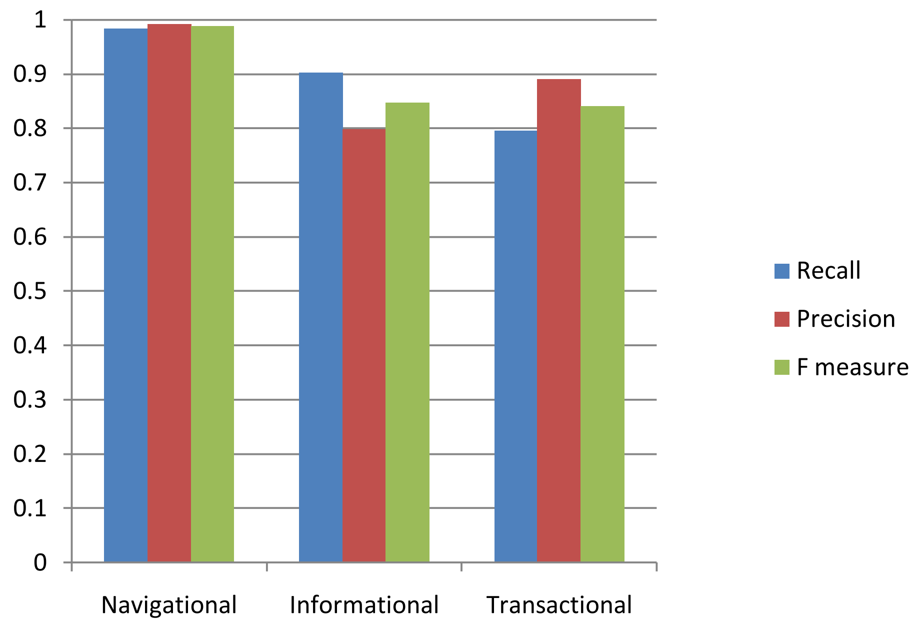

Figure 15: Classification accuracy across classes using Random Forest Algorithm

Navigational	Informational	Transactional	← Classified as
125	1	1	Navigational
0	**83**	9	Informational
1	20	**82**	Transactional

Table 9: Confusion Matrix for using Random Forest Algorithm

SMO Algorithm

84.7826% 10 fold cross validation accuracy is achieved using the SMO algorithm

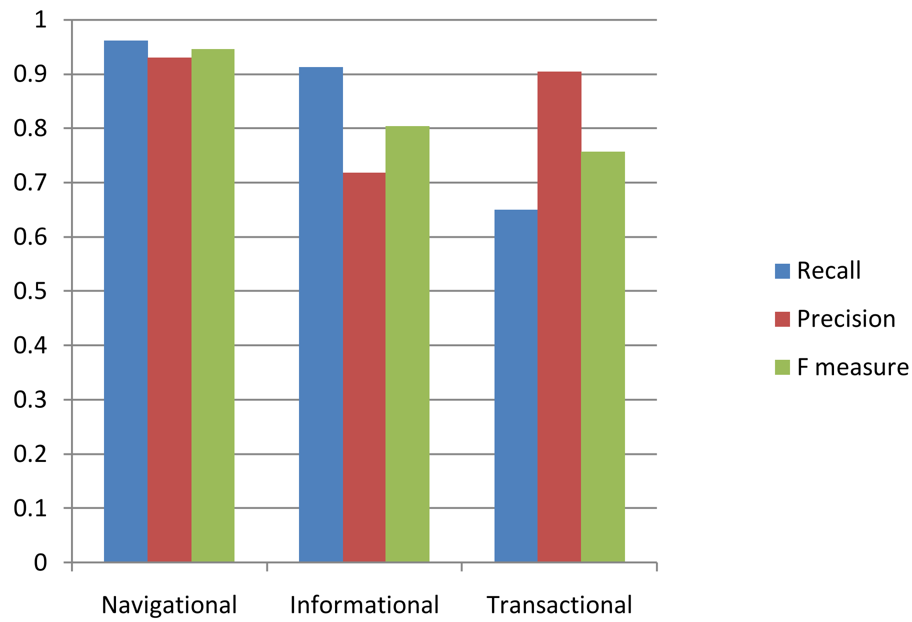

Figure 16: Classification accuracy across classes using Random Forest Algorithm

Navigational	Informational	Transactional	← Classified as
122	5	0	Navigational
1	**84**	7	Informational
8	28	**67**	Transactional

Table 10: Confusion Matrix for using SMO Algorithm

RandomCommitte Algorithm

91.3043% 10 fold cross validation accuracy is achieved using the SMO algorithm

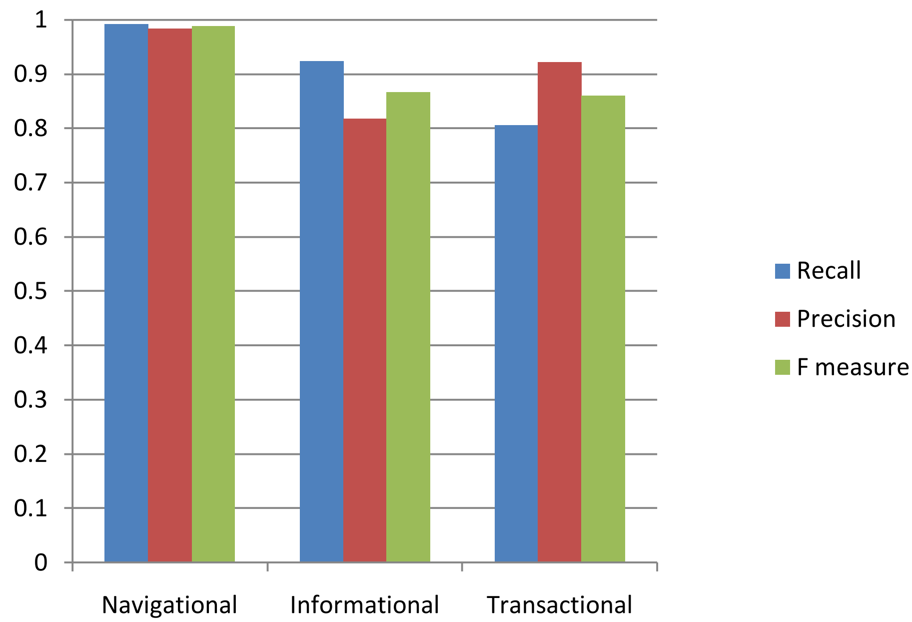

Figure 17: Classification accuracy across classes using Random Forest Algorithm

Navigational	Informational	Transactional	← Classified as
126	0	1	Navigational
1	**85**	6	Informational
1	19	**83**	Transactional

Table 11: Confusion Matrix for using RandomCommittee Algorithm

Comparison of different classification algorithms

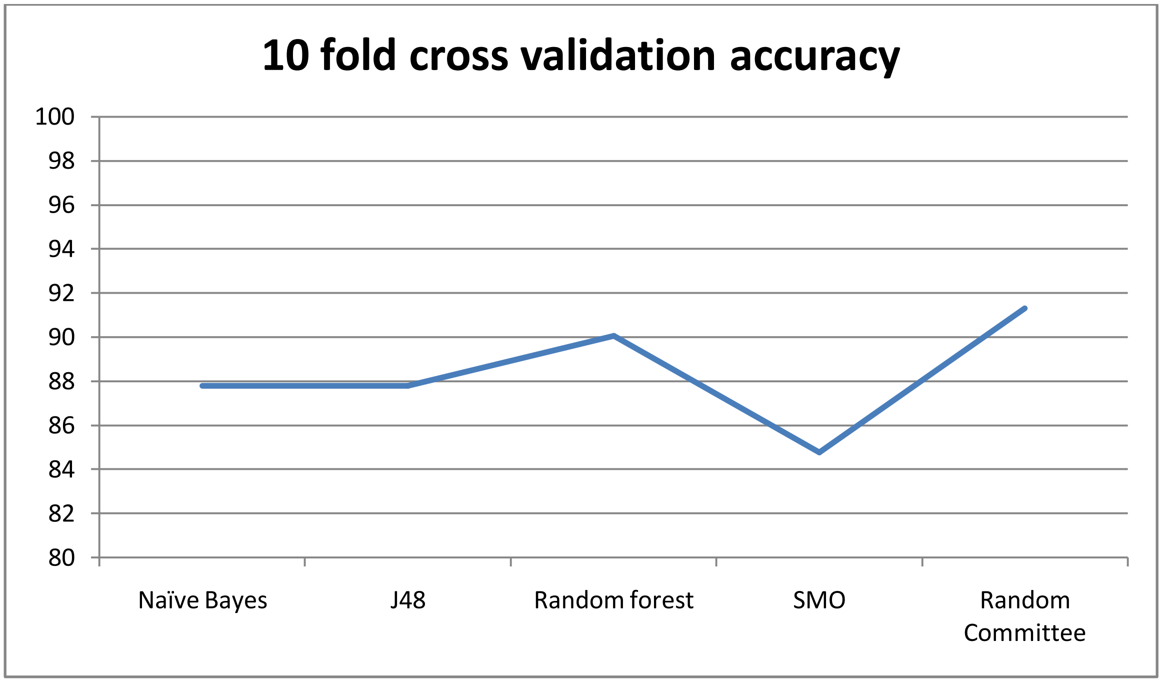

Figure 18: 10 fold cross validation accuracy for different classification algorithms

4.5 Feature Analysis

4.6 Conclusion

Chapter 5

Automatic Classifier for Queries

Once the classifier is ready, we can continue with the automatic classification of our queries. We test our query classification algorithm on the benchmark set of queries that have been manually classified by 30 users. The web pages corresponding to the click-through urls of the queries are downloaded and classified by the RandomCommittee Meta classifier built. The approach to classify the queries is simple. For each query, we check how many users (clicks) have visited navigational pages, how many have viewed transactional pages and how many have viewed informational pages. While counting the navigational pages, it has to be kept into consideration that there exists only one correct navigational page for a query. Hence, the navigational page with maximum clicks is taken to be navigational clicks whereas the clicks for other navigational pages are added to the transactional clicks. If a clear majority exists for a particular class type, the query is said to be having that user goal else the query is termed to have no predictable goal.

5.1 Algorithm for Automatic Query Classification

Following are the steps of the algorithm used to classify a query into navigational, informational, transactional or ambiguous query:

1. For each query, classify each click-through result into three classes: navigational, informational or transactional
2. Count the number of informational and transactional clicks for the query
3. For the navigational results, compare the domain name of the website to compare the similarity. If they are similar, add their counts into one
4. For the navigational results, the navigational result with the maximum clicks is taken to be the navigational representative. Other navigational clicks are added to transactional clicks for the query
5. The belongingness value for each class is calculated by dividing the number of clicks for each class with the total number of clicks for the query
6. The class with maximum belongingness value and the one with 2^{nd} maximum belongingness value are chosen and the difference d between them calculated. If d is greater than a threshold value t, the query is classified to belong to the class with maximum belongingness value else it is termed ambiguous with belonging to both the maximum and 2^{nd} maximum classes. Various values of threshold are experimented with and the value chosen for t is finally .2

Following table shows the click-through and classification of corresponding click-through pages for the query 'Microsoft'.

Query	Click-through url	Clicks	Class of Web Page
Microsoft	http://www.microsoft-watch.com/	1	N
Microsoft	http://windowsupdate.microsoft.com/	256	N
Microsoft	http://office.microsoft.com/	38	N
Microsoft	http://microsoft.com/	600	N
Microsoft	http://toolbar.msn.com/desktop/results.aspx	1	I
Microsoft	http://www.joewein.de/sw/joewein.htm	2	I
Microsoft	http://adcenter.looksmart.com/security/login	2	I
Microsoft	http://terraserver.microsoft.com/image.aspx?PgSrh:NavLon=86.405&PgSrh:NavLat=32.73694	2	T
Microsoft	http://connect.microsoft.com/onenote	2	I
Microsoft	http://www.lindqvist.com/en/el-gordo-de-la-primitiva-lottery-international-promotions-programmes	1	T
Microsoft	http://www.symantec.com/security_response/writeup.jsp?docid=2000-122015-2522-99	1	I
Microsoft	http://moneycentral.msn.com/investor/home.asp	1	I
Microsoft	http://messenger.msn.com/Resource/Emoticons.aspx	1	T
Microsoft	http://support.microsoft.com/	104	N
Microsoft	http://research.microsoft.com/aboutmsr/labs/cambridge	1	I

Table 12: Click-through and classification information for query 'Microsoft'

As we can see, the navigational pages http://microsoft.com/, http://www.microsoft-watch.com/, http://windowsupdate.microsoft.com/, http://office.microsoft.com/ and http://support.microsoft.com/ have the same domain name and hence their clicks are added up into one navigational page's clicks. By summing up we find that total navigational clicks are 999, transactional pages are 4 and informational pages are 10. The belongingness values in navigational/transactional/informational are respectively 0.986(999/1013), 0.004(4/1013) and 0.010(10/1013). Hence the query is classified as navigational with the difference between the max class(navigational) and 2nd max class(informational) is >.20

5.2 Results

Now we present the automatic classification results in the order they were presented in the manual classification results section. The queries not classified correctly are analyzed and the reason behind the wrong classification presented. The comparison between the manual classification and the automatic classification results are presented in the appendix.

Navigational Queries

Out of the 15 navigational queries, all were detected to be navigational by our query classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)	Predicted Type
Hotmail	0.924	0.039	0.037	Navigational
Google	0.971	0.005	0.025	Navigational
Espn	0.801	0.02	0.179	Navigational
Imdb	0.932	0.006	0.062	Navigational
Honda	0.782	0.11	0.207	Navigational
Yahoo	0.975	0.025	0.005	Navigational
Ask	0.924	0.009	0.068	Navigational
Amazon	0.926	0.005	0.069	Navigational
Thesaurus	0.792	0.068	0.140	Navigational
Suzuki	0.726	0.258	0.726	Navigational
Microsoft	0.986	0.010	0.004	Navigational
Encyclopedia	0.698	0.256	0.046	Navigational
Dell	0.611	0.080	0.310	Navigational

Pogo games	0.634	0.072	0.294	Navigational
Ebay	0.984	0.001	0.015	Navigational

Table 13: Automatic classification results for Navigational Queries

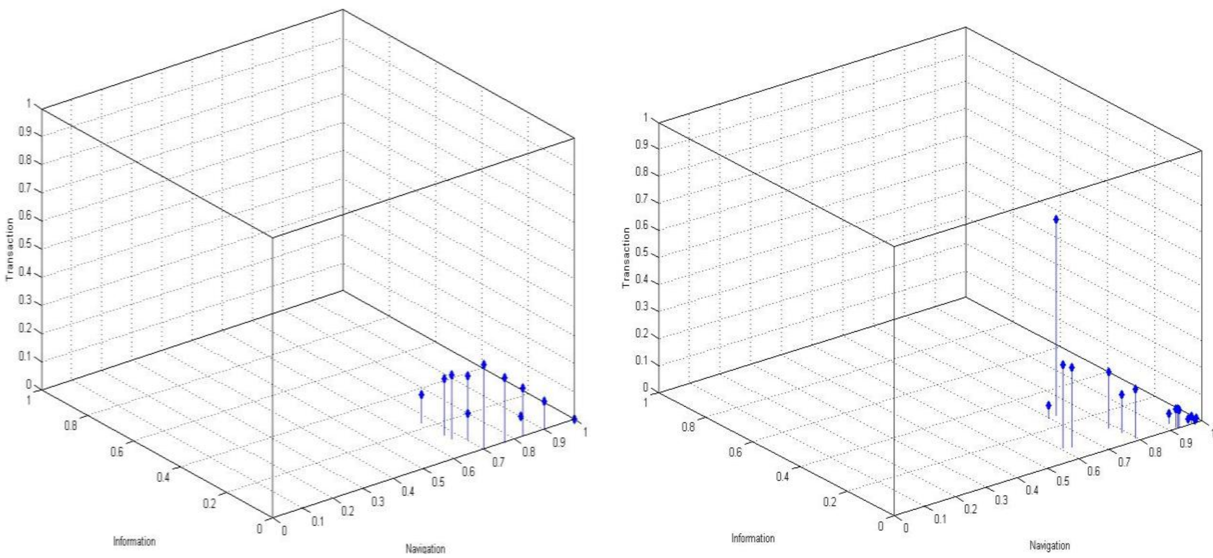

Figure 19: Automatic classification vs. manual classification of navigational queries

Transactional Queries

Out of the **19 transactional queries**, **18** were correctly identified as transactional by our classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)	Predicted Type
Mortgage Calculator	0.005	0.011	0.984	Transactional
Myspace Layouts	0.252	0.009	0.739	Transactional
Tattoos	0.203	0.273	0.524	Transactional
Cigarettes	0.111	0.333	0.556	Transactional
Funny Pictures	0.317	0.019	0.665	Transactional
Free music downloads	0.157	0.319	0.524	Transactional
Msn messenger	0.034	0.000	0.996	Transactional

Free ringtones	0.252	0.214	0.535	Transactional
Download	0.238	0.629	0.132	**Informational**
Ipod	0.095	0.047	0.858	Transactional
Screensavers	0.057	0.343	0.600	Transactional
Netscape	0.234	0.012	0.753	Transactional
Deal or no deal	0.001	0.041	0.958	Transactional
Shoes	0.064	0.242	0.694	Transactional
Airsoft guns	0.143	0.218	0.639	Transactional
Aol media player	0.000	0.048	0.952	Transactional
Itunes	0.007	0.050	0.943	Transactional
Internet explorer	0.000	0.021	0.979	Transactional
Sudoku	0.300	0.152	0.548	Transactional

Table 14: Automatic classification results for Transactional Queries

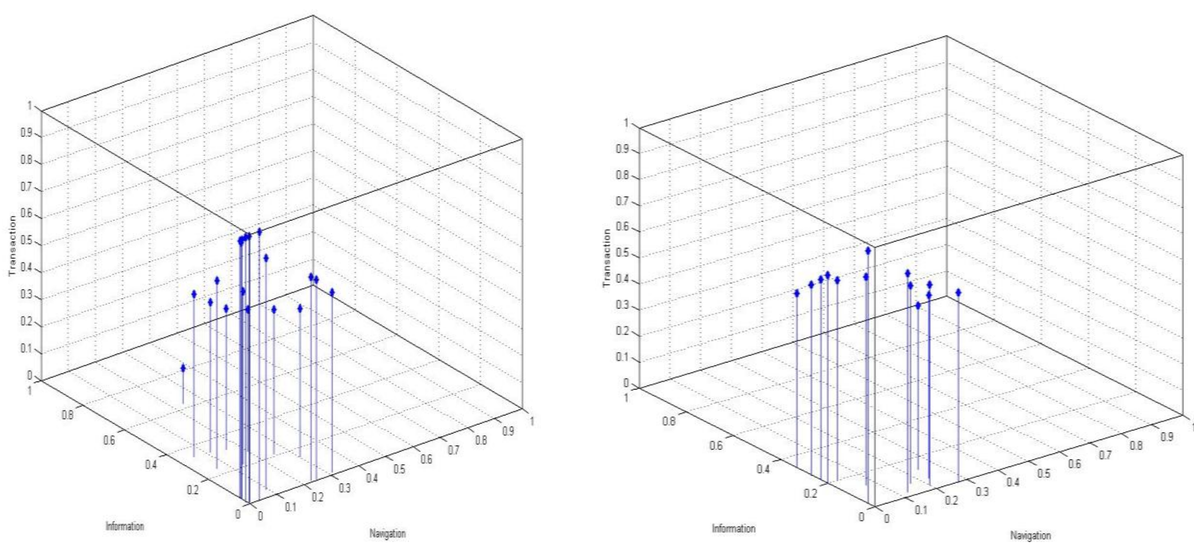

Figure 20: Automatic classification vs. manual classification of navigational queries

Informational Queries

Out of the **19 Informational queries**, **11** were correctly identified as informational by our classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)	Predicted Type
Kidney stones	0.059	0.715	0.226	Navi / Informational
Bird flu	0.509	0.366	0.125	**Informational**
Employment	0.024	0.847	0.129	Informational
Motorcycles	0.057	0.665	0.278	Informational
Html	0.045	0.829	0.126	Informational
Pregnancy	0.183	0.510	0.183	Informational
Snakes	0.141	0.772	0.087	Informational
Optical illusions	0.135	0.723	0.142	Informational
Exe	0.009	0.846	0.145	Informational
Guns	0.177	0.622	0.201	Informational
Florida lottery	0.186	0.707	0.106	Informational
Airline tickets	0.018	0.618	0.365	Informational
Anna benson	0.002	0.130	0.868	**transactional**
Jessica simpson	0.014	0.324	0.662	**transactional**
Paris Hilton	0.004	0.373	0.622	**transactional**
Baby names	0.319	0.011	0.670	**transactional**
Jessica alba	0.279	0.420	0.301	**Info/transactional**
Kelly blue book	0.000	0.174	0.826	**transactional**

Recipes	0.097	0.154	0.749	**transactional**

Table 15: Automatic classification results for Informational Queries

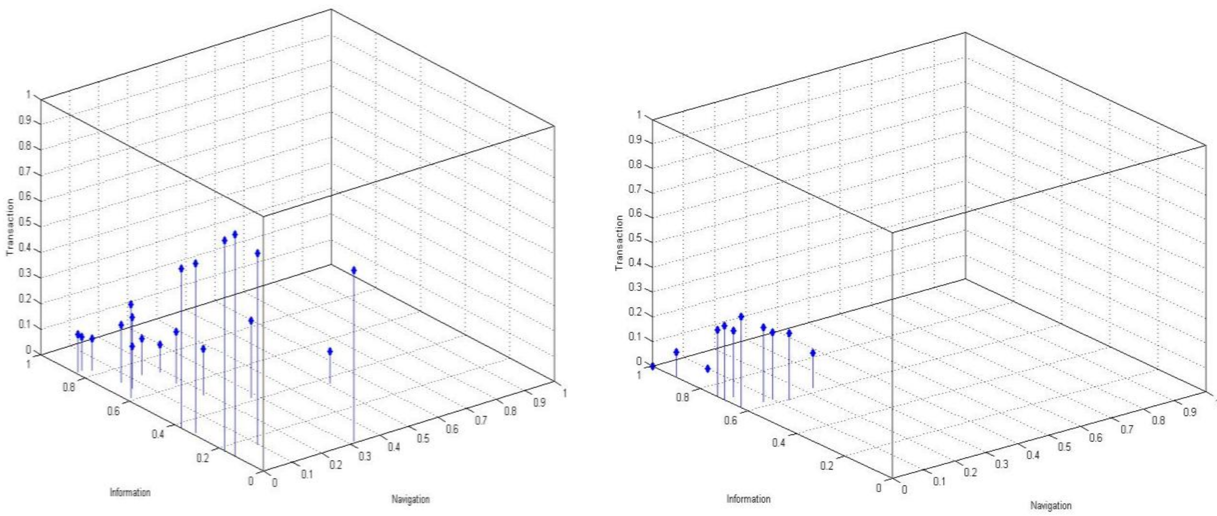

Figure 21: Automatic classification vs. manual classification of navigational queries

Informational-Transactional Queries

Out of the **10 informational-transactional queries, only 2** were detected to be so by our classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)	Predicted type
Furniture	0.112	0.391	0.498	Info / transactional
Online games	0.112	0.369	0.519	Info / transactional
Costa rica	0.338	0.053	0.609	Transactional
Britney spears	0.003	0.213	0.784	Transactional
Shakira	0.627	0.323	0.049	Navigational
Kelly Clarkson	0.139	0.198	0.663	Transactional
Reverse lookup	0.003	0.962	0.035	Informational

David blaine	0.522	0.064	0.413	Navi / Transactional
Movies	0.435	0.201	0.364	Navi / Transactional
Cars	0.307	0.279	0.414	Navi / Transactional

Table 16: Automatic classification results for Informational-Transactional Queries

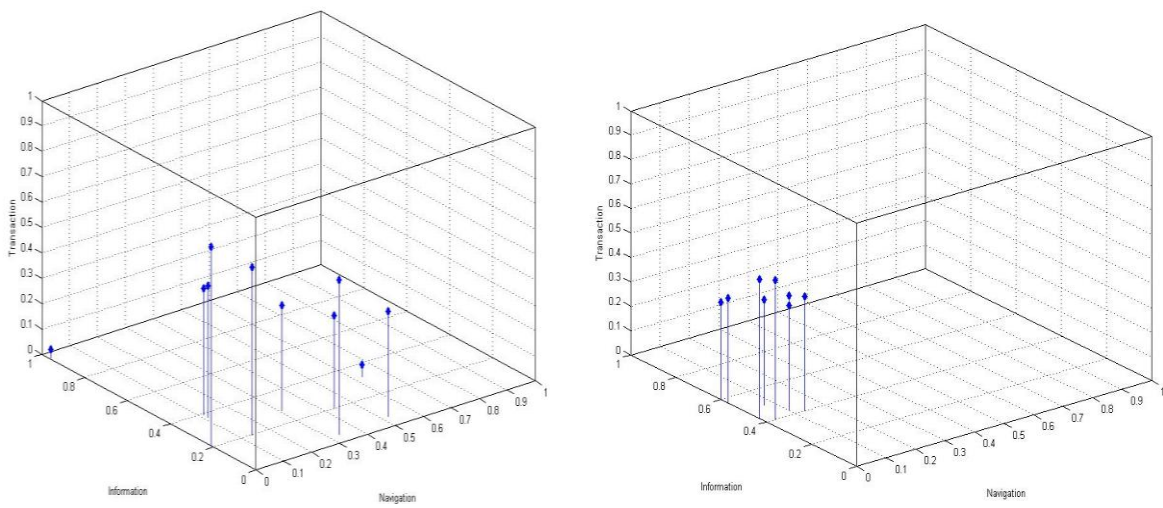

Figure 22: Automatic classification vs. manual classification of navigational queries

Informational-Navigational Queries

Out of the **only navigational-informational query,** it was detected to be so by our classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)
Harry Potter	0.347	0.406	0.247

Table 17: Automatic classification results for Informational-Navigational Queries

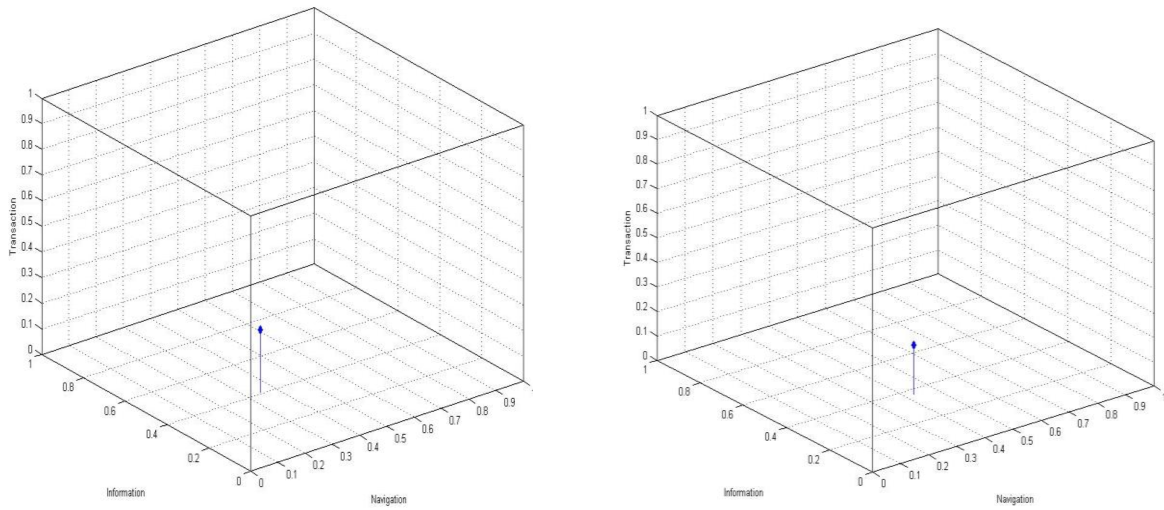

Figure 23: Automatic classification vs manual classification of navigational queries

Transactional-Navigational Queries

Out of the **only navigational-transactional query,** it was detected to be so by our classification algorithm. The respective belongingness values for the queries for various classes are presented in the following table:

Query	N(q)	I(q)	T(q)
Bible	0.424	0.126	0.450

Table 18: Automatic classification results for Transactional-Navigational Queries

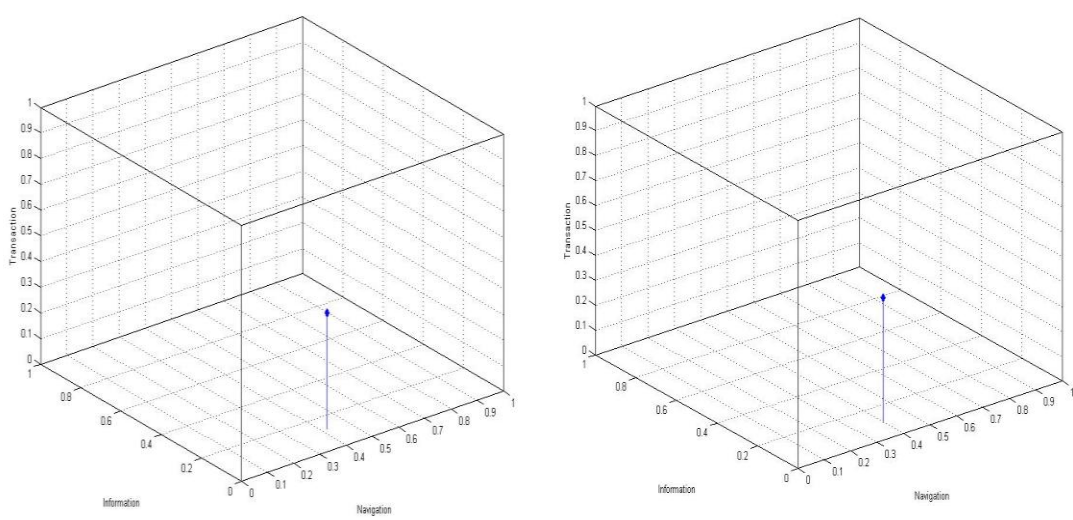

Figure 24: Automatic classification vs. manual classification of navigational queries

5.3 Analysis of Misclassified Queries

5.4 Conclusion

Chapter 6

Search Engine Design Based on User Goals

6.1 First Generation Search Engines

6.2 Second Generation Search Engines]

6.3 Third Generation Search Engines

Chapter 7

Conclusion