I give permission for public access to my thesis and for any copying done at the discretion of the archives librarian and/or the College librarian

June 25, 2007

……………………….
Carolyn T. Hafernik

Automatic Methods To Disambiguate Geospatial Queries

A Thesis Presented

by

Carolyn Hafernik

This thesis was prepared under the guidence of

Professor Lisa Ballesteros

Submitted to the Faculty of Mount Holyoke College

in partial fulfillment of the requirements

of the degree B.A. with Honors in Computer Science

South Hadley, MA

May, 2007

ACKNOWLEDGMENTS

I would like to thank all of the people who helped and encouraged me with this thesis.

First, I would like to thank my advisor Professor Lisa Ballesteros for all of her time, encouragement, support and supervision during this research project. I especially appreciate her encouraging me to do research.

I am also grateful to my committee members Professor Claude Fennema, Professor Xiaoyan Li, and Professor Frederick McGinness for their support, encouragement and suggestions.

In addition, I would like to thank my classmates and friends who took an interest in my research project and gave me their support while I was working on this thesis.

Lastly, I would like to thank my parents for always encouraging and supporting me.

ABSTRACT

Today an unprecedented amount of digital information is available, but locating information of interest can be difficult. Information Retrieval (IR) is the area of Computer Science that aims to locate information by automatically organizing, storing, and managing it. IR systems search large databases, separating non-relevant items from the relevant items, and return a list of the documents likely to match the information need as described by the user's query. Geographical Information Retrieval (GIR) aims to develop IR systems with spatial awareness and exploit geospatial information to improve retrieval effectiveness. This research explores GIR, as in GeoCLEF, and uses geospatial information to automatically disambiguate geospatial terms. The hypothesis is that automatically disambiguating geospatial terms will improve retrieval effectiveness.

Two challenges to IR and GIR are language ambiguity and improving retrieval automatically, instead of manually, by query modification. Language ambiguity can be problematic for both queries and documents because there are many ways to describe concepts or ideas. Separate documents describe the same information differently. Similarly, independent users looking for the same information may use different words in their queries. A query describing a concept in one way will fail to retrieve relevant documents that use different vocabulary. Users often fail to precisely specify information they require, which may cause the system to miss some important relevant documents. Manual approaches to query modification require a person to determine other concepts and words which not only better describe the needed information, but that others may have chosen for the same topic. This is not realistic in a real world situation. Often there is not enough time for individuals to modify the queries themselves and they might not know how to improve queries. Thus automatic methods, which can be done without a human, are needed in order for IR or GIR methods to be practical.

This work aims to exploit geospatial information in queries to improve retrieval by automatically disambiguating geospatial terms within the queries using outside geospatial knowledge gathered from the internet, including city names, countries, regions, parts of countries and location information. Our approach combines simple linguistic analysis with query modification via the addition of geospatial information. Geospatial terms were chosen in several different ways. First, terms were added from retrieved documents assumed to be relevant. Another method gave higher weight to more important query words. A third procedure added terms selected from a geographic thesaurus. Finally, attempts were made to perform spatial disambiguation by using longitude and latitude to infer an upper bound on distance terms like “near.”

TABLE OF CONTENTS

Chapter **Page**

3. Methods 58

LIST OF FIGURES AND TABLES

Figures **Page**

Tables **Page**

CHAPTER 1: INTRODUCTION

1.1 Searching for Information

Today there are unprecedented amounts of digital information available to users. The information can come in many forms: text, images, videos, audio or a combination of these. The information is spread out over the web or may be contained in proprietary databases. With all of the information available, it is particularly challenging to find the specific information that meets one's needs. Information is often readily accessible, but locating information in which one is interested can be more difficult.

One example of the density of electronic information is that accessible via the internet. There is so much data on the web that websites devoted to searching the internet, like Google, have developed. Search engines take queries and return results. Textual queries are the way a user describes the information he or she needs. The results in this case are web pages that the search engine infers contain the information the user's query described. For instance, a user might type in the query "Mount Holyoke College" hoping to find information about Mount Holyoke College. In this case, Google or any other search engine would return a list of pages that it found included the words "Mount Holyoke College." Still there is no guarantee that the list would contain useful results and the user may have to go through pages of results to find the desired information.

This thesis deals with the emerging research area of Geographical Information Retrieval (GIR). It attempts to find ways to exploit geospatial information within queries and geospatial information available on the internet to modify queries so that retrieval will be improved.

Section 1.2 introduces the research area of Information Retrieval (IR), how documents' relevance is determined, how IR systems are evaluated and the area of Geographic Information Retrieval (GIR). Section 1.3 examines the use of stop words. Section 1.4 examines how terms are indexed in GIR. Section 1.5 examines word mismatch and query specification challenges. Section 1.6 examines language ambiguity and its effect on GIR. Section 1.7 discusses different methods of query expansion such as manual and automatic expansion, blind feedback, Local Context Analysis (LCA), and thesauri expansion and locating information for query expansion. Section 1.8 explores re-weighting terms in the query. Section 1.9 describes my approach and section 1.10 gives an overview of the layout of the rest of the thesis.

1.2 Information Retrieval (IR)

Information Retrieval (IR) is the area of Computer Science in which the goal is to organize, store, and manage information to make it easier to search automatically. IR systems search large databases, weeding out the non-relevant items from the relevant items, and return a list of the documents that are most likely to match the information described in the user's query. Relevant documents are those that provide information that the user is seeking.

One of the most difficult parts of IR is inferring which documents are relevant and which are not. Recall that queries are the user's method of informing a search system of what information they are seeking. Relevant documents will thus be documents that provide the information that the user is looking for. In traditional IR systems, the system infers which documents are relevant by matching query words to words in the documents. IR systems often use a "bag of words" approach. This assumes that every word in the query is independent from the others. Traditional IR uses these words to get a sense of the "aboutness" of documents but does not generate or analyze a linguistic sense of the meaning of documents or queries.

One common weighting scheme for words is TF.IDF, which uses term frequency (TF) and inverse document frequency (IDF). TF takes into account how many times a word appears in a document and IDF takes into account how often a term appears over the entire collection of documents. TF is important because we assume that the more times a word appears in a document the greater the likelihood that the word describes what the document is about. On the other hand, for IDF we assume that words which appear in many documents are less helpful than words that appear in fewer documents. Words that appear in many documents therefore have little discrimination power because they do not separate relevant from non-relevant documents. An IR system often ranks documents based on the TF.IDF scores for query words in documents. TF rewards a term for high frequency in a document; IDF penalizes the terms that occur frequently across the collection. The document score is a function of the TF.IDF scores for all of the words in the query.

To evaluate IR systems, retrieval is performed on test collections. Each test collection consists of a set of queries and a collection of documents for which there are relevance judgments indicating which documents are relevant for each query. Relevance judgments enable a system to be evaluated to see how effective retrieval is by comparing the list of documents returned by the system to the list of documents known to be relevant to the query. This is necessary because it allows the evaluator to tell whether results returned by a search engine are relevant without having to read every document and subjectively decide which documents are relevant. Nevertheless, the lists of relevant documents for test collections are normally created manually by human experts reading documents and assessing relevance.

IR systems are typically evaluated based on precision and recall. Precision measures the proportion of retrieved documents that are relevant and gives a sense of how successful the system is at separating relevant and non-relevant documents. High precision tells the evaluator that the system succeeded in returning less garbage. Recall measures the proportion of all relevant documents that were successfully retrieved. Recall shows whether the system succeeded in returning many of the relevant documents that it could have returned. There is a trade-off between precision and recall. On one hand, one wants to get high precision so that users do not have to wade through too much junk, but the cost of this is lower recall. This means users might miss some of the most useful documents. Some applications, like law searches, need high recall; others need high precision; and others want a balance between the two. Additionally, statistical tests such as the signtest can be performed to see whether or not a difference in results is significant.

There are many different types of IR systems. Each type deals with a different type of retrieval. Some systems retrieve text documents; others retrieve multimedia documents, such as images or videos. My thesis will focus on text retrieval and in particular Geographical Information Retrieval (GIR), a new research area aiming to develop IR systems with spatial awareness. The goal of a GIR system is to exploit geospatial information to improve retrieval effectiveness. It faces many of the same challenges as traditional IR plus other problems due to geospatial information.

1.3 Stop words

One technique often used in IR is the removal of stop words from the queries and documents. Stop words are words that appear commonly in documents. Stop words include common terms such as “the,” “and,” “of,” and “in.” They are removed because most, if not all, of the documents have these words in them. Thus the words are not very helpful in dividing relevant and non-relevant documents because they appear in both. In addition, the words do not contain context that will help find the relevant documents. Similarly, stop phrases are often removed from queries. These are phases that are common in the queries but will most likely not be in relevant documents. For instance, “relevant documents will contain” is a stop phrase that could be removed, because relevant documents most likely will not have that in them.

GIR is different from traditional IR in that geospatial relationships within the queries could be important for retrieving the relevant documents. Traditional IR treats the words in the query as independent of each other. GIR sometimes believes connections between the words are important. Needing to consider the connections between words is one of the challenges of GIR. This can also be important for traditional IR, but it may be especially important for GIR. Also, many of the words that specify geospatial relationships like “near” or “in” are considered stop words in traditional IR. This means they are removed from the query and not used to do retrieval. For GIR, the stop word lists might need to be modified to take into account words that would generally be stop words but are necessary for GIR. Another possible solution would be to do analysis of the query prior to processing to taking into account these possible stop words. GIR may benefit from exploiting these words for geo-analysis because they describe the geospatial relationships within the query. Geo-analysis could help to identify the geospatial relationships within the query and terms that might be useful for disambiguating the geospatial terms.

1.4 Indexing

The documents within the test collection are indexed. This index allows fast searching over large volumes of data. Normally stop words would not be indexed. This means they would not be used in searches. One way to do indexing is single word indexing. This means that every word is treated individually and independently. Often times before being indexed terms are stemmed. Stemming reduces all words with the same root to a single form [35]. Often only suffixes are removed. For example the words “book” and “books” would both be stemmed to the same root, “book.”

Another way of detecting word variants is by calculating a string similarity measure between the query term and each distinct term in the index. One common approach to this is n-gram coding, which fragments a word into a sequence of n-grams. It breaks the terms into strings of n adjacent characters and then estimates the similarity between two sets of n-grams [35]. One common implementation of n-grams is bi-grams, which divides the terms into strings of 2 adjacent characters.

1.5 Word Mismatch and Query Specification

One problem for IR and GIR is the word mismatch problem. Different documents use different words to describe the same information. This is difficult to deal with because the words in the documents cannot be changed. Similarly, different users looking for the same information may use different words in their queries. For instance, one user might type “San Francisco restaurants” and another type “Bay Area restaurants.” Both queries are looking for the same information, but the users chose different words and so might get completely different results. Additionally, due to poor query specification, it is often necessary to use information that is not provided in the query. Poor query specification occurs when the user does not specify the information they are looking for as precisely or specifically as they might have. A query might be poorly specified because the user knows only a few relevant terms describing her information need or because she used a more general term such as “U.S.” instead of a more specific term “California.” In addition, there are many ways to describe concepts or ideas. If only one way of describing a concept is used to query the system, the system will fail to retrieve relevant documents that describe the concept differently. A poorly specified query may lead the system to miss some important relevant documents that have different terms than the query. For instance, if a user entered a query mentioning “U.S.” the user would get documents with the term “U.S.”, but not documents that only contain names of states in the U.S. and not the term “United States” or “United States of America.”

1.6 Language Ambiguity

One challenge to finding relevant documents is the ambiguity of language. Language ambiguity refers to the fact that words or phrases can have several different meanings and thus be interpreted in different ways depending on their context. Ambiguity of language can be problematic in both queries and documents. One common cause of ambiguity is the use of homonyms or words that are the same but mean different things. For instance, the word “book” could refer to an object to read or to reserving a ticket of some sort. This ambiguity can cause the system to retrieve documents containing words used in a different context than that meant by the query.

Like IR, GIR is made more difficult by language ambiguity. Geospatial words can be ambiguous. Often it is hard to tell exactly what a query is intended to convey. For instance, in the query “bookstores near San Francisco,” the geospatial terms would be “near” and “San Francisco.” In this case, the system needs a way of defining what is “near” and what is not “near.” For instance, in the San Francisco case, Berkeley and Oakland would be near but Los Angeles or San Diego would not be. Additionally, locations could be ambiguous. One could have two locations with the same name in different places (Concord, Ca and Concord, Ma). One possible solution to this is to use geographical disambiguation to disambiguate words in the query. For the Concord example, one might look at other words within the query and see which Concord they were more likely to be referring to. Geographical disambiguation would use other information to decide to which of the locations the query is most likely referring. In the case of two places with the same name, looking at the other words in the query might give one an idea of which one the query was referring to or one could use blind feedback (discussed in section 1.7.3) to expand the query.

Geospatial information may be used to disambiguate geospatial terms. Disambiguating the queries involves making them more precise. This allows them to give more guidance to the system on what a relevant document might contain. Disambiguating words would make a query clearer. Words with no clear definition would include words like “near” and “far.” Different people will likely have different definitions of “near”. One might say “near” is within 10 miles, another that “near” is within 50 miles. The computer, on the other hand, has no knowledge of the meaning of the word.

One could use geospatial information to try to deal with the ambiguity of words like “near.” By looking at the longitude and latitude of locations, one can potentially discover which locations are closer to the locations in the query. This information about which locations are closer to the locations in the query can then be used to add locations that fulfill the correct spatial relationship to the query. In the “hiking in the Bay Area” example, the information about proximity would be used to add locations, which fell within a certain distance of the “Bay Area” to the query.

As mentioned above, one possible way to improve GIR is to disambiguate the geospatial terms within the query using geospatial information. Some examples of geospatial information that could be used are locations, the distance between places, longitude and latitude, and the country or regions where a place is located. There are several techniques that attempt to disambiguate queries including query expansion, which add words that describe the ambiguous query terms more exactly (See section 1.7) and re-weighting of query terms (See section 1.8).

1.7 Query Expansion

The challenges of queries and the ambiguity of language have led to approaches for improving IR techniques that use information not originally in the query. One common way of including extra information in a query is to use query expansion. Query expansion aims to reduce the likelihood that the query would fail to retrieve relevant documents because of a mismatch in the words in the query and the documents. It does this by adding words or phrases with similar meanings or other relationships to the words in the query and in the set of relevant documents. The idea is that the expanded query will match more of the words contained in relevant documents and thus retrieve more relevant documents. These extra terms are often found in dictionaries, thesauri or in documents that are believed to be relevant. One way to do query expansion would be to add synonyms of the query words.

Below section 1.7.1 addresses finding geospatial information that could be used for expansion. Section 1.7.2 is a discussion of the advantages and disadvantages of automatic and manual query expansion. Following that sections 1.7.3, 1.7.4 and 1.7.5 examine specific methods of query expansion: blind feedback, local context analysis and thesaurus expansion.

1.7.1 Locating Geospatial Information to Use in Expansion

One of the challenges of using geospatial information in IR is that there is no central repository for geospatial knowledge; information can be found scattered all over the place. There are many different kinds of geospatial information on the web, such as information on different names for the same place, population figures, the distance between different places and the location of a place. For example, for a city there is information on where it is in terms of longitude and latitude, what country it is in, variations in the way to which it is referred (for example, Los Angeles and LA are the same place), its population, and nearby landmarks. Some information such as name variants or landmarks can be used for query expansion in order to disambiguate queries. For example, if one was looking for “hiking trails in the Bay Area” the addition of “San Francisco” and “Muir Woods” would generate a more specific query. In this case, locations identified as part of the “Bay Area” could be used for disambiguation. Furthermore, other facts about the “Bay Area” such as other names for it, population statistics, and nearby landmarks might also be used. Using this knowledge, the system would find documents related to hiking and the “Bay Area” more easily than it would without the geospatial information it received from the gazetteer.

Looking on the web there are many sites that provide geospatial information, but they do not all provide the same information. One of the first tasks of doing GIR is to decide where the geospatial information one is going to use will come from. The next step is to gather this information into one place so that the GIR system can find and use the information for retrieval. Finally, it must be determined how best to use the information because no one knows exactly how it should be used, which makes GIR even more difficult.

1.7.2 Automatic versus Manual Expansion

There are two types of query expansion: manual expansion and automatic expansion. Manual expansion involves a person choosing which terms should be added to the query. This can be useful because a person might be able to choose more relevant words to add, but it also has many disadvantages. For instance, it requires a person to look at the queries. This requires time. Manual expansion can not be done quickly because a person must be located and then must choose words. This is a problem because users may not want to wait for a person to manually expand a query and because there might not always be a person who can expand the queries. The user could choose her own words, but that assumes that she is willing to spend extra time modifying the query and know what words should be chosen. Also, manual expansion assumes that the person expanding the queries has enough expertise to choose relevant words. In many cases the person might not know what words should be chosen or might even choose words that hurt retrieval when added. A different person might be needed to expand every query because one person could not have the expertise to expand all topics. So, manual expansion, though it sometimes achieves better results than automatic expansion, is typically unrealistic for IR systems to use on a regular basis. Ideally expansion is done automatically.

Automatic expansion is when the computer does the expansion without a human looking at the queries and choosing what words to add. Thus automatic expansion is much quicker than manual expansion. It is also more realistic since it does not require human intervention. Automatic expansion has been shown to work as well or better than manual expansion. One challenge of automatic expansion is that it requires some sort of method for finding relevant words to expand queries with. There are several sections below discussing methods of automatic expansion including blind feedback, Local Context Analysis (LCA), and thesauri expansion.

1.7.3 Blind Feedback

One way to do query expansion is to use blind feedback. This is where the original query is used to retrieve a set of documents and the top n documents are assumed to be relevant. Words that are found to occur across relevant documents are ranked and the top m words are then added to the query [35]. The added terms will hopefully make the query more precise and enable the system to retrieve more relevant results. For instance, if the query is asking about "Shark Attacks Off California and Australia" then adding terms such as the names of the oceans near California and Australia and the names of major cities on the coasts would specify the query more clearly and hopefully retrieve more relevant results. The effectiveness of blind feedback is dependent upon the number of relevant documents available for the query and the choice of n and m.

The benefit of blind feedback is that it can be done automatically without human intervention, which means it is realistic to be able to perform it on queries without users getting bored and leaving before the IR system returns results for them. On the other hand, blind feedback can also, in some cases, hurt retrieval. If the original formulation of the query does not retrieve many relevant documents then potentially blind feedback will be adding terms from irrelevant documents to the query, which will likely hurt retrieval. So, blind feedback requires that the designer choose n and m wisely so as to minimize the number of non-relevant documents looked at and the number of non-relevant terms that are added to the query.

1.7.4 Local Context Analysis

Another query expansion technique is Local Context Analysis (LCA) [37]. LCA uses both global and local analysis. Global analysis techniques examine word occurrences and relationships over the entire corpus or collection of documents. Local analysis explores word occurrences in only the top ranked documents retrieved by a query and is designed to exploit context. Both global analysis and local analysis have advantages and disadvantages. Global analysis is more expensive than local analysis due to the number of computations required. On the other hand, experiments done with small test collections have not been promising for local analysis [37]. Small test collections are not realistic, but the results on them can be important because the methods may not do as well on a larger collection as on a smaller one. The simple version of local analysis adds words from the top-ranked passages of documents retrieved for the original query. How effective this technique is depends on the proportion of relevant documents in the top documents retrieved. This means that queries that perform poorly without added terms will most likely perform even worse after local feedback. This is because if a query is performing badly to start with, adding terms from the top ranked passages will not help much because those passages are less likely to be relevant.

LCA uses ideas from global analysis, such as context and phrase structure, and applies these to a local document set instead of the entire collection of documents as in global analysis. In local context analysis, concepts or groups of terms are selected for query expansion based on co-occurrence with query terms. Co-occurrence measures the frequency with which terms occur together throughout the corpus. These concepts are chosen from the top ranked documents using the best passages instead of entire documents. This means that the most relevant portions of the documents can be used to choose concepts instead of an entire document of which only parts are relevant. Local context analysis has several advantages. It is computationally practical. Once the ranked passages are available, query expansion is fast and does not filter out frequent concepts. This means that concepts that occur in many passages can still be chosen to expand a query [37].

1.7.5 Thesaurus Expansion

As mentioned earlier, it is important to have a way of storing geographic information that can be used for expansion. One way it can be stored is in a geographic thesaurus. A geographic thesaurus, like all thesauri, is a controlled vocabulary arranged in a specified order with relationships between terms represented by known relationship markers. A geographic thesaurus would specifically store geographic information. So, thesauri are databases that provide information on words and phrases and the relationships between them. For instance, they often provide synonyms, meronyms, holonyms and other types of related terms. These words can be used to expand queries and hopefully improve retrieval. For GIR, a thesaurus being used would be most likely to be one that focused on geographic relationships between words. Thus for a location a thesaurus might provide, larger entities that the location was in, entities within the location, the locations next to the given location, the longitude and latitude of the current location and other information.

One difficulty of using a thesaurus is knowing how best to use the thesaurus. What sort of information within the thesaurus might actually help retrieval as opposed to hurting retrieval? For instance, most likely one wouldn't want to add antonyms of query words to the query because that might hurt retrieval. But not all of the relationships between words are as clear about whether they would be helpful or not. Thus people have experimented with what sort of terms they add as will be discussed in the context of related work in chapter 2.

Another difficulty of using thesauri is that in order to use one it must first be built. Where does all of the information come from? One common thesaurus in IR is WordNet, which provides much of the information that might be wanted for general expansion. For GIR, it is difficult because as mentioned in section 1.7.1 geographic information is spread all over the web and must be gathered and placed into a thesaurus. Thesauri can be created manually or automatically, automatic formation being preferred if possible because in that case the thesaurus does not require the services of a human to be built.

1.8 Re-weighting the Query

Along with query expansion another important technique for improving IR and GIR is re-weighting query terms, Re-weighting is when one gives more weight to certain words within the query for a certain reason. The hope is that re-weighting terms will improve retrieval. There are several ways that terms can be re-weighted. One simple method is to count the number of relevant documents a term or concept is in (rdf). This is useful because it gives more weight to the more common query terms, but it does not take into account if the term is mentioned once or many times. A second method would be log relevant term frequency times document frequency. This is useful because it takes into account both how frequently the term occurs and how many documents it is in. A third method would be taking the sum of the term frequency in relevant documents. This is useful because it would give more weight to the terms that were more commonly found in the relevant documents.

A fourth method is the Rocchio formula [33]. The Rocchio formula is one of the most popular methods for learning in IR. The idea is that by looking at the frequency of the terms in documents that are thought to be relevant the system can learn more about the "meaning" of the document and what terms are more likely to be relevant. Those terms that are more relevant will hopefully receive larger weights. The method also looks at the inverse document frequency. The method does not know which terms are relevant but it infers which are most likely to be relevant by giving the words a weight. It was originally designed for optimizing queries from relevance feedback, but it has also been adapted to other uses such as text categorization and routing problems [20]. The major component of the algorithm is the TFIDF (term frequency / inverse document frequency) word weighting scheme [20].

1.9 My approach

My hypothesis is that using geospatial information for query expansion and re-weighting the query terms based on geospatial components will improve retrieval effectiveness. This improvement will occur because the expanded query will be more clearly specified and will address the vocabulary mismatch problem. My approach to GIR will look at three different ways to disambiguate queries using geospatial information, particularly the geospatial portions of the query. I will look at re-weighting terms, query expansion and disambiguating terms such as “near.”

Why are geospatial terms so important? Geospatial terms contain information that can be used to more precisely specify a query’s meaning. For instance, the term “Bay Area” includes all of the locations that are part of the “Bay Area” in it, not only for example San Francisco. My approach will attempt to take advantage of the information that geospatial terms can provide to the query and to use this information to disambiguate terms.

Query expansion, as mentioned in section 1.7, is when terms are added to a query to increase its specificity and improve retrieval. For GIR it can be used to add geographic terms to a query. Expanding queries with geographic terms that are related to query terms will hopefully help disambiguate the query and improve retrieval.

One way to expand the query that I will use is blind feedback. As mentioned in section 1.7.3, blind feedback involves adding the top m words from the top n documents. The terms added by blind feedback help disambiguate the query. Also there is the chance that some of them may be geographic terms, in which case they could be used to help disambiguate geographic terms within the query. Still there is no guarantee that the terms added will be geographic, so it would be helpful to compare blind feedback with methods that specifically attempt to disambiguate geographic terms, such as adding words from a geographic thesaurus.

Re-weighting will be used in connection with expansion and on its own. This will help to give more weight to the expansion terms that are most relevant to the query. It could also conceivably help to see which terms are the most important to add to a query to disambiguate it. Rocchio is one of the most commonly used weighting schemes and the one I will use along with other weighting schemes, such as logrtfidf, rdf, rdfidf, rtf, rtfidf, and tfidf.

As discussed in section 1.6, central to my approach is the idea that geospatial information can help disambiguate geospatial terms. Since geospatial information in queries is often ambiguous it would be beneficial to disambiguate terms such as “near,” which have no clear definition. Longitude and latitude could be used to disambiguate words such as “near”. The longitudes and latitudes of two locations can be used in a formula to calculate the distance between the locations. Thus one could test whether a location falls within a certain distance of another location. This would allow one to test whether a location is defined as “near” in several ways and see which is the best definition to use. Also one could look at what “near” means for different sorts of objects. “Near” might mean one thing for a city and another thing for a country. In order to tell whether this is the case, one can change the definition of near and see whether queries where the location is a city do better with a different definition of near than queries with a country for a location.

1.10 Organization of the Thesis

This thesis is organized in the following way. Chapter two describes the previous work that has been done in both IR and GIR, focusing on work done at GeoCLEF 2005 and GeoCLEF 2006. Chapter three describes the strategy I used to test my hypothesis and how the programs I wrote fit into this strategy. Chapter four describes how I set up my experiments, what experiments I performed, and the results of the experiments. Chapter five presents a summary of the results of my experiments, my conclusions and suggestions for future work.

CHAPTER 2: BACKGROUND

GIR is an extension of traditional IR. As such it builds on techniques used in traditional IR as well as attempting to develop methods specific to GIR. The general techniques that can be used in GIR include blind feedback, query expansion, question-answering modules, passage retrieval, co-occurrence models, Named Entity Extraction, term expansion and Natural Language Processing (NLP). This chapter will examine work that has been done at GeoCLEF 2005 and GeoCLEF 2006 [12, 13], which focused specifically on GIR.

Many research papers on GIR were written for the Geographic Track of the Cross Language Evaluation Forum (CLEF). CLEF aims to develop infrastructure for the testing, tuning, and evaluation of IR systems on European languages both in monolingual retrieval and cross language retrieval. GeoCLEF was introduced as a new track of CLEF in 2005. The track was designed for research in GIR. It specifically deals with text GIR retrieval. GeoCLEF's focus is the retrieval of multilingual documents with an emphasis on geographic search. The goal is to provide a framework for evaluating GIR systems for search tasks involving spatial and multilingual aspects [13]. Since my research is monolingual, I will focus on the monolingual results from GeoCLEF.

For GeoCLEF 2005, the document collections were newspaper articles. The English collection consisted of articles from the *Glasgow Herald* (1995) and the *Los Angeles Times* (1994). The documents were not geographically tagged and did not contain any location specific information. The topics or queries were broken up into title, narrative, and description fields and had location, spatial relationship and concept tags that had been manually created. Eleven groups participated in GeoCLEF 2005.

In 2006, GeoCLEF became a regular track in CLEF. The purpose remained the same; to test and evaluate cross-language GIR. Participants were offered TREC style ad hoc retrieval based on existing CLEF collections. Based on the results of GeoCLEF 2005 it was seen that more work needed to be done on identifying the research and evaluation issues around GIR. In order to not favor systems relying on keywords, GeoCLEF 2006 concentrated on more difficult geographic entities like historical or political names that are used to refer to geographic regions. Some topics required the use of external geographic information. An additional difference from GeoCLEF 2005 was that the creators of the topics tried to include a wider variety of geographical relations and different location types [12]. GeoCLEF 2006 used a tentative classification of topics into eight categories:

1. Geographic subject with non-geographic restriction
2. Geographic subject restricted to a place
3. Non-geographic subject restricted to a place
4. Non-geographic subject that is a complex function of place
5. Geographical relations among places
6. Geographical relations among events

7. Relations between events which require their precise localization [12]

There were more participants in 2006, with a total of 17 participants, 9 of whom had not participated in 2005.

The techniques used by the various groups at GeoCLEF 2005 varied from basic IR approaches (such as query expansion) with little use of spatial and geographic reasoning to deep Natural Language Processing (NLP) [12]. Groups at GeoCLEF 2006 used many different techniques as well including: Ad-hoc techniques (e.g., blind feedback, manual query expansion), Gazetteer construction, Gazetteer-based query expansion, question-answering modules using passage retrieval, Geographic Named Entity Extraction, term expansion using terms from WordNet, automatic and manually constructed geographic thesauri, resolution of geographic ambiguity, Natural Language Processing (NLP), and part of speech tagging [12].

The various groups at GeoCLEF 2005 and GeoCLEF 2006 represented geospatial information in various manners such as gazetteers [6, 7, 9, 17, 22, 29, 30], semantic networks [27, 28], Geographic Knowledge Bases [5], and Geographical Thesauri [7, 8, 9, 19]. Below I examine some of the common approaches before examining methods used at GeoCLEF.

The rest of this chapter explores specific techniques and approaches towards GIR. In section 2.1, I explore Natural Language Processing including parsing and Named Entity Extraction. Section 2.3 discusses the use of different forms of query expansion. Sections 2.3-2.12 describe specific methods used at

GeoCLEF 2005 and GeoCLEF 2006. Section 2.13 summarizes the previous work done. The chapter concludes with section 2.14, which discusses my approach and its connection to the previous work.

2.1 Natural Language Processing (NLP)

Natural Language Processing (NLP) studies automatic generation of and understanding of human language. NLP, like IR, can be done with spoken language or written language. NLP approaches try to extract the meaning of words or documents from what is known about the documents. NLP is used frequently for Artificial Intelligence and for linguistics. Two NLP techniques that were used at GeoCLEF were parsing (section 2.1.1) and Named Entity Recognition (section 2.1.2).

2.1.1 Parsing

One example of a NLP technique is the formation of parse trees to understand the syntax of a sentence in order to disambiguate it and thus get an idea of the meaning of the sentence. One thing NLP sometimes tries to do is to convert human language into a more formal version that a computer can understand. This computer understandable version would attempt to convey the meaning of the original words to the computer. This is what a parse tree is attempting to do. As mentioned in section 1.6, ambiguity of language is a problem for IR. One method of disambiguation would take terms with multiple meanings and try to identify the intended version by parsing the sentence.

Parsing can be useful but one of the cons of using it is that it is computationally expensive. It is also difficult to do correctly and there are no systems that can do it completely. It can be done for specific domains, but there is no one parsing system that works for a general domain. It is difficult because while there are specific grammatical syntaxes for smaller domains (for example academic writing or legal writing) there is no grammatical syntax that is common in all domains. In addition, in many cases for IR one doesn't necessarily want the exact meaning or the grammatical syntax of a sentence. As long as the computer can find an idea of the meaning for retrieval, it doesn't necessarily need as detailed an idea of the meaning as a parser tries to find.

One group at GeoCLEF manually split the geographic part of the query into a parse tree of conjunctions and disjunctions [32]. Each document that was retrieved by the text retrieval engine was checked for terms within the parse tree; if it did not have any it was eliminated. The group noted that their manually created parse trees resembled the trees an automatic query parser might create and they planned to implement an automatic query parser in the future (See Section 2.7 for more discussion of this group's work) [32]. The University of Hagen experimented with doing GIR with Deep Sentence Parses [27]. They used a modified version of the InSicht Question Answer system for GIR to do Deep Sentence Parses (See section 2.10 for discussion of semantic networks). They also used a semantic parser to analyze the query text prior to retrieval [28] (See Section 2.2.3). In addition, groups tried geo-parsing, which aimed to find geospatial terms and relationships between them within queries [18] (see Section 2.4).

2.1.2 Named Entity Recognition (Named Entity Extraction)

Most of the groups at GeoCLEF used a form of Named Entity Recognition, which is a NLP technique, to locate geographic terms and places in the documents and queries. Named Entity Recognition aims to find pre-specified information in the text and differentiate this information from words found in the dictionary. Named Entity Recognition attempts to locate and classify individual elements in the text into predefined categories, such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc. It does this by matching patterns for types of information to terms in the texts or queries. In GIR, Named Entity Recognition would be used to classify and tag terms that are geographic entities. Named Entity Recognition systems have been created that use linguistic grammar based techniques as well as techniques based on statistical models. Grammar based systems typically are better, but have a higher cost of work as they require experienced linguists to define the grammar when the system is being designed. In addition, they are hard to transfer from language to language. Grammar based systems only work in limited domains. Statistical systems, on the other hand, typically require large amounts of manually tagged training data, but can be used for other languages, domains or genres of text more rapidly and often require less work overall. Unfortunately the training data is often difficult to get.

For GeoCLEF, almost all of the groups used some form of Named Entity Extraction to find and extract geospatial terms [5, 6, 7, 9, 10, 18, 19, 22, 23, 26, 27]. Often the usefulness of this depended on the amount of geospatial data available. One complaint for GeoCLEF 2005 was that often when attempting Named Entity Extraction it was difficult to find geospatial information related to queries that referred to very specific locations. For instance, one group pointed out that their Named Entity Recognizer did not recognize “Scottish Trossachs” because they had no data on this location [27]. This implies that crucial to the use of a Named Entity Recognizer is geographical data, because the Named Entity Recognizer can only recognize those entities that it has data about.

2.2 Query Expansion

Query expansion, as discussed previously in section 1.7, modifies a query into a new form by adding words or phrase to the query. This modified query will hopefully retrieve more relevant results than the original query. Query expansion can include adding synonyms of words, finding all of the morphological forms of stemmed words, fixing spelling errors and re-weighting terms in the original query. Below is a discussion of the different query expansion methods attempted at GeoCLEF. In 2.2.1 manual and automatic query expansion is discussed, followed by a discussion of specific techniques for query expansion; 2.2.2 examines blind feedback; 2.2.3 examines thesaurus expansion; 2.2.4 discusses gazetteer expansion; and 2.2.5 discusses query expansion with re-weighting.

2.2.1 Manual versus Automatic Query Expansion

Manual query expansion and automatic query expansion, as discussed in section 1.7.2, are two forms for query expansion. Manual expansion relies on modification by a human expert, while a computer can do automatic expansion.

Several groups at GeoCLEF compared results of these two techniques, including the State University of New York at Buffalo group. They used pure IR techniques, including, single word terms (as discussed in section 1.4), word bigrams (strings of 2 letters as discussed in section 1.4) and blind feedback, to improve GIR [34]. They did both manual and automatic runs. For the manual runs they created a Boolean query by manually adding terms from geographical resources on the web. The average performance of the manual and automatic runs are similar, but a query by query analysis shows that on 8 of the 25 queries there were significant improvements for the manual run. Also, for 5 queries the manual runs perform significantly worse than the automatic runs [34].

Berkeley2 also used manual expansion. They found that manual expansion of geographic references was detrimental to retrieval performance [14]. Berkeley used two different systems based on the Logistic Regression algorithm. This is a model of probabilistic IR. It attempts to estimate the probability of relevance for each document based on a set of statistics about a document collection and a set of queries as well as a set of weighting coefficients for the statistics. Berkeley used two different implementations of the TREC2 Logistic Regression algorithm, one was in experimental software developed by Aitao Chen and the other was in the Cheshire II IR system. The basic behavior of the algorithm is the same in both systems, but there are differences in pre-processing and indexing elements. The software developed by Aitao Chen treats the text as a single “bag of words” to be extracted and indexed. It uses a decompounding algorithm to extract component terms from German compounds. The Cheshire II system uses an XML structure and extracts selected portions of the structure for indexing and retrieval and does not do decompounding of German terms. Berkeley also implemented a form of blind relevance feedback as a supplement to the TREC2 logistic regression algorithm. From their experiments, Berkeley drew the conclusions that manual expansion of selected topics shows a small improvement over automatic methods. Also they noted that for 2006, they did not do any specific geographic processing but are planning to add that in the future [25]. California State University San Marcos also manually processed the topics with gazetteers [16].

As seen from these experiments manual expansion can in some cases do better than automatic expansion. Unfortunately, it does not do better for all cases and manual expansion has the disadvantage of requiring humans to expand the queries. In a real life situation, most likely the user would not be willing to wait while a different person expanded her query. So, while manual expansion can sometimes do better it seems that automatic expansion is more feasible for the average system to use. In addition, in general IR, it has been shown that automatic expansion systems often do as well as manual expansion systems.

2.2.2 Blind Feedback

Blind feedback, as discussed in section 1.7.3 is where retrieval is performed and the top n documents returned are assumed to be relevant. High-ranking terms from these documents are then added to the query in hopes that since they are presumably in a relevant document they will help to locate more relevant documents.

At GeoCLEF some groups experimented with blind feedback and using that to expand the queries with geographic terms. For Berkeley2 at GeoCLEF 2005, blind feedback only improved certain kinds of queries, particularly queries for German monolingual and bilingual runs. The most improved queries were ones to which many proper names and word variants were added, but few irrelevant words were added [14]. This makes sense because those queries were made more specific and did not have a lot of junk added to them. This might suggest that having a Named Entity Recognizer look over the new query words to see if they were proper names or having method check to see if the terms were word variants of query terms might improve retrieval.

The University of Hildesheim used blind feedback and named entity based query expansion [1]. They combined the weighting and expansion of geographic named entities with a Boolean retrieval approach, which means that the terms in the query are linked by Boolean operators, such as AND, OR, NOT. In Boolean retrieval, in order to be retrieved a document has to fulfill the Boolean expression that the query forms. They explored the effect of adding particular geographic named entities within blind feedback. The results indicate that geographic named entities can improve blind feedback and that geographical expansion with Boolean retrieval does not necessarily lead to better results [1]. If a different retrieval model was used adding geographical terms might lead to better results.

2.2.3 Thesauri

Another method of query expansion is the use of thesauri, databases that provide information about the relationships between words and phrases, as discussed in section 1.7.3. Several groups at GeoCLEF expanded using related words such as synonyms (words that have similar meanings, e.g. “interesting” and “fascinating” are synonyms), meronyms (words that name a part of a larger whole, e.g. “finger” is a meronym of “hand”) or holonyms (words that name the whole of a smaller part, e.g. “hand” is a holonym of “finger”). An example meronym expansion in the context of GIR is expanding “Europe” to include all the names of the individual countries of Europe. This expansion was found to be disastrous at GeoCLEF 2005 because it caused the query to contain too many individual countries that on their own were not found in many of the relevant documents [2, 14, 26]. Other expansions, like using holonyms, were more successful [2].

Several groups at GeoCLEF used term relationships to choose words to expand with. The Universidad Politecnica de Valencia, which examined the use of WordNet (a semantic lexicon for the English language that can be downloaded from the internet) based expansion for GIR [2, 3], did query expansion using synonyms and meronyms. They found that expanding queries hurt retrieval [2]. They later continued to examine the use of WordNet synonyms and holonyms to expand index terms in the hope that it might lead to finding implicit geographic information from the text, particularly in the case where the containing geographic entity is omitted (e.g. France is not mentioned but Paris is). Their system was based on the Lucene search engine. Their results show that expansion could improve recall for some cases, but that the ranking function they used needed to be better in order to obtain better precision [3]. One other group also found that query expansion with WordNet meronyms was not effective [26].

The University of Hagen experimented with metonymy recognition in documents [28]. Metonymy recognition is identifying the substitution of one word for another word that it is associated with. They used a separate index for location names and for identifying and indexing metonymic locations names separately. For these experiments, the University of Hagen used a modified version of the GIR system they used for GeoCLEF 2005. They used a classifier to identify metonymic location names in order to preprocess the documents. The classifier was based on shallow features (e.g. position of words in a sentence, word length, base forms of verbs) and trained manually. After preprocessing, documents contained additional information for locations that were indexed separately. They used two methods to generate IR queries. The first used a semantic parser to analyze the query text and then translated the resulting semantic net into a query. The second method used a Boolean combination of a bag-of-words with location names. The results indicated that excluding metonymic senses of location names improves mean average precision in most experiments. Using the narrative field of the topics decreased mean average precision. Query expansion and the use of separate indexes improved the performance of the GIR application [28].

Berkeley1 also used word relationships for expansion. They used the Cheshire II system to test the fusion of multiple probabilistic searches against different XML components using both Logistic Regression algorithms and a version of the Okapi BM-25 algorithm [24]. Berkely1 geo-referenced proper nouns in the text using a gazetteer and expanded the place names for regions and countries in the queries with names of countries or cities in that region or country. In their approaches, they indexed and extracted terms and searched GeoCLEF collections. For results, the queries using the location tags and expansion did better than those that did not use these [24].

In addition, several groups specifically used geographic thesauri to expand terms. One group that specifically used a geographic thesaurus was the SINAI group, who experimented by expanding topics with geographical information [9]. Their system had three parts:

1) A translation subsystem

2) A query expansion subsystem that uses a Named Entity Recognizer, a gazetteer, a thesaurus expansion module and a geographic information module

3) An IR module

SINAI did several different runs combining these modules. The results showed that geographical and thesaurus information for query expansion did not improve retrieval, but that more research needs to be done [9].

Expanding with related terms such as synonyms, meronyms and holonyms can be useful but it can also hurt retrieval. It appears that in expanding specific terms, it is important to be careful how far to expand the query. For instance, adding all of the related terms may expand the query too much and hurt retrieval, while not expanding will have no effect on retrieval. So, a balance needs to be reached between expanding the query with too many terms and not expanding at all. When expanding with a geographic thesaurus, one has to be careful about how far the queries are expanded. The types of geographical relationships and the breadth of those relationships are important. For instance, in expanding a location, what sorts of relationships are good to add to the query and which will hurt it? In addition, the system is dependent on the information in the thesaurus. So, if the thesaurus is missing information expansion may have less affect than otherwise.

2.2.4 Gazetteer

A gazetteer is another repository for geographic information. A gazetteer is a dictionary for place names, which provides information about the locations in it. Expanding with geographic terms can include expansion with synonyms, meronyms and holonyms. Thus it shares the problem of how far something should be expanded with geographic thesaurus expansion. In addition, again for this, one is limited by how much geographic information one has in the gazetteer. If one doesn't have any information on a location, then that location cannot be expanded. So, for this it is important to have a resource that provides geographic information that might be used to expand terms.

Several groups at GeoCLEF used a gazetteer to do expansion instead of a geographic thesaurus. The Microsoft Research Asia (MSRA) Columbus Project used a gazetteer and rule based approach to extract locations from the corpus of documents [30]. They used both text indexing, which indexed single word tokens, and geo-indexing, which attempted to find and index locations. They compared five runs:

1) MSRAWhitelist, based on the title field of the query and using the geographic knowledge base to expand locations and then manually expanding locations that the geographic knowledge base could not find

2) MSRAManual, based on the title and description fields of the query and then some textual terms were manually modified

3) MSRAExpansion, based on the title and description fields of the query, where the original queries were used to search the corpus and then locations were extracted from the documents and the 10 most frequent locations were added to the query

4) MSRALocal, based on the title field of the query, where locations were simply extracted from the queries

5) MSRAText, based on the tile, description and narrative fields of the query, where the text search engine was used to process the queries

The results show that MSRAManual did the best followed by MSRAWhitelist. MSRALocal and MSRAText performed similarly and MSRAExpansion performed the worst due to the addition of unrelated locations to many of the queries [30].

The NICTA group experimented with geographic-based query expansion by using a gazetteer to extend geospatial terms to nearby locations. This process used a named entity recognition system, a toponym resolution component to assign probabilistic likelihoods to geographic candidates from the gazetteer and probabilistic GIR approach. They expanded location names in both documents and queries and used a normalization process to adjust term weights. Their GIR runs showed little improvement over the baseline runs [29].

The University of Alicante examined how geographic knowledge could be incorporated into GIR. They used IR-n (an IR module) as well as a Geographic Knowledge module (Geonames) [36]. Geonames is a geographical database which integrates geographical data (e.g. names of places in various languages, elevation, population and others) from various sources. Geonames was used to expand initial topics and queries by adding geographic items. Geographic items and relations were extracted from the topics and queries using the Geonames database. Geographical information related to the items and relations was returned by Geonames. This information was then incorporated into the topics, which were processed by IR-n. The results show that adding geographic knowledge has a negative impact on precision, but some topics obtained better results. This implies that geographic knowledge could be useful but research needs to be done to determine how to apply it correctly [36].

2.2.5 Weighting Schemes

Re-weighting query terms can be a form of query expansion. In addition, other methods of query expansion, such as blind feedback often re-weight query terms in the process of expanding the query. There are many different weighting schemes, many of which use TF and IDF to calculate weights.

In my research, I have primarily used the Rocchio formula which was discussed in section 1.8. The Rocchio formula was developed as a method for optimizing queries using relevance feedback. Originally the steps involved in relevance feedback, according to Rocchio, would be to first perform retrieval, and then have a user specify whether the top n documents were relevant. Those documents said to be relevant would be used to modify the query and find new words to add to it. The additional words and the original query terms would be weighted using the Rocchio formula [33]. The version I use is similar but it assumes that the top n documents returned in the first retrieval run are relevant. Thus it performs automatically without any need for a user to manually choose which documents are relevant. This means that the queries can be expanded more quickly than if a user was needed, but it also means that the words added to the query may not be relevant, since how many of the top n documents are relevant depends on the choice of n.

2.3 Use of Thematic and Geographic Aspects of documents for GIR

For GIR the queries have both thematic and geographic aspects. The thematic aspects deal with the overall theme of the query and may or may not have a geographic part. The geographic aspects, on the other hand, are made up of the parts of the query that are geographic in nature. For example, geographic terms within the queries would be part of the geographic aspect.

The Computational Linguistics Group examined the thesis that both thematic and geographic aspects of documents may be useful for GIR [11]. This meant that they wanted to test whether geographic parts of documents and the entire document could be useful. They located geographic parts of the documents automatically using a GeoTagger, called Alias-I LingPipe, to detect place names, geographic concepts, spatial relations and adjectives referring to things, people or language connected to a place. This was a form of Named Entity Extraction. They created two indexes, one for geographically relevant terms and one for reference document collections. They did runs using just the index of the entire document collection and using the index of the document collection and the index of extracted geographic terms from the topics. The extracted geographic terms were expanded using major cities, towns and places from their geographic knowledge base. In using these two indexes for GIR, they did not observe any significant improvement through the use of geographic query expansion, but noted that more research needed to be done [11].

Looking at the thematic and geographic aspects of GIR might prove useful, with more research, in that it attempts to separate the GIR part of the process from the IR part of the process. It attempts to see how useful the geographic terms actually are for GIR. At the same time, a possible disadvantage of this idea is that there is automatically an overlap between the two parts. Some of the geographic parts are used in the thematic aspect and thus their influence would be looked at twice. There is no way to completely separate thematic and geographic aspects because the geographic aspects can also be thematic.

2.4 Splitting the Process into Textual Retrieval and GIR

A method that several groups explored was splitting the retrieval process into two parts: textual retrieval and GIR. Textual retrieval is like traditional IR and does not specifically take geographic information into account. GIR, on the other hand, takes the geographic information into account. The experiments of several groups that used a combination of textual retrieval and GIR are described.

The MIRACLE group focused on creating multilingual gazetteers, recognizing geo-entities, processing spatial queries, document tagging and document and topic expansion. The MIRACLE group used a Boolean model for geo-entities recognition and a probabilistic model for textual retrieval. Their model for topic expansion was based on determining the existing geographical resources (e.g. continent, country, region, country, city) in a space region defined by at least three points (North, South, West, East) [22]. Their baseline approach to processing documents and topic queries included standard steps like stemming words, removing stop words, and converting words to lowercase. The Miracle group observed that topic expansion improved precision results slightly for some topics, but hurt other results. They also found that the fundamentals of a GIR system were a Named Entity Recognition system in conjunction with GIR, as the geo-entity recognition process that they used could not distinguish named entities correctly. This caused the runs that used only the location tag for topic expansion to do better than those that used all of the text [22].

Later the MIRACLE team attempted to test the effects of GIR from documents with geographical tags [23]. They tried to isolate geographic retrieval from textual retrieval by replacing geo-entity textual references in topics with associated tags. The tags specified a geographical path (e.g. continent names, country names, region name…) to a unique place in the gazetteer. They also split the retrieval process into two steps:

1) Textual retrieval without geo-entity references

2) Geographical retrieval using tagged text generated by a topic tagger, where a named entity tagger was employed to tag geographical entities after they had been identified by a named geo-entity identifier

The textual and geographical results for each query were combined by taking either the union (OR), the intersection (AND), difference (AND NOT) or the external join (LEFT JOIN). Each of these techniques re-ranked the output and computed new relevance measures values from the input values. The Miracle GIR system consisted of linguistic tools oriented to textual analysis and retrieval and resources and tools oriented towards geographical analysis. These different tools were combined to carry out different phases of the system. Compared to their results of GeoCLEF 2005 it can be seen that mean average precision gets worse when textual geo-entity references are replaced by geographical tags. This is partially due to the Miracles system returning zero pertinent documents if no documents fulfilled the geographical subquery. If those results which returned zero documents are analyzed, the remaining results show an improvement in recall precision values. The conclusion drawn by the Miracle group was that it is necessary to improve their named entity module, because that is necessary to recognize geo-references within the text [23].

Several other groups also separated text retrieval and geographic retrieval. One was the group from the University of New South Wales (UNSW) [18], who used a system that consisted of four different modules: a geographic knowledge base, an indexing module, a document retrieval module and a ranking module. The geographic knowledge base stored, organized and represented geographic data and knowledge. They used an object-orient modeling method for their data scheme. The indexing model created and maintained a textual index and a geographic index separately. The geographic index was built in three steps: 1) matching strings in documents to a place name list derived from the geographical knowledge base, 2) a Named Entity Recognition process that tagged person, location and organization entities, 3) matching the results from the previous two steps. To match locations from the previous two steps, first locations from the first step that were not tagged as locations in the second step were eliminated and then place names in the stop word list that were tagged as locations in the second step were added to the geographic index. The document retrieval module retrieved documents using a four phase procedure involving query parsing, textual searching, geographic searching and Boolean intersection. The ranking module was a genetic programming based algorithm to discover ranking functions and to rank the documents. UNSW used a Boolean model that required documents to meet both textual and geographic criteria before they could be retrieved. Their results showed that the geographic knowledge base, the indexing module and the retrieval module are useful for GIR but that the ranking function they used needed to be improved [18]. One other group [32] also separated textual and geographic parts of retrieval and used the geographic retrieval to eliminate documents retrieved by the text retrieval system (See section 2.7 Co-occurrence).

In looking at the separation of textual retrieval and GIR, several pros and cons appear. On the pro side, separating the two makes sure that the system takes non-geographic material into account as well as the geographic material. On the other hand, textual retrieval and GIR use some of the same information, which results in the geographic parts considered for retrieval in both parts of the retrieval process, which may be either beneficial or harmful.

2.5 Geo-Scopes

Geo-scopes are a way of assigning some sort of rank to documents. The rank is based on geographic information within the documents and within geographic resources. Hopefully, the higher the ranking of the document the more relevant that document is. The idea behind assigning geo-scopes is to combine information and disambiguate among different possible scope assignments that each document could be assigned [5].

The XLDB group experimented with geo-scopes [5]. They used CaGE, which recognized geographical references and assigned a geo-scope to the documents. The researchers first found geographic references and gave them a weight based on their frequency. A geo-scope was then assigned to the entire document by looking at the geographical references in the documents, their frequencies and the relationships between them. This rank was used to calculate the geo-scope of the document and to help decide which documents were relevant. They found that using location terms had better precision than methods using geo-scopes. They also found that graph-based assignment scopes, geo-scopes that were based on the distance between different nodes on a graph, had better precision than the other geo-scope methods [5].

The XLDB group also tested text mining methods that used an ontology to extract geographic references from text and assign documents geographic scopes [31]. The scopes were used in document retrieval by the ranking function; documents that had a similar scope to the query were given higher rankings. In addition, the XLDB group tested a topic augmentation method that was based on using a geographic ontology. The steps of the augmentation method were: 1) locate concepts in the ontology, 2) if a relation term from the topic title is "near" use the ontology to get the top k nearest locations and top k adjacent regions, 3) rank the list of concepts from the previous steps and 5) select place names from the top 10 ranked concepts to add to the original topic. The results showed that a relatively simple augmentation scheme for geographic terminology can outperform the text mining approach [31].

Geo-scopes allow geographic information within the documents to be taken into account in the ranking function. This is potentially a useful tool. On the other hand, they do not take into account non-geographic aspects of the documents, for instance the non-geographic thematic aspects. Thus it seems that the geo-scopes might miss some documents that did not have as many geographic terms but had the thematic aspect, so they need to be used in combination with regular retrieval. In addition calculating geo-scopes would be dependent on being able to accurately recognize geographic terms within the documents.

2.6 Geographic Tags

Geographic tags are tags on either the documents or the queries that highlight the geographic aspects of the document or query. These can be used to find at least some of the geographic information within either the documents or the queries. The tags can be added manually by a person who reads through and tags geographic aspects of the document or query or they can be added automatically by a Named Entity Recognition module that tags geographic entities. Next I describe some uses of geographic tags at GeoCLEF.

The University of Alicante system was developed to retrieve documents containing geographic tags [6]. The system has three parts:

1) An IR module (IR-n)

2) A named entity recognition module based on machine learning (NERUA)

3) A rule based named entity recognition module (DRAMNERI)

The researchers applied different combinations of these three modules to see which modules were the most useful in the context of GIR. IR-n is a passage retrieval system. This means that it studies the appearance of query terms in fragments of documents, referred to as passages, rather than in entire documents. NERUA uses three different machine learning techniques (K-nearest neighbors, maximum entropy and hidden Markov models) to identify named entities. The system has two parts: one for entity detection and the other for entity classification. NERUA uses lexical features, contextual features, gazetteers, trigger word lists, and morphological features. Its performance is mainly due to a weighted voting strategy. NERUA uses three different classifiers and each classifier gets a vote in deciding what category an element goes in. DRAMNERI is a rule based system that identifies and classifies named entities. One of its aims is to be as customizable as possible. In looking at the monolingual results obtained by the University of Alicante, it can be seen that NERUA improves the results for English but not for German. This is because the system was prepared for the English language and lacked resources for German. The results also show that DRAMNERI did not obtain good results, likely because it needs more resources [6].

Berkeley2 also used geographical tags for GeoCLEF 2005 in combination with blind feedback [14]. They experimented with using the geographical tags provided in the queries and by manually expanding the location tag to add more specific location tags. Manual expansion of the location tags was a losing strategy because it expanded the locations too much (also see section 2.2.2 Blind Feedback) [14].

Geographic tags can help to locate geographic information within a document or query, but at the same time these tags have to be created either automatically or manually, which could conceivably be a problem. Automatically creating the tags would require using a Named Entity Recognition system to find terms to tag. Manually creating tags, on the other hand, would require a person to look at the documents and tag things correctly, which takes a long time.

2.7 Co-Occurrence models

Co-occurrence takes into account how often certain terms occur together within text windows. It is assumed that words near each other are more likely to be connected to each other. For co-occurrence one has word pairs that can occur together. These two words are looked for within the documents and can only be a certain distance apart from each other. This information on how often the word pairs co-occur can be used to improve retrieval because then the system can look for phrases as well as individual words. For instance, if the query includes the phrase “white house” one would want documents that include “white house” and not “the white walls in their new house…” Co-occurrence could help identify if words that are connected by looking at how often they occur together.

One group experimented with place disambiguation using co-occurrence models [32]. The group used co-occurrence models for name disambiguation as well as using named entity recognition tools and text indexing applications. Their system was split into two stages, a batch text and geographic indexer and a real time query engine. The geographic indexer took named entities tagged by a Named Entity Recognizer and disambiguated them based on how they co-occurred in their co-occurrence model. The query engine took manually crafted queries, which separated the text component from the geographic component. The text query was handled by the Lucene search engine and the geographic query was manually split into a tree of conjunctions and disjunctions. Each document that Lucene retrieves is checked for locations that match the geographic query, if it has no locations that match the geographic query it is removed. The results of the experiments showed that while there was significant need for improvement in the system, co-occurrence models are a suitable method for place name disambiguation [32].

Like any other method there are pros and cons to co-occurrence models. One pro is that it allows the fact that some words occur frequently together to be taken into account. One difficult part is deciding exactly how far apart the words should be from each other and also that some of the pairs may be found in situations that are not relevant.

2.8 Question Answer (QA) Based Systems with Geographic indexing

Question Answer (QA) is a task in which the queries are seen as questions to which the answers rather than documents are returned. Typically an IR search engine is a component of a QA system. The search engine suggests which documents are likely to contain answers; then those documents are analyzed to extract potential answers.

TALP experimented with using a Question Answer (QA) based IR system, linguistic analysis and a geographical thesaurus [7]. Their system had two phases: topic analysis and document retrieval. Topic analysis extracted and analyzed relevant keywords using a keyword selection algorithm based on linguistic and geographical analysis of the topics. They also used a geographical thesaurus from a geographical gazetteer. During topic analysis the named entities that are classified as locations or organizations were geographically analyzed. This step had two components: one was a geographical thesaurus and the other was a NEC correction filter, which corrected some common errors. Document retrieval was based on the Lucene system. It used a modified version of the passage retrieval module used by the TALP Question Answering System. It also took into account geographical terms providing that geographical terms could be looked for strictly (which needs a region, country and coordinates), or relaxed (which needs only the region and country; this returns all cities and regions inside the country). The results of their experiments showed that geographical indexing and retrieval could improve the performance of a GIR system [7]. For GeoCLEF 2006, TALP modified the GeoTALP-IR system and used JIRSPassage Retrieval software and Lucene with the Alexandria Digital Library Feature type thesaurus [8]. They modified GeoTALP-IR system by adding two extra steps to the process: an extra retrieval step and a ranking step. They used the JIRS system to do textual document retrieval and Lucene to detect geographically relevant documents. The documents were ranked with the top-scored documents retrieved by JIRS that were also retrieved by Lucene given the highest rank followed by the other top documents retrieved by JIRS until 1,000 documents had been selected. Their experiments showed that using JIRS obtained better results than combining both JIRS and Lucene. Their results could be explained by several reasons: 1) the JIRS system might not have been appropriate for GIR, 2) the system did not deal with geographical ambiguities, 3) the system lacked query expansion methods, 4) the system needs relevance feedback methods, and 5) there were errors in the Topic Analysis phase [8]. The University of Hagen also used an adapted Question Answer (QA) system and semantic networks in their experiments (See sections 2.10 on semantic networks) [27].

2.9 Geo-Filtering

Geo-filtering involves filtering documents based on geo-spatial information. This classifies documents as either relevant or not relevant. One group [26] did spatial retrieval based on named entity tagging, toponym resolution (maps an entity to a spatial representation, given the context) and re-ranking by geographic filtering. They did both plain word retrieval and phrasal retrieval and used three different geo-spatial filtering techniques which eliminated documents or portions of a document that did not fall within the geographic area of interest described by the query. The geo-filtering predicates were called “Any-inside”, “Most-inside” and “All-inside.” “Any-inside” was the most conservative and attempted to avoid discarding true positives but risked underutilizing the discriminative power of geographic information for IR. It filtered out documents that did not mention any locations in the query. “Most-inside” was more aggressive than the first filtering technique and discarded documents that mentioned more locations outside of the query than locations inside the query. “All-inside” was the most aggressive filtering technique and discarded all documents that mentioned even one location outside of the query. They found that the most conservative geo-filtering (“Any-inside”) outperformed the other types of geo-filtering [26]

The University of Twente experimented with geo-filtered document retrieval [17]. Their approach was to retrieve documents based on content and then filter them by geographical relevance using a gazetteer. The gazetteer was used to tell whether the locations within a document were within a certain geographical scope. If a document had at least one location within the geographical scope it was deemed relevant. Unfortunately, their results did not show an improvement in retrieval performance when geographical information was taken into account. They suspected that there was a bug in their geographic filtering system, that the Wikipedia pages used to help with the filtering had too much extra information that was not relevant to the topic or that the gazetteer was too large and should be modified so that importance scores were given to locations with a single name [17].

Two difficulties with geo-filtering, as seen by the experiments at GeoCLEF, are how conservative the filtering should be and how much non-relevant information the sources used for filtering contain. These sources would need to be edited so as not to have as much non-relevant information. In addition, more experiments would be useful to see how conservative or open the filtering should be.

2.10 Semantic Networks

Semantic networks are a way to represent knowledge. They are often used for Artificial Intelligence. A semantic network is a directed graph consisting of vertices and edges. The vertices or nodes represent concepts while the edges represent the relationships between the vertices. The structure of the network gives it meaning. Some relationships that can be included in semantic networks include: “part of”, “near”, “is a” or any other relationship that one could define. In order for a machine to understand the network, it has to know what the different relationships are and how to recognize them. This is similar to a thesaurus, but semantic networks offer different methods of navigation through the data and they go farther in defining the relationships between concepts allowing them to be more easily manipulated by systems than thesauri are.

The University of Hagen group [27] represented their background knowledge, queries and documents as semantic networks and expanded geographic concepts with semantically related concepts. The semantic networks represented the information as labeled nodes connected by edges or arcs that represent different relationships between the nodes. The semantic network represented the relationships between different objects in the network, focusing on the topological, directional, and proximity relationships between concepts. Some of the relationships in their semantic networks included: ATTACH (attachment between two objects), ATTR (one object is an attribute of another object), CIRC (situational circumstance), LOC (x took place or is located at y), and *IN (x is in y). The University of Hagen group tried three different experiments. The first one was a basic traditional IR approach. The second one, which gave the best results, used the natural language description of a topic transformed into a semantic network and transformed the semantic network of the topic into an intermediate representation of a database query and then expanded the thematic, temporal and geographic terms in the query. The third method transformed queries and document sentences into semantic networks and used an adapted Question Answer (QA) algorithm to match the semantic representations of queries to the semantic networks for document sentences to measure their similarity.[27].

Semantic networks are useful for representing relationships but they also require that one define the relationships beforehand. If there are two concepts that are related in a way different from the relationships defined in the current network they would not be connected on the network unless one added a new relationship. So, when using a semantic network the relationships have to be carefully defined before the semantic network can be used and the computer must have an algorithm for going through the network. This implies that semantic networks will do better in environments that have clearly defined relationships between concepts. It also implies they would be more useful in specific environments where there is one set of relationships than in a general environment where there are more relationships that need to be considered.

2.11 Combinations of methods

In addition, to using one of the methods described above, different methods previously mentioned can be combined. This could be done even to the extent that several different systems could be used and then the results combined to hopefully take advantage of the strengths of the different systems. On the other hand, combining systems is difficult because it is hard to find a way of combining the systems in a way that takes advantage of the strengths of all the systems, but does not underestimate or overestimate the importance of one approach. One example of combining systems that shows this difficulty was seen when the SINAI group, the University of Alicante group and the group doing WordNet based expansion worked together to see if a combination of their systems would result in better retrieval [10]. In order to do this they designed a simple voting scheme for the systems, where the scores for each system where normalized and then added together to get a final score for the document. The average precision obtained by this mixed system was poor, implying that more studying of the strengths of each system should be done and that a better voting system should be developed [10].

2.12 Other Approaches

Since research in GIR is just beginning, it is not surprising that some people have tried not using geographic reasoning for GIR or if they used it found little or no improvement due to it. Several groups at GeoCLEF 2005 found themselves doing keyword searches. Hopefully as more research is done and the differences between IR and GIR become clearer, GIR will not seem as close to a keyword search.

In addition to groups that used spatial and geographic reasoning, several groups used basic IR techniques with none or few modifications to take geographic reasoning into account. These groups got some of the best results for GeoCLEF 2005 and found that GIR was very similar to a basic keyword search [15, 19, 21]. Part of the reason for this is that most of the queries used for GeoCLEF 2005 used the spatial relationship “in” which requires looking for a location. This is basically a keyword search. When topics contain more complex spatial relationships than “in”, the task is more challenging and is not as close to a keyword search.

The groups that did not use much geographic information included California State University San Marcos [15]. They manually processed the topics with gazetteers [16]. A second group that did not use much geographic information was the NICTA i2d2 group [19]. They did use geographic information from the Getty Thesaurus of Geographic Names as they did Named Entity Recognition, but they noticed no improvement in results of the geospatial topics versus the baseline topics [19]. Similar to NICTA, MetaCarta, participating in GeoCLEF 2005, found that they were doing a keyword search. They used bounding boxes for each region in the topics but found that the geographic and non-geographic results were very close [21].

2.13 Summary of Previous Work

As discussed in this chapter there have been a wide variety of techniques tried for GIR, some successful, others not. Many results showed little improvement over the baseline or in some case a negative impact showing that research needs to be done to identify how GIR should be done [11, 19, 21, 22, 29, 36]. One thing that most groups were in agreement on was that some type of Named Entity Recognition was necessary for GIR [5, 6, 7, 9, 10, 18, 19, 22, 23, 26, 27], but more work needs to be done on the Named Entity Recognition modules as they did not always work correctly [23, 27]. Also, from the work done it appears that overall manual expansion was not much better than automatic expansion, though it can improve individual queries [16, 34]. In addition, some forms of expansion such as meronym expansion were found to be detrimental to performance [2, 14, 26]. Other forms of expansion (e.g. holonym expansion) showed more promise [2, 3]. So, expansion is a method that shows some promise but it depends on what terms are actually added to the query. The results of previous work also show the lack of geographical resources can be a problem [6, 27] and that in some cases geospatial information was useful and the IR system worked well but more work needed to be done on the ranking functions [10, 18]. So, the previous work showed that geospatial information can be useful if one is careful about what information one uses and that more research needs to be done to see how the geospatial information could be used and on ranking results of GIR systems.

2.14 My Approach

My approach uses several of the methods and techniques described above and has several steps to it. My goal was to automatically disambiguate queries. My first step was to explore expansion techniques without using geographic reasoning. I tried using blind feedback. I also explored expanding terms using a geographic thesaurus built from information downloaded from the internet. I used the geographic thesaurus to expand queries as the SINAI group did [9]. For blind feedback, I explored using different weighting schemes, such as TF.IDF and Rocchio to weight the original terms of the query and newly added terms in the query. I also explored using a combination of blind feedback and thesaurus expansion. The third step in my approach was to attempt to do geographic disambiguation using longitude and latitude values for locations.

CHAPTER 3: METHODS

My goal in my experiments was too improve retrieval by using query expansion to disambiguate geospatial terms within the queries. My approach is similar to that of many of the groups that participated in GeoCLEF 2005 and GeoCLEF 2006 and builds on their work. I used the INQUERY Retrieval Engine [4] to perform retrieval. I stored geospatial information in a geographical thesaurus and used this geospatial information to perform four different experiments. The results of all of the experiments were compared to a baseline that does not use geospatial analysis. This comparison allows me to see whether retrieval has actually been improved.

This chapter first, in section 3.1, describes the test collection I used for my experiments, and then, in section 3.2, describes the geospatial data I used. Section 3.3 describes the evaluation measures for the experiments. Section 3.4 discusses how queries were processed prior to retrieval. Section 3.5 examines the expansion approaches I used in my experiments including blind feedback, re-weighting, geospatial expansion, and disambiguating the word “near.” Section 3.6 focuses on the programs and tools I used to run my experiments.

3.1 Test Collection

The experiments used the document collection and queries from GeoCLEF 2005. The geospatial information used was downloaded from the internet. The documents in the collection were newspaper articles from the

Glasgow Herald (1995) and the *Los Angeles Times* (1994). The *Glasgow Herald* collection was 119 MB large and the *Los Angeles Times* collection was 422 MB. Combining the articles from both newspapers gave 158037 documents, with the average document length being 531 words. The documents were not geographically tagged and did not contain any location specific information. They were all in English.

There were a total of 25 queries. The queries had an average of 35 relevant documents each. More specifically, five queries had fewer than 10 relevant documents, 9 queries had between 10 and 25 relevant documents, five queries had between 35 and 45 relevant documents and six queries had between 65 and 110 relevant documents. The GeoCLEF queries were provided with xml tags (See Figure 3.1 for a sample query).

As Figure 3.1 shows, each query had a number tag, an original number tag, a title tag, a description tag, a narrative tag, a concept tag, a spatial relation tag and at least one location tag. The number tag gave the number of the topic for GeoCLEF 2005. The orignum was the number that was first assigned to the topic when it was created; often the topics for GeoCLEF had been used in previous CLEF tracks. The title gave a brief overview of what the topic was about. The description field explained what sorts of documents were relevant. The narrative field gave a more detailed explanation of what relevant documents would contain for this query. The concept, spatial relation and location tags were added particularly for GeoCLEF. The concept tag gave the key concept for the query.

The spatial relation tag gave the main spatial relationship within the query. The location tag or tags gave locations mentioned within the query.

```
<top>
<num> GC001 </num>
<orignum> C084 </orignum>
<EN-title> Shark Attacks off Australia and California </EN-title>
<EN-desc> Documents will report any information relating to shark attacks on
humans.  </EN-desc>
<EN-narr> Identify instances where a human was attacked by a shark, including
where the attack took place and the circumstances surrounding the attack. Only
documents concerning specific attacks are relevant; unconfirmed shark attacks or
suspected bites are not relevant. </EN-narr>
<!-- NOTE: This topic has added tags for GeoCLEF -->
<EN-concept> Shark Attacks </EN-concept>
<EN-spatialrelation> near </EN-spatialrelation>
<EN-location> Australia </EN-location>
<EN-location> California </EN-location>
</top>
```

Figure 3.1: Sample query from GeoCLEF 2005 showing the different fields given of a GeoCELF query.

Each query when all of the given fields were considered had between 23 and 56 terms, with an average of 34.88 terms total. Each query had between 7 and 38 unique terms with an average of 18.24 unique terms. The queries based on all of the fields in a query were long and more precisely specified queries than shorter queries would have been. Unfortunately, most users are not willing to type in long sentences. They will enter on average 2 words, which as one can see is much shorter than the long queries. For this reason I also used queries based solely on the title field. These queries were shorter and thus more realistic than the long queries. Using only the title field created shorter queries with fewer repeating terms. When only the title was considered the queries had an average of

3.8 unique terms per a query. The average number of terms per a query was the same as the average number of unique terms.

3.2 Geospatial Information

The geospatial information was downloaded from the Geographic Names Database (GNDB) [38]. The information was separated into 251 files, each representing a country. Each country file had between 2 and 354157 records. Each record represented a location within the country and provided the region the location was in, its latitude and longitude, its feature classification (e.g., beach or valley), its populated place classification, the country it was in, and the administrative district (e.g., California is an administrative district of the United States) it was in. The data had six different regions: “Western Europe/ Americas,” “Eastern Europe,” “Africa/ Middle East,” “Russia/ Central Asia,” “Asia/ Pacific,” and “Vietnam.”

Figure 3.2 shows an example of the data for one city from GNDB. This example shows one of the challenges of using geospatial information. The data contains a lot of extraneous information. One has to look though the extra information to find the information one wants to use. Looking at figure 3.2 it is clear that additional information is needed to interpret the data. This is normal for geospatial information on the internet. Once the information is gathered one still needs some sort of key to interpret it and each place the information comes from will use a different format to represent the information. In addition, only some of the information provided is useful. In figure 3.2 only the information in bold proved useful for my experiments. I only used the RC (region classification), LAT (Latitude), LONG (Longitude), CCI (Country Code), ADM1 (An Administrative District), ADM2 (A second Administrative District) and the FULL_NAME_ND (full name of the location) parts of the data.

Labels	**RC**	UFI UNI **LAT** **LONG**DMS_LAT DMS_LONG UTM JOG FC DSG PC **CC1** **ADM1** **ADM2** DIM CC2 NT LC SHORT_FORM GENERIC SORT_NAME FULL_NAME **FULL_NAME_ND** MODIFY_DATE
Sample Data	**1**	-1307690 -1891634 **12.5333333** **-70.0** 123200 -700000 CP98 ND19-14 H STMI **AA** **00** N AFO ROOI Rooi Afó **Rooi Afo** 1993-12-21

Figure 3.2: Example data from GNDB showing one entry from a country file. Above the sample data from the country file are the labels associated with the parts of the data.

3.3 Evaluation Measures

To evaluate results of my experiments I used precision and recall (See section 1.2). Precision, as mentioned earlier, is the proportion of retrieved documents that are relevant, while recall is the proportion of relevant documents that have been retrieved. When evaluating results, one looks at the precision values at different recall levels and needs to strike a balance between recall and precision. One wants high recall because that means that relevant documents are actually being retrieved, but the average user will not be willing to look through very many documents so one needs high precision. I used the Signtest and Wilcoxon sing-ranks with p-value 0.05 to determine statistical significance

3.4 Query Processing

Prior to retrieval, the queries were processed. Queries were first analyzed to split them into their fields (e.g., title, description, narrative etc.). As queries were analyzed, stop words and stop phrases were removed from the queries as discussed in section 1.3. In addition, all words were converted to lowercase. Queries were then put into a format for INQUERY to read. The queries were also stemmed by INQUERY before retrieval was performed. In the case of blind feedback, possible feedback terms were collected and then the original query terms and the terms added by expansion were stemmed. In the geospatial experiments, the queries were analyzed to find any locations in them.

3.5 Expansion

Query expansion aims to improve retrieval by adding terms that will make the query more specific. In my experiments, I used several different approaches which will be described here. Some approaches specifically used geospatial information, others did not. I also combined some of the approaches in later experiments.

3.5.1 Blind Feedback

As explained in 1.7.3 blind feedback is a form of query expansion, where retrieval is performed on the queries in order to create a list of the top documents. A certain number of documents from this list are then assumed to be relevant and a certain number of words from these documents are added to the queries. My goal in using blind feedback was to determine whether blind feedback expansion is effective and if so with what parameters. Should queries be expanded with 5 terms? with 10 terms? with 50 terms? with 100 terms? When is the query expanded so much that it hurts retrieval? My goal was to figure out how many terms I could add to the query while still improving retrieval effectiveness. I also looked at how many documents should be considered relevant when using blind feedback to expand a query. I varied the number of documents that were considered relevant and the number of terms added to the query between 5, 10, 25, 50, and 100.

3.5.2 Re-weighting

Re-weighting queries, as discussed in section 1.8, is when terms within the query and that are added to the query are given new weights to show what words the system considers important. There are many different ways that the terms can be re-weighted. Here I will explain the different ways that I used.

Below I describe the weighting schemes I used, but first I will define some components of the weighting schemes. TF is the number of times a term appears in a documents (as discussed in section 1.2). IDF is inverse document frequency or how many documents a term is in overall (as discussed in section 1.2).RTF is the sum of the frequencies within the relevant documents for an individual term. RDF is the number of relevant documents that a term is in. Alpha, beta and gamma are constants which modify the other values. For my experiments, alpha was set to 1.00, beta to 2.000 and gamma to 0.5000.

The primary weighting scheme I used was Rocchio, which was discussed earlier in sections 1.8 and 2.2.5. The version of Rocchio, I used had the formula beta*(avg rel bel) – gamma* (avg nonrel bel). In the formula, avg stands for average, rel stands for relevant document, nonrel stands for non-relevant document, and bel stands for belief. The equation takes the sum of the relevant documents belief and divides it by the number of relevant documents to get an average. The second half of the equation does the same thing with non-relevant documents. Then the average for relevant documents is modified by the constant beta and the average for non-relevant documents is modified by gamma. The non-relevant value is then subtracted from the relevant document value in order to down weight the score for the relevant documents. This is done because terms will occur in both relevant and non-relevant documents, so one wants to take into account how often it appears in both. By subtracting the non-relevant document value from the relevant document value, one gets a weight that takes into account whether the term can distinguish between relevant and non-relevant documents.

I compared the results of Rocchio to the baseline and the results of 6 other weighting schemes:

1) logrtfidf (beta*log(rtf)- gamma*log(rtf) * idf)

2) rdf (number of relevant documents the term is in)

3) rdfidf ((beta*rdf-gamma*nrdf)*idf)

4) rtf (sum of tf in relevant docs)

5) rtfidf ((beta*rtf – gamma*nrtf) *idf)

6) tfidf (tf*idf) (tf is for entire collection)

Experimenting with re-weighting terms included expanding using blind feedback. During blind feedback, the weighting scheme re-weighted the original query terms as well as giving a weight to the terms added by blind feedback. The only thing that changed between different runs was which weighting scheme was used to decide the terms that should be added. Different schemes added different words to the queries and some did better than others, as will be discussed in section 4.3.

3.5.3 Geospatial Expansion

The third approach that I explored was adding terms from my geospatial database to the queries. Figure 3.4 shows a diagram of how the three experiments with using the geographic thesaurus to expand queries worked. The belief was that if the geospatial terms, primarily locations, could be disambiguated and made more precise by adding terms from my thesaurus, retrieval would be improved. I experimented with adding regions and administrative districts of countries to the queries. See Figure 3.3 for an example of terms that might be added given a specific geospatial term.

Term: Berlin	Term: Germany
Region: Americas/ Western Europe	Region: Americas/ Western Europe
Administrative District: Berlin	Administrative Districts: Germany, Baden-Wurttemberg, Bayern, Bremne, Hamburg, Hessen, Niedersachesen, Nordrhein-Westfalen, Rheinland-Pfalz, Saarland, Schleswig-Holstein, Brandenburg, Mecklenburg-Vorpommern, Sachsen, Sachsen-Anhalt, Thuringen, Berlin

Figure 3.3: Examples of locations and geospatial terms that might be added to the queries for a term.

3.5.4 The word “near”

Another approach I tried was disambiguating the term “near”. “Near” is an ambiguous word. For different locations and different people, it can mean different things. My goal was to discover what definition of “near” worked best for different types of locations. I used a distance formula that converted longitude and latitude into an approximate distance between two places. I varied how “near” was defined and added cities considered near to the query. I did this to see what distance might be considered “near”. For instance, was 5 miles near, 10 miles, 25 miles, 100 miles, 500 miles, etc? I added the locations that fell within the range of “near” to the query. The goal here was to discover how “near” could be more precisely defined.

3.5.5 Combinations of Previous Approaches

The approaches described above can be done individually. They can also be combined to hopefully improve retrieval more than using only one approach. Figure 3.4 shows the different ways that geospatial information could be combined with other approaches.

The first way method of expansion shown in Figure 3.4 is the one described in section 3.5.3. The second and third methods combine sections 3.5.3 and section 3.5.1. The second method starts by expanding using geospatial terms from the thesaurus as described above and then performs blind feedback on the queries this had generated (Figure 3.4). The last method changes the order of the operations. First the queries are expanded through blind feedback and then they are expanded with geospatial terms from the database (Figure 3.4).

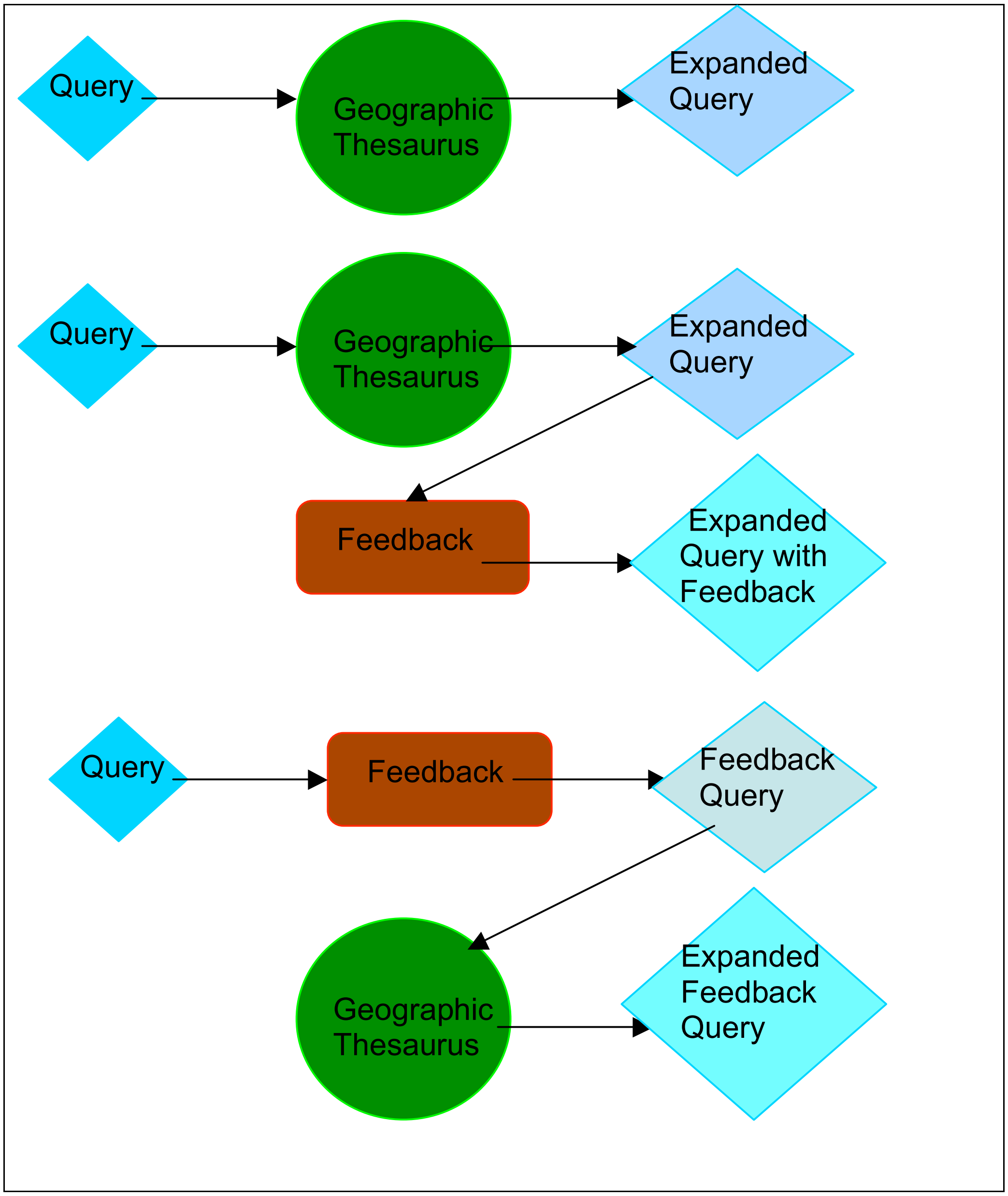

Figure 3.4: Diagram of three different ways to use information from geospatial thesaurus or database to expand the queries. Shows combinations of the different methods mentioned above.

The goal of combining these methods is to see if retrieval is improved by using more than one method and if the methods should be used in a certain order. Does it matter if blind feedback is done first or if the queries are expanded with geospatial terms first?

3.6 Tools

3.6.1 Tools already created

I used three programs that had previously been created by others. The INQUERY Retrieval Engine [4] took a file of queries and relevance judgments for those queries, performed retrieval using an already created database of documents and returned a file with evaluation information and, if requested, a list of the top n documents for each query.

The second program I used was trec_eval. This program took evaluation files created by INQUERY and compared the retrieval results specified by the different evaluation files. Trec_eval printed information about each query, including the precision and recall values and the percent change between the different runs. In addition, it summarized the average results over all queries and performed three different statistical tests, where p = 0.05, Signtest, T-test and Wilcoxon test, on the results in order to show whether the differences between results were significant or not. The T-test assumes that the data has a normal curve, which is not true for my data so it did not provide reliable information for my experiments. I used the results of the other two tests to determine if a difference between runs was significant.

The third program I used was feedback, which performed blind feedback on queries. It allowed me to specify how many documents should be assumed to be relevant, how many terms should be added and what weighting scheme should be used to weight the current query terms and terms to be added to query. It provided all of the weighting schemes that I used in my experiments.

3.6.2 Tools I created

In addition to the three programs already created, I also wrote several programs to perform tasks that were necessary for my experiments. All of these programs were written in Java.

3.6.2.1 Format Converting Tools

I wrote two programs that put files in the correct format for INQUERY and feedback. One program converted a list of top documents from the format created by INQUERY into a format that feedback could use. Figure 3.5 shows the two different formats.

Query count 1 doc_cnt: 158037	
doc LA121594-0267 has rank: 1 (0.459113)	1 LA121594-0267 1
doc LA121594-0181 has rank: 2 (0.458293)	1 LA121594-0181 1
doc LA122194-0180 has rank: 3 (0.456510)	1 LA122194-0180 1
doc GH950614-000127 has rank: 4 (0.455631)	1 GH950614-000127 1
doc LA041794-0356 has rank: 5 (0.455279)	1 LA041794-0356 1

Figure 3.5: Examples of top document file formats. On the left is the beginning of a top documents file created by INQUERY. On the right is the same file converted into the format that feedback expects.

3.6.2 Query Processor

The third program I wrote was a query processor to process the query and prepare it to be sent to INQUERY. It read in a file of queries and created a new file of queries in the correct format for INQUERY. The program could create queries using the different fields of the queries. For instance, I used it to create queries using different combinations of the fields for the baseline (See section 4.1 for more details). As an example of formats, I had the program create queries using all of the fields of the queries and also using only the title field. These were the two formats I used in my experiments. As well as choosing different fields to form the query, the query processor also removed stop words and stop phrases from the queries (As discussed in section 1.3). The query processor program also interacted with the geospatial database in order to expand queries with geospatial terms.

3.6.3 Geospatial Database or Thesaurus

The geospatial database or geospatial thesaurus stored and accessed geospatial information downloaded from the internet. It created the database by reading in files of locations within countries. It stored information about regions, countries and administrative districts of countries, while writing city names to inverted indexes. One inverted index was used for each letter of the alphabet. So, all city names that start with an “a” would be in the same file while those that start with “t” would be in a different file. These city files were used when the database needed to find a location that was not a country, region or administrative district, while the data stored in the database itself could be used for countries, regions, and administrative districts.

Once the database was created it interacted with the query analyzer to facilitate the expansion of queries with geospatial terms from the database. Figure 3.6 shows a diagram of the interaction between the queries and the database.

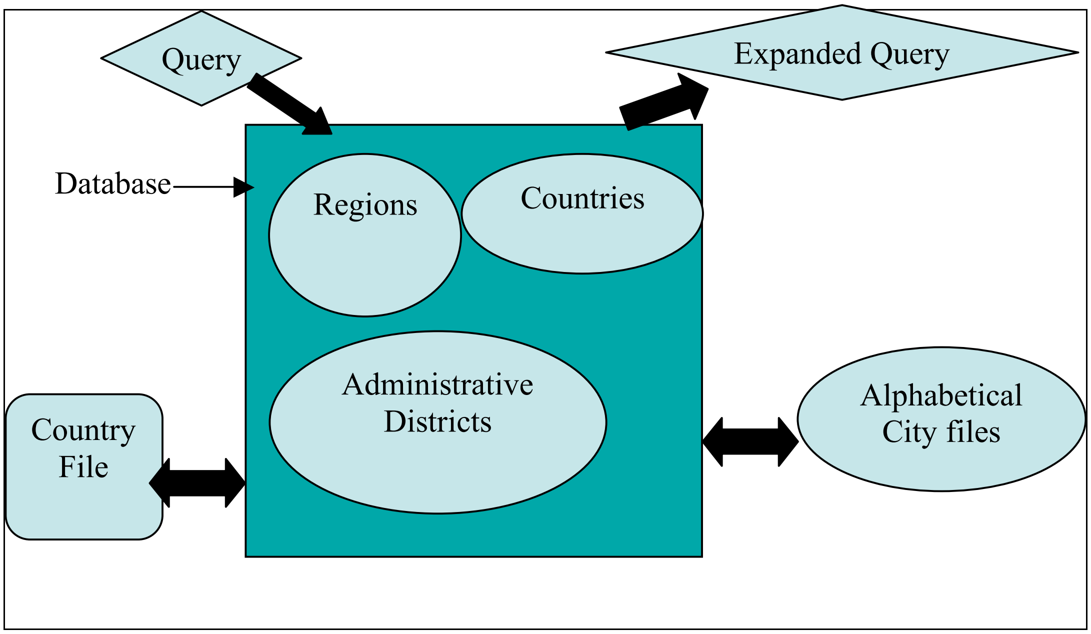

Figure 3.6: Diagram of the geospatial database and the process of expanding queries.

The query analyzer sent possible geospatial locations from the location tags of the queries or other parts of the query to the database program. The program then used the database to figure out which sort of location the term referred to and what type of terms would be appropriate to add. For instance, the location "Germany" would be determined to be a country, which would mean that terms to be added might include the regions it was in and the administrative districts of the country. After determining what sort of terms could be added, terms would be added and sent back to the query analyzer which once it had expanded all of the location words for the queries could then produce an expanded version of the queries. See Figure 3.6 for a diagram of the interaction between the queries and the database.

CHAPTER 4: EXPERIMENTS AND RESULTS

This chapter describes the experiments and their results. Section 4.1 describes the baseline used. Section 4.2 describes the blind feedback experiments. Section 4.3 discusses the re-weighting experiments. Section 4.4 examines the four different geospatial experiments done and compares their results to the results of the baseline and the blind feedback experiments. Lastly, section 4.5 summarizes the results presented in this chapter.

4.1. Baseline

The first set of experiments was designed to create a baseline. This was done so that later experiments would have something to be compared to in order to see if they improved retrieval. Thirteen different formations of the unmodified query were run to see which had the best results. The thirteen different formations tested were:

1) allstp – Uses all of the fields from the query
2) descnarrstp – Uses the description and narrative fields of the query
3) descnarrtagsstp – Uses the description, narrative, concept, spatial relation and location fields from the query
4) descstp – Uses the description field of the query
5) desctagsstp- Uses the description, concept, spatial relation and location fields of the query
6) narrstp – Uses the narrative field of the query

7) narrtagsstp– Uses the narrative, concept, spatial relation and location fields of the query
8) tagsstp– Uses the concept, spatial relation and location fields of the query
9) titledescnarrstp– Uses the title, description and narrative fields of the query
10) titledescstp– Uses the title and description fields from the query
11) titlenarrstp- Uses the title and narrative fields of the query
12) titlestp– Uses the title field of the query
13) titletagsstp– Uses the title, concept, spatial relation and location fields of the query.

Stop words were removed from the queries. Recall from Section 1.3 that stop words are words that are common in most documents and have little discrimination power, thus are not very useful separating relevant from non-relevant documents. Stop phrases were also removed from queries. Like stop words, stop phrases are not content bearing. Stop phrases were removed because when examining the queries it was noticed that many queries had phrases that would be unlikely to improve retrieval (See Figure 4.1 for some sample stop phrases). Though it should be noted this removal only affected queries which used the narrative and description fields of the topics. The other fields of the topics did not have any stop phrases in them. As shown in Figure 4.1, stop phrases were phrases like “a relevant document…” The documents that are relevant will most likely not say “a relevant document.” So, it is useful to remove stop phrases.

a relevant document will provide information
all documents which
are relevant
are not relevant
articles reporting
documents will report any information relating to
identify instances where
not relevant are stories which
relevant documents describe

Figure 4.1: Examples of stop phrases that were removed from queries

	Stop Phrases Left in	Stop Phrases Removed	% Difference
Average Precision	0.3873	0.3915***	1.08%

Table 4.1: Average precision for queries with and without stop phrases
The table shows the average precision for queries based on all fields with and without stop phrases removed and the percent difference between the two average precisions. *** Represents a statistically significant difference according to the Wilcoxon and the Signtest.

The removal of stop phrases is a standard retrieval approach. Runs were done with stop phrase both removed and not removed in order to confirm that removing stop phrases improves retrieval. As seen in Table 4.1, the queries with stop phrases removed did statistically significantly better, according to both Wilcoxon and Signtest, than the queries with stop phrases included. Obviously removing stop phrases is beneficial for retrieval. So for all other experiments stop phrases were removed.

For the baseline experiments weighted and un-weighted versions of the queries were run. The weighted version used the Rocchio weighting scheme. The point of running both weighted and un-weighted was to be able to choose one of them as the baseline. As shown in Table 4.2, there is not a significant difference between the un-weighted and weighted queries. Since there was no significant difference between the two, the weighted queries were chosen as the baseline because the results of later expansion experiments were weighted.

	Un-Weighted	Weighted	% Difference
Average Precision	0.3915	0.3868	-1.2%

Table 4.2: Average precision for un-weighted and weighted queries
The table shows the average precision for un-weighted and weighted queries based on all fields and the percent difference between the two average precisions.

The results for the experiments using the 13 different query formations produced showed that queries using all of the fields and queries using the title, description and narrative fields had the best results. There was no significant difference between the two. This may indicate that using the tags to form the query is not necessary since that was the only difference between those two formulations. This is useful to know because most queries do not have tags. Tags are added manually which makes their presence unrealistic.

Because of these results, for the rest of the experiments three query formats were used for the baseline. For all three baseline query formations, queries were weighted and stop phrases were removed. The first one was longer queries using all of the fields in the query (henceforth referred to as long queries). Experiments using all fields except the tags (non-tag long queries) were also run. The results of these experiments were very similar to the long query results. Remember, tags are manually added to queries and thus unrealistic. The tags did not provide any new index terms for the query. So, removing tags results in a query with the same terms as the long query. However, tags may prove useful for query analysis later. Since there was little difference between the results of the two I will only discuss the results for the long queries.

Unfortunately, most users will not write a query that is several sentences long. The average user creates two word queries. Since the long queries were unrealistic for the average user, experiments were run with a more realistic format of the query. This format was a shorter query format that used only the title field, thus giving a shorter more realistic query (henceforth referred to as the short queries). The experiments were done with both long and short query formations because using only the short queries is more realistic for the average user as those queries were only a few words long. So, the short queries are closer in length to real user queries than the long queries.

4.2 Blind Feedback

Recall from sections 1.7.3 and 3.5.1, expansion by blind feedback is assumed to generate a more precise query that will improve retrieval. The purpose of blind feedback is to add words from documents that one assumes to be relevant to the query in order to improve retrieval. This section discusses the experiments done using blind feedback.

All expanded queries were formed from the baseline. All of the resulting expanded queries were weighted using Rocchio. For these experiments no geospatial information was used. The long and short queries were used. To test how many documents should be considered relevant and how many terms should be added for each query formation, 25 runs were done. The training set was 100 documents and was generated by performing retrieval on the baseline queries to create a list of the top 100 documents. The number of documents from the training set considered relevant and the number of terms added to the queries were varied between 5, 10, 25, 50 and 100. Every possible combination of these two variables was tried. For the long queries, the runs was named afbndmt (the short queries were titlefbndmt), where n was the number of documents used and m was the number of terms added to the queries. For example, afb5d5t was expansion of the long queries with the top 5 terms from the top 5 docs. Section 4.2.1 presents and discusses the results for blind feedback using the long queries and section 4.2.2 presents the results for the short queries.

4.2.1 Long Queries

In general for the long query blind feedback experiments the more terms the query was expanded with and the more documents that were assumed to be relevant the worse the results were. Displayed in Table 4.3 are the top 3 results for blind feedback expansion with long queries, which were obtained by assuming that the top 5 documents were relevant and expanding with 5, 10 or 25 words. Figure 4.2 shows a graph of the long query baseline and the top 3 blind feedback runs.

Recall	Precision			
	Long Query Baseline	afb5d5t	af5d10t	afb5d25t
0	0.8307	0.8438	0.7955	0.7979
0.1	0.6658	0.703	0.6643	0.6738
0.2	0.6185	0.6505	0.6280	0.6130
0.3	0.5363	0.5917***	0.5536	0.5489
0.4	0.4491	0.4930***	0.4672	0.4760
0.5	0.3948	0.4235	0.4119	0.4419
0.6	0.3137	0.3489	0.3642*	0.3723
0.7	0.2453	0.2700	0.2709	0.2545
0.8	0.1799	0.2049	0.2057*	0.2016
0.9	0.1243	0.1371*	0.1399	0.1373
1	0.0676	0.0767	0.0739	0.0805*
Avgerage precision	0.3868	0.4175*	0.4035	0.4024
# documents returned	Precision			
5 docs	0.5760	0.6240	0.6080	0.6080
10 docs	0.4600	0.4920	0.4800	0.4760
15 docs	0.4053	0.4347	0.4267	0.4213
20 docs	0.3580	0.3840	0.3820	0.3760
30 docs	0.2987	0.3147	0.3187	0.3133
100 docs	0.1716	0.1804*	0.1808***	0.1848***
200 docs	0.1134	0.1158	0.1170	0.1168
500 docs	0.0558	0.0572	0.0575*	0.0581***
1000 docs	0.0310	0.0317	0.0318	0.0320*

Table 4.3: Results for long query blind feedback experiments. The table shows the recall and precision values for several of the blind feedback runs on the long queries. * indicates a statistically significant difference according to Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column on the left shows recall levels. The second column shows the precision values for the long query baseline. The third column (af5d5t) shows the precision values for doing feedback with 5 documents and 5 terms. The fourth column (afb5d10t) shows the precision values for doing feedback with 5 documents and 10 terms. The final column (afb5d25t) shows the precision values for doing feedback with 5 documents and 25 terms.

As seen in Figure 4.2 adding 5, 10 or 25 terms improved retrieval over the baseline. However, as can be seen in Table 4.3, expanding with 5 terms was the only run that did statistically significantly better than the baseline, according to the Wilcoxon test. It was 7.91% better than the baseline. Expanding with 10 words or 25 words does not do as well as adding 5 terms. Adding 10 or 25 words did better than the baseline by about 4%, but these improvements were not statistically significant. However, as shown in table 4.3, expanding with 25 words does statistically significantly better than the baseline when looking at the precision for 100, 500, and 1000 documents.

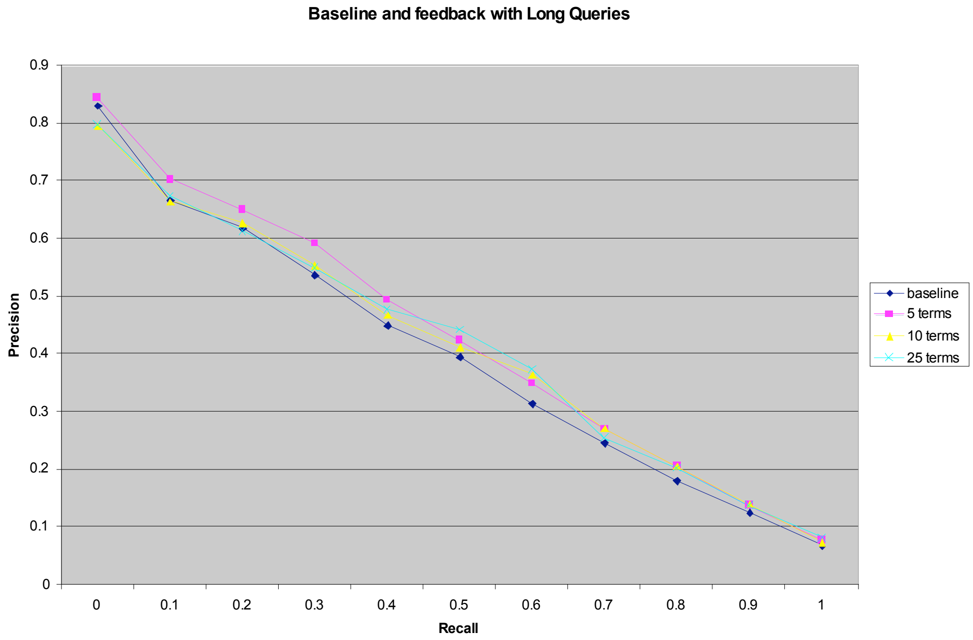

Figure 4.2: Graph of baseline and feedback with long queries. The graph shows the precision and recall for the long query baseline, the feedback with 5 documents and 5 terms results, the results for feedback with 5 documents and 10 terms, and the results for feedback with 5 documents and 25 terms.

The fact that adding 25 terms did better at higher recall levels implies that adding more words would favor recall. One explanation for this is that as more relevant documents are retrieved there is more chance that the terms added to the query will be in relevant documents. Unfortunately, users most likely would not want to look through that many documents. So, improving the precision when 100, 500, or 1000 documents have been returned is not helpful for most users.

Another thing shown by the results of the long query experiments is that using more documents to select terms and adding more terms is bad for retrieval. The results' getting worse as the number of documents used went up can be explained by there being few relevant documents in the collection. Recall, from section 3.1, that most of the queries used have fewer than 100 relevant documents. This means that as the number of documents goes up so does the percentage of those documents that are not relevant. So, in this environment the more documents that are assumed to be relevant, the fewer will actually be relevant.

The same trend was shown in adding terms. The fewer terms added the better the results. One explanation is that there are few relevant documents, so adding a lot of terms will likely draw in more non-relevant terms. Also, the long queries are already fairly well specified so it is harder to find relevant terms that might help the query. So, it seems for this set of queries the more terms that were added, the more likely it was that the terms being added would not be relevant.

The blind feedback on the long queries established that adding 5 terms from the top 5 documents was the most beneficial to retrieval. So, for later experiments involving blind feedback, the long queries will be expanded with 5 terms from 5 documents.

4.2.2 Short Queries

Recall that users are more likely to enter a few terms for a query. Thus blind feedback was done with short queries to give a more realistic idea of blind feedback's effect on retrieval. Table 4.4 displays the short query baseline and the top 3 results for blind feedback on short queries.

Recall	Precision			
	Short Query Baseline	Tfb5d5t	Tfb5d10t	tfb5d25t
0	0.7180	0.7186	0.7006	0.6780
0.1	0.5926	0.6325	0.6317	0.6037
0.2	0.4958	0.5366	0.5549	0.5266
0.3	0.4450	0.4795	0.4965	0.4858
0.4	0.3693	0.4394	0.4413*	0.4153
0.5	0.3375	0.3945	0.3857***	0.3618
0.6	0.2735	0.3168	0.3254	0.3054
0.7	0.2281	0.2580	0.2514	0.2524
0.8	0.1770	0.2141***	0.2034***	0.1699
0.9	0.1236	0.1501*	0.1461*	0.1356
1	0.0633	0.0862	0.0884*	0.085
Average Precision	0.3337	0.3699	0.3675*	0.3488
# Documents Retrieved	Precision			
5	0.4480	0.4960	0.5200***	0.5040
10	0.4320	0.4320	0.4640	0.4400
15	0.3813	0.3920	0.4027	0.3867
20	0.3260	0.3520	0.3580	0.3520
30	0.2653	0.2973*	0.2960	0.2840
100	0.1564	0.170	0.1744***	0.1728**
200	0.1094	0.1148***	0.1150*	0.1152***

Table 4.4: Short query blind feedback results. The table shows the results for top four blind feedback runs using the short queries. * indicates a statistically significant difference according to the Wilcoxon test. ** indicates a statistically significant difference according to the Signtest. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column on the left shows recall levels. The second column shows the precision values for the long query baseline. The third column (tf5d5t) shows the precision values for doing feedback with 5 documents and 5 terms. The fourth column (tfb5d10t) shows the precision values for doing feedback with 5 documents and 10 terms. The final column (tfb5d25t) shows the precision values for doing feedback with 5 documents and 25 terms.

The top 3 results all assumed that 5 documents were relevant and added the top 5, 10 or 25 terms to the short queries. All three of these improved the average precision over the baseline. Figure 4.3 shows a recall precision graph with the short query baseline, adding 5 terms, adding 10 terms and adding 25 terms. As seen on Figure 4.3, adding 5 or 10 terms does better than adding 25, but as shown in Table 4.4, adding 10 terms was the only one that did statistically significantly better than the baseline.

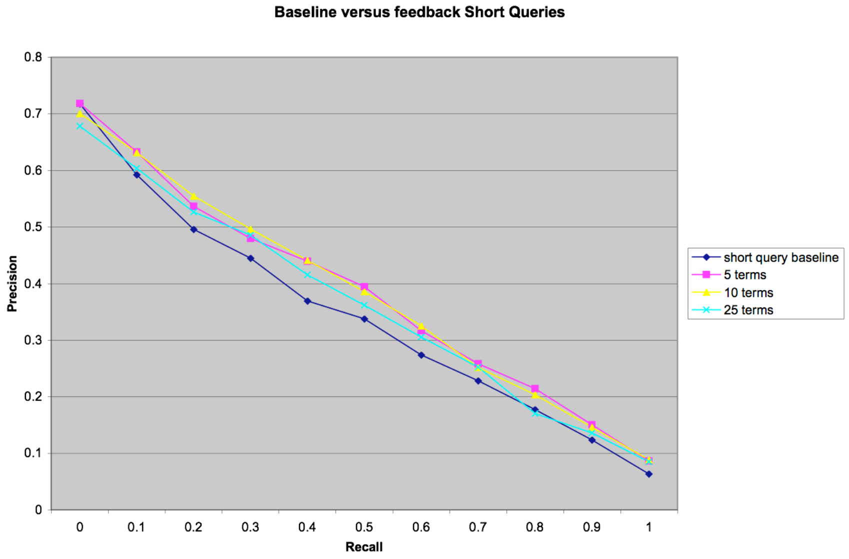

Figure 4.3: Graph of baseline and feedback with short queries. The graph shows the precision and recall for the short query baseline, the feedback with 5 documents and 5 terms results, the results for feedback with 5 documents and 10 terms, and the results for feedback with 5 documents and 25 terms.

The results for the short queries show some similarities and differences when compared to the long query results. In both cases, having fewer relevant documents and adding fewer terms was better. But for short queries, unlike for long queries, adding 10 terms was actually better than adding 5. One possible explanation for this difference in results is the fact that the short queries had fewer terms to begin with, so there was more room for expansion than in the long queries. The long queries were already fairly well specified, so adding terms had less of an effect than adding terms to the less well specified short queries.

So, from the blind feedback experiments on short queries it can be seen that the short queries did best when feedback was done with 5 documents and 10 terms. For the rest of the experiments, using blind feedback short queries will use 5 documents to select the top 10 documents.

4.3 Re-weighting Experiments

In addition to varying the number of documents assumed to be relevant and the number of terms added to queries, I also experimented with different weighting schemes for choosing those terms. The goal was to improve retrieval by re-weighting the queries terms to give more weight to terms based on their frequency in relevant documents. Several weighting schemes were tried in order to see which one was the most beneficial to retrieval. The weighting schemes I used as discussed in section 3.5.2 were:

1) logrtfidf (beta*log(rtf)- gamma*log(nrtf) * idf)

2) rdf (number of relevant documents the term is in)

3) rdfidf ((beta*rdf-gamma*nrdf)*idf)

4) rtf (sum of tf in relevant docs)

5) rtfidf ((beta*rtf – gamma*nrtf) *idf)

6) tfidf (tf*idf) (tf is for entire collection)

7) Rocchio- beta*(avg rel bel) – gamma* (avg nonrel bel).

In all cases, alpha was set to 1.00, beta to 2.000 and gamma to 0.5000. For the statistical tests performed on the results, Wilcoxon and Signtest, p= 0.05. Based on the blind feedback experiments discussed in section 4.2, for each weighting scheme, the top 5 documents were assumed to be relevant and the queries were expanded with 5 terms for long queries or 10 terms for short queries.

4.3.1 Long Queries

As shown in Table 4.5, for the long queries only those queries that used Rocchio as the weighting scheme for selecting expansion terms had a statistically significant improvement, based on the Wilcoxon test, when compared to the weighted baseline. They were 7.91% better than the baseline. In addition, rtfidf was statistically significantly worse than the weighted baseline.

	Long Query Baseline	logrtfidf	Rdf	rdfidf	rtf	rtfidf	Tfidf	Rocchio
Average Precision	0.3868	0.3784	0.3975	0.3908	0.3957	0.3312***	0.3924	0.4175*

Table 4.5 Average precision for the long query baseline and the 7 different weighting schemes. The table shows the average precision for the long query baseline and the 7 different weighting schemes tried. Each column shows a different weighting scheme. * indicates a statistically significant difference from the baseline according to the Wilcoxon test. *** indicates a statistically significant difference from the baseline according to both the Wilcoxon and Signtest.

Since Rocchio and rdf gave the best results, the two of them were also compared to each other. In comparing the Rocchio and rdf runs it was found that there was no significant difference in average precision. However, Rocchio did statistically significantly better at recall points of 0.0, 0.2, 0.3, 0.5, 0.8, and 0.9 as well as having significantly higher precision for 5 documents, 15 docs (documents), 20 docs and 100 docs. This indicates that Rocchio is the better weighting scheme to use, especially since it did better when fewer documents had been retrieved. This is important because users might not be willing to look through very many documents. So, realistically one can't expect the average user to go through 100 documents. Rocchio does a better job overall when fewer documents have been retrieved.

Looking at the results query by query, Rocchio had a higher average precision than rdf on 14 of 25 queries and rdf did better on 11 queries. A query by query analysis shows that, Rocchio and rdf added on average 0.92 of the same terms to each query. Where terms differed, rdf tended to add more general terms while Rocchio added more specific ones. For instance, in the query "Shark Attacks off California and Australia," Rocchio added the terms "klimley" (a Google search revealed this to be the last name of multiple authors some of whom have written on sharks), "species", "oceanography", "Rosenblatt" and "bite". Rdf added "base", "great", "species", "bite", and "witness". "Klimley", "oceanography" and "rosenblatt" are more specific than "base", "great" and "witness." For this case, Rocchio did better showing that more specific terms being added is useful in some cases, but there could also be cases where the terms added are too specific and thus do not help retrieval. Rdf may have done better on some queries because for those queries the more specific terms added by Rocchio were not included in many relevant documents while the more general terms rdf used were more commonly found in relevant documents. The number of relevant documents was roughly the same for where Rocchio or rdf did better.

One advantage of Rocchio over rdf it that is tended to give higher precision at low recall levels. Table 4.6 shows the average precision at 5 docs, 10 docs, 15 docs, 20 docs, and 30 docs.

	Rdf	Rocchio	% difference
5 docs	0.5760	0.6240**	8.33%
10 docs	0.5000	0.4940	-1.2%
15 docs	0.4027	0.4347***	7.9%
20 docs	0.3460	0.3840*	10.98%
30 docs	0.3000	0.3147	4.9%

Table 4.6: Rdf and Rocchio comparison when few documents have been retrieved The table shows the average precision at 5 docs, 10 docs, 15 docs, 20 docs and 30 docs for rdf and Rocchio. The last column shows the % difference between the rdf and Rocchio.

Rocchio and rdf tied more often than not when few documents were retrieved. However, when considering only15 or 20 documents, Rocchio does better on 10 queries. So, if average users only look at the first few documents returned, Rocchio is a better choice for term selection.

For the long queries, Rocchio had the best results for the weighting schemes. One explanation for why the other schemes do not do as well is that many of the variables they use are not as useful for these experiments or are combined in ways that make them less useful. For instance, recall that rtf takes the sum of the term frequencies across all relevant documents. This gives more weight to long documents where the term appears many times than to shorter documents where it does not appear as many times. On the opposite side of the spectrum is rdf which is the number of relevant documents a term appears in. This does not take into account how many times a term occurs in a document. One other explanation for weighting schemes with rdf and rtf in them not doing as well as Rocchio is that not all of the documents assumed to be relevant actually are. For long queries when 5 documents have been retrieved precision is .576. This means that about 3 of the top 5 documents are actually relevant. So, looking at the top five documents introduces some non-relevant terms into the queries and the weighting schemes which take into account relevant and non-relevant documents will be considering some non-relevant documents and terms.

Rocchio was the only weighting scheme that statistically significantly improved retrieval over the baseline. One reason for this is that it succeeded in choosing more specific query terms to add to queries than other weighting schemes, like rdf, did. Based on the results discussed in this section, Rocchio was chosen as the weighting scheme to be used in the rest of the experiments.

4.3.2 Short Queries

Using the short queries showed slightly different results than using the long queries. Remember in section 4.2.2 it was found that using the top 5 documents to select 10 terms to expand the query with was best for short queries. Thus the re-weighting experiments assume the top 5 documents are relevant and expand queries with the top 10 terms.

Table 4.7 shows the average precision for the seven weighting schemes and the short query baseline. As the table shows, when 10 terms were added to the queries, three weighting schemes showed a statistically significant difference from the baseline. Rocchio had an average precision 10.14% higher than the weighted baseline; this was statistically significant according to the Wilcoxon test. Logrtfidf and tfidf did statistically significantly worse than the weighted baseline according to the Wilcoxon test.

	Short Query Baseline	logrtfidf	Rdf	Rdfidf	rtf	rtfidf	tfidf	Rocchio
Average Precision	0.3337	0.3036*	0.3535	0.3262	0.3493	0.3192	0.3068*	0.3675*

Table 4.7 Average precision for the short query baseline and the 7 different weighting schemes. The table shows the average precision for the short query baseline and the 7 different weighting schemes tried. Each column shows a different weighting scheme. * indicates a statistically significant difference from the baseline according to the Wilcoxon test.

The Rocchio weighting scheme again gave the best results. This may be because it gives the best combination of different variables to consider. It looks at tf and idf scores and looks at relevant documents while also taking into account non-relevant documents. So, it takes into account how many documents a term is in overall, how many times it appears in a document, the length of the documents and whether the term is in document assumed to be relevant, in non-relevant documents or in both. On the other hand, it appears that logrtfidf is not a good weighting scheme to use for this environment. One reason for this may be that rtf is not the best variable to use in connection with idf, perhaps because combining them gives more weight to longer documents since they will have the terms occurring more frequently. Idf, downweights rtf by taking into account how many documents a term appears in, but it does not take into consideration how long the documents are. In addition, as mentioned for long queries, some of the assumed relevant documents are not actually relevant. For short queries about 2 documents out of the top 5 are actually relevant. This means that even more than for long queries, the weighting schemes may be giving more weight to non-relevant terms than would be ideal. Tfidf also did worse on the short queries. One explanation for this is that it does not take into account whether terms occur in relevant documents or non-relevant documents. Even if all of the top 5 documents are not all actually relevant, it is still beneficial to take into account that a higher percentage of them are relevant than the percentage of relevant documents over the entire collection. So, taking relevant documents into account is helpful.

So, from these results it would seem that Rocchio is the best weighting scheme to use with either short or long queries. In the case of short queries, 10 terms are best for expansion. Again as discussed in section 4.2, this may be because there is more room for new terms to improve the short queries than the long queries.

4.4 Geospatial Experiments

The goal of the geospatial experiments was to use geospatial information to improve retrieval. There were four different geospatial experiments. The first one expanded queries using the geographic thesaurus, adding administrative districts and regions for country locations found in the queries. The second experiment did feedback after expanding queries with geographic thesaurus. The third experiment switched these two steps and expanded blind feedback queries with data in the geographic thesaurus. The fourth experiment used longitude and latitude to define “near” and added cities to the queries based on their distance from cities mentioned in the query. Each experiment was done with long queries and short queries. The goal of these experiments was to see how adding geospatial terms affected retrieval as compared to the baseline and the blind feedback results.

4.4.1 Geographic Thesaurus Expansion

The experiments with finding expansion terms in the geographic thesaurus aimed to exploit the geospatial information within the queries in order to improve retrieval. Another goal was to see whether the types of relationships used to expand the queries improved retrieval. The first experiment expanded queries using the geographic thesaurus. For the long queries, the terms in the location tags were used to select expansion terms. Each location tag was checked for a country; if it contained a country, then that country was used to choose expansion terms from the geographic thesaurus. For the short queries, the words from the title field were used to identify expansion terms. The title field, like the tag field of long queries, was checked for countries; countries found were used to choose expansion terms. The words added were regions and administrative districts of countries. An example of a region that might be added was “Western Europe/ Americas.” Administrative districts referred to areas within a country. For instance, the individual states within the United States would be considered administrative districts. There were three different expansion runs done. In each case, countries were identified via location tag (for long queries) or via query analysis if tags were not present and expansion terms were chosen according to one of 3 different types of relationships to country information. The first set of queries was expanded with administrative districts and regions. The second set of queries was expanded with administrative districts. The third set of queries was expanded with regions.

In section 4.4.1.1 the results of using terms from the geographic thesaurus to expand long queries are discussed. In section 4.4.1.2 the results of using geographic thesaurus terms to expand short queries are discussed.

4.4.1.1 Long Queries

Tale 4.8 shows the baseline and the three runs for expanding via the geographic thesaurus. The results of expanding the queries with geographic thesaurus terms overall show a slight improvement over the baseline. There is not a statistically significant difference in average precision for any of the three runs.

When a query by query analysis is done for the three runs, adding regions and administrative districts does better on 9 queries, adding administrative districts does better on 9 queries and adding regions does better on 13 runs. Unfortunately, adding administrative districts and regions does worse on 15 queries, adding administrative districts does worse on 15 queries and adding regions does worse on 12 queries.

That all three runs did slightly better than the baseline is encouraging because this shows that, perhaps with more research, adding terms from the geographic thesaurus might improve retrieval more. The administrative districts and regions do not appear to help retrieval, but they do not hurt it, either. This implies that some of what they are adding might be useful. Administrative districts add a lot of terms. It might be beneficial to filter out some of the administrative districts so that only a few were added to the queries. By looking at the context of the queries, there might be a way to select only some of the administrative districts.

	Long Query Baseline	Administrative Districts Regions	Administrative Districts	Regions
Recall	Precision			
0	0.8307	0.8183	0.8183	0.8215
0.1	0.6658	0.6615	0.6616	0.6804
0.2	0.6185	0.6031	0.6024	0.6151
0.3	0.5363	0.5425	0.5418	0.5437
0.4	0.4491	0.4587	0.4581	0.4532
0.5	0.3948	0.3927	0.3914	0.3874
0.6	0.3137	0.3371*	0.3354	0.3260
0.7	0.2453	0.2504	0.2474	0.2540
0.8	0.1799	0.1787	0.1766	0.1821
0.9	0.1243	0.1304	0.1296	0.1344
1	0.0676	0.0547	0.0576	0.0580
Average Precision	0.3868	0.3881	0.3877	0.3911
# documents retrieved				
5	0.5760	0.5840	0.5840	0.5840
10	0.4600	0.4760	0.4800	0.4840
15	0.4053	0.4053	0.4027	0.4053
20	0.3580	0.3460	0.3480	0.360
30	0.2987	0.2893	0.2867	0.2933

Table 4.8: Results for long query geographic thesaurus expansion experiments. The table shows the recall and precision values for the geographic thesaurus expansion runs on the long queries. * indicates a statistically significant difference according to the Wilcoxon test. The first column on the left shows recall levels. The second column shows the precision values for the long query baseline. The third column shows the precision values for expanding queries with administrative districts and regions. The fourth column shows the precision values for expanding queries with administrative districts. The final column shows the precision values for expanding queries with regions.

So, for long queries expanding via geographic thesaurus did not improve the baseline significantly, but it did not hurt retrieval. This shows that more research could be done on selecting terms from the thesaurus that might improve retrieval.

4.4.1.2 Short Queries

Geo-expansion of short queries did not show any improvement over the baseline. Table 4.9 shows the results of expanding the short queries with geographic thesaurus terms. Expanding with both administrative districts and regions and expanding with only administrative districts had statistically significantly lower average precisions than the baseline. Expanding with regions had a lower average precision, but the difference was not statistically significant.

The results of the short queries indicate that the regions and administrative districts are not good words to add to queries. In fact they are bad terms. This is not as obvious for the long queries, but they do not improve the long queries significantly either.

One explanation for why regions are bad terms is that they are general terms that would be more likely to be found in non-relevant documents than in relevant documents. For instance, for the “Shark Attacks off California and Australia” query (See Figure 3.1) the region added was “Western Europe / Americas.” The words in that region would be more likely to be found in a document that was not about shark attacks because shark attacks would only be covered in a small percentage of the articles about this region. Thus adding this region to the query harms it.

	Short Query Baseline	Administrative Districts Regions	Administrative Districts	Regions
Recall	Precision			
0	0.7180	0.5981	0.5808	0.7225
0.1	0.5926	0.4237*	0.3962*	0.5792
0.2	0.4958	0.3346***	0.3234***	0.4662
0.3	0.4450	0.2554***	0.2446***	0.3919*
0.4	0.3693	0.2005***	0.1901***	0.3272
0.5	0.3375	0.1643*	0.1552*	0.2941
0.6	0.2735	0.1400***	0.1380***	0.2448
0.7	0.2281	0.1124***	0.1115***	0.1971
0.8	0.1770	0.0927***	0.0917***	0.1432
0.9	0.1236	0.0808***	0.0810***	0.1065
1	0.0633	0.0302***	0.0305***	0.0486***
Average Precision	0.3337	0.2031***	0.1948***	0.3053
# documents retrieved				
5	0.4480	0.3280	0.3200	0.4480
10	0.4320	0.2760***	0.2800***	0.3960
15	0.3813	0.2400***	0.2373***	0.3653
20	0.3260	0.2220***	0.2200***	0.314
30	0.2653	0.1933***	0.1893***	0.256

Table 4.9: Results for short query geographic thesaurus expansion experiments. The table shows the recall and precision values for the geographic thesaurus expansion runs on the short queries. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column on the left shows recall levels. The second column shows the precision values for the long query baseline. The third column shows the precision values for expanding queries with administrative districts and regions. The fourth column shows the precision values for expanding queries with administrative districts. The final column shows the precision values for expanding queries with regions.

For administrative districts, part of the problem is that every administrative district within a country is added to the queries. This is probably too many terms to add. Most of those administrative districts will most likely not be mentioned in relevant documents. A better way is needed to select administrative districts to add so that only ones that might be relevant to the query are added.

One explanation for why the short queries are affected adversely by adding administrative districts and regions and the long queries are not is that the short queries are not as well specified as the long queries. So, more importance is given to added terms than in the longer queries. This means that bad terms added have a larger effect and are not balanced out as they might be in a longer query.

So, the short queries showed that regions and administrative districts are not good terms to add to short queries and that more research needs to be done on selecting terms from the geographic thesaurus and on limiting the number of terms that are added to the queries. They also showed that short queries show the effects of adding bad terms more clearly than long queries.

4.4.2 Blind Feedback on Geographic Thesaurus Expansion

Expansion with administrative districts and regions alone was not useful, but for the long queries it did not hurt retrieval either. Blind feedback, on the other hand, successfully improved retrieval for both long and short queries. The hope is that by combining the two different methods of expansion, they might improve retrieval more than one of them alone could. Thus the second geospatial experiment did additional expansion using blind feedback on top of the queries already expanded using the geographic thesaurus. It performed blind feedback, assuming the top 5 documents were relevant and adding 5 terms (long queries) or 10 terms (short queries) to the queries that had been expanded by the geographic thesaurus. Below I discuss the results of these runs on the long queries (Section 4.4.2.1) and short queries (Section 4.4.2.2).

4.4.2.1 Long queries

The long queries overall showed an improvement due to using blind feedback on queries previously expanded by the geographic thesaurus. Remember there were three way of expanding queries, with administrative districts and regions, with only administrative districts, and with only regions. The results of doing feedback on top of all three of those are discussed below.

Table 4.10 shows the results of adding the top 5 terms through feedback to the queries with administrative districts and regions. Table 4.10 shows the results of expanding the administrative district and region queries with 5 terms next to the results of the original administrative district and region queries. As can be seen, adding 5 terms improved the average precision when compared to the unexpanded administrative district and region queries. Unfortunately, the difference was not statistically significant. So, feedback does not significantly improve the retrieval of the queries with administrative districts and regions.

Feedback does not significantly improve results over just doing geographic thesaurus expansion with administrative districts and regions. One explanation for this is that the administrative districts and regions were not good terms. The effect of adding them to queries cannot be offset by doing feedback on top of them. Perhaps this is because there are so many administrative districts added that feedback cannot improve the query very much.

	Administrative Districts Regions	5 terms
	Precision	
Recall		
0	0.8183	0.8481
0.1	0.6615	0.6857
0.2	0.6031	0.6367
0.3	0.5425	0.5704
0.4	0.4587	0.4967
0.5	0.3927	0.4234
0.6	0.3371	0.3492
0.7	0.2504	0.2614
0.8	0.1787	0.1868
0.9	0.1304	0.1275
1	0.0547	0.0666**
Average Precision	0.3881	0.4084
	Precision	
# documents retrieved		
5	0.5840	0.6320
10	0.4760	0.4880
15	0.4053	0.4267
20	0.3460	0.3680
30	0.2893	0.2973

Table 4.10: Results from doing feedback on long queries expanded with administrative districts and regions. The table shows results for blind feedback on long query expansion via the geographic thesaurus runs where administrative districts and regions were added to the query. ** indicates a statistically significant difference according to the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts and regions added. The third column shows precision values for feedback on top of queries with administrative districts and regions.

As well as looking at the results for comparing the administrative district and region queries with the feedback expanded administrative district and region queries, it is also important to compare the results to the baseline for the long queries. This comparison is important because the baseline was the starting point;

and even though combing feedback and geographic thesaurus expansion with administrative districts did not show significant improvement, it is possible that it does improve results over the long query baseline. Unfortunately for administrative district and region queries expanded with 5 terms, there was not a statistically significant difference, except when 5 documents were retrieved after expansion with 5. This may be important because users are unlikely to want to look at all the documents retrieved. So, retrieving more relevant documents in the top 5 is good.

Table 4.11 shows the results of the administrative district queries and the results for expanding the administrative districts queries. The feedback expanded administrative district queries also showed improvements due to the addition of new terms through blind feedback, but the improvement in average precision was not statistically significant. It was, however, statistically significant when 5 documents were retrieved.

The fact that doing feedback on the administrative district queries significantly improved retrieval when 5 documents were retrieved is encouraging. This implies that at low recall levels doing feedback on top of the geographic thesaurus expanded queries may be useful. Since most users will not be willing to look through all of the queries, doing well when only a few queries are retrieved is important. The results show that feedback is able to partially cancel out the effect of adding administrative districts. This seems to imply once again that administrative districts are not good terms to add.

	Administrative Districts	5 terms
Recall		
0	0.8183	0.8448
0.1	0.6616	0.6852
0.2	0.6024	0.6352
0.3	0.5418	0.5686
0.4	0.4581	0.4966
0.5	0.3914	0.4236
0.6	0.3354	0.3498
0.7	0.2474	0.2551
0.8	0.1766	0.1862
0.9	0.1296	0.1343**
1	0.0576	0.0665
Average Precision	0.3877	0.4084
# documents retrieved		
5	0.5840	0.6400**
10	0.4800	0.4840
15	0.4027	0.4267
20	0.3480	0.3720
30	0.2867	0.2987

Table 4.11: Results from doing feedback on long queries expanded with administrative districts. The table shows results for blind feedback on long query expansion via the geographic thesaurus runs where administrative districts were added to the query. ** indicates a statistically significant difference according to the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts added. The third column shows precision values for feedback on top of queries with administrative districts.

Like with the administrative district and region queries, the administrative district queries will now be compared to the long query baseline. The feedback on the administrative district expanded queries improves retrieval over the long query baseline, but again none of the improvements in average precision were statistically significant, though at 0.4 recall adding 5 terms showed a statistically significant improvement over the baseline as did retrieving 5 documents.

So, adding administrative districts and then doing feedback with long queries may be useful at low levels of retrieval when compared both to the baseline and to the administrative district queries, but overall feedback does not significantly improve the queries.

The third geographic thesaurus expansion added regions to queries. Table 4.12 shows the results of doing feedback on these queries. Queries with regions added to them improved due to the addition of terms through blind feedback. The improvement in average precision, as seen on Table 4.12, was statistically significant.

The results from this experiment show that region queries do well when expanded using feedback. This seems contradictory to the fact that from section 4.4.1.2 it appears that the regions are bad terms to add. One explanation is that the long queries are well specified enough that adding the regions, which are between 1-3 words, does not hurt them, so that doing feedback significantly improves the region queries.

Comparing the feedback expanded region queries to the baseline, it can be seen that adding 5 terms improved average precision; the improvement was statistically significant. In addition, adding 5 terms also statistically significantly improved precision when 5 documents had been retrieved.

	Regions	5 terms
Recall	Precision	
0	0.8215	0.8440
0.1	0.6804	0.7039
0.2	0.6151	0.6484
0.3	0.5437	0.5906***
0.4	0.4532	0.4910***
0.5	0.3874	0.4223
0.6	0.3260	0.3571
0.7	0.2540	0.2652
0.8	0.1821	0.2005
0.9	0.1344	0.1325
1	0.0580	0.0717*
Average Precision	0.3911	0.4166*
# documents retrieved		
5	0.5840	0.6320
10	0.4840	0.4960
15	0.4053	0.4320
20	0.3600	0.3860
30	0.2933	0.3080

Table 4.12: Results from doing feedback on long queries expanded with regions. The table shows results for blind feedback on long query expansion via the geographic thesaurus runs where regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with regions added. The third column shows precision values for feedback on top of queries with regions.

So, combining geographic thesaurus expansion with regions and blind feedback does improve queries over the baseline. Overall the results for blind feedback on the long queries previously expanded using the geographic thesaurus show some promise. In particular the ones where regions were added to the query and then blind feedback adds 5 terms to the query show a statistically significant improvement over the baseline.

4.4.2.2 Short Queries

	Administrative Districts Regions	10 terms
Recall	Precision	Precision
0	0.5981	0.5894
0.1	0.4237	0.4306
0.2	0.3346	0.3476
0.3	0.2554	0.3325*
0.4	0.2005	0.2844***
0.5	0.1643	0.2601***
0.6	0.1400	0.2292***
0.7	0.1124	0.1608***
0.8	0.0927	0.1329*
0.9	0.0808	0.1055***
1	0.0302	0.0562
Average Precision	0.2031	0.2526***
# documents Retrieved		
5	0.3280	0.3840
10	0.2760	0.2920
15	0.2400	0.2773
20	0.2220	0.2660***
30	0.1933	0.2373*

Table 4.13: Results from doing feedback on short queries expanded with administrative districts and regions. The table shows the results for doing blind feedback on short query expansion via the geographic thesaurus runs where administrative districts and regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts and regions added. The third column shows precision values for feedback on top of queries with administrative districts and regions added.

As with the expansion by geographic thesaurus, the results of doing expansion via thesaurus followed by feedback on the short queries are not as promising as the results for doing expansion via thesaurus followed by feedback on the long queries. As shown in Table 4.13, using feedback to expand the administrative district and region short queries with the top 10 terms improved average precision statistically significantly.

Blind feedback offset the effect of expanding queries with administrative districts and regions. This is important because it shows that the results of the administrative district and regions queries can be improved. Feedback was able to improve the queries with administrative districts and regions. One reason for this is that original query terms were still in the queries, so doing blind feedback was able to add some relevant terms even though more of the top documents were non-relevant.

Unfortunately, when the results of doing feedback on administrative district and region queries are compared to the short query baseline, it can be seen that the best thing that can be said is that feedback expansion with 10 terms is not statistically significantly worse than the baseline. However the average precision was -24% as compared to the baseline. So, feedback improves the queries over the geographic thesaurus expanded queries without blind feedback, but does not improve them over the baseline. This again emphasizes that administrative districts and regions are not good terms to add. Feedback was only able to cancel out part of their bad effect.

Table 4.14 shows that for queries with administrative districts, similarly to queries with administrative districts and regions, feedback with 10 terms statistically significantly improves the results. In addition, the improvement was statistically significant at most recall levels.

	Administrative Districts	10 terms
Recall		
0	0.5808	0.5783
0.1	0.3962	0.4137
0.2	0.3234	0.3338***
0.3	0.2446	0.3162***
0.4	0.1901	0.2865***
0.5	0.1552	0.265***
0.6	0.1380	0.2333***
0.7	0.1115	0.1610***
0.8	0.0917	0.1332***
0.9	0.0810	0.1046***
1	0.0305	0.0577**
Average Precision	0.1948	0.2485***
# documents Retrieved		
5	0.3200	0.3680
10	0.2800	0.2920
15	0.2373	0.2827***
20	0.2200	0.2700***
30	0.1893	0.2373***

Table 4.14: Results from doing feedback on short queries expanded with administrative districts. The table shows results for doing blind feedback on short query expansion via the geographic thesaurus runs where administrative districts were added to the query. ** indicates statistically significant according to the Signtest. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts added. The third column shows precision values for feedback on top of queries with administrative districts added.

So, once again the feedback counterbalances the harm caused by adding terms from the geographic thesaurus. Unfortunately the queries even with the improvement of feedback do not do better than the short query baseline. Again the difference is not statistically significant but adding any number of terms is over - 20% lower than the average precision for the baseline. However, blind feedback showed an improvement over the administrative district queries without blind feedback, as that was -41.63% worse than the baseline.

As can be seen in Table 4.15, region queries were the only geographical thesaurus expanded short queries which were not statistically significantly improved by feedback. However there was a statistically significant improvement when 5 documents were retrieved. This means that once again feedback counterbalanced some of the effects of adding region terms to the queries. Significantly, feedback was enough to improve the queries at low recall level, which could be good for users as they most likely would not want to wade through too many documents to find what they are looking.

Comparing the results for blind feedback on region queries to the baseline is more promising than comparing the other two feedback-expanded, geographic-thesaurus expanded short queries results. Adding 10 terms shows an improvement over the baseline. The improvement is not statistically significant, but adding 10 terms shows a statistically significant improvement over the baseline when 5 documents have been retrieved. This means that region queries were the only short queries expanded with geographic thesaurus terms that feedback could improve enough for them to do even a little better than the baseline. This is likely because regions add only 1-3 terms, so they have a smaller affect on queries than administrative districts. This means that, even though regions are bad terms to add, their effects are easier to cancel out because the effects are not as large as the effects of adding administrative districts.

	Regions	10 terms
Recall		
0	0.7225	0.6782
0.1	0.5792	0.6144
0.2	0.4662	0.5407
0.3	0.3919	0.4818*
0.4	0.3272	0.4064***
0.5	0.2941	0.3522***
0.6	0.2448	0.2893
0.7	0.1971	0.2379
0.8	0.1432	0.1789***
0.9	0.1065	0.1308*
1	0.0486	0.0778***
Average Precision	0.3053	0.3462
# documents Retrieved		
5	0.4480	0.5120***
10	0.3960	0.4520
15	0.3653	0.3893
20	0.3140	0.3340
30	0.2560	0.2840

Table 4.15: Results from doing feedback on short queries expanded with regions Table showing results for blind feedback on short query expansion via the geographic thesaurus runs where regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with regions added. The third column shows precision values for feedback on top of queries with regions added.

The results of the doing feedback on top of geographically thesaurus expanded queries show that feedback can improve all of the queries, but that when compared to the baseline for both the long queries and the short queries, adding only regions shows the most promise. Perhaps adding administrative districts adds too many words, while adding the region adds fewer so is not as likely to have added as many unhelpful words or to have as large an impact on the query.

4.4.2.3 Comparison to Blind Feedback Expansion

It is important to compare the results of feedback done on the geographic thesaurus expanded queries to the results of just doing feedback to see if there is a difference. By comparing the results of blind feedback on the baseline and on the geographic thesaurus expanded queries, it can be seen whether an improvement can be gained by expanding queries with the geographic thesaurus before doing feedback.

Unfortunately, for both long and short queries this analysis confirms that the terms added by the geographic thesaurus do not help retrieval. For instance, figure 4.4 shows a graph with the long query baseline, feedback with 5 terms on that baseline, geographic expansion with regions on the long query baseline and feedback done on queries with regions. The graph shows that for the long queries doing feedback with 5 terms on top of the baseline and doing feedback with 5 terms on top of the query with regions added achieve almost exactly the same results. This shows that using regions does not help retrieval over simply doing feedback. Most likely this is because, as mentioned earlier, the regions were not good terms to add. The administrative districts were even worse terms to add because of how many of them there were, so doing feedback on top of them did not help retrieval when compared to simply doing feedback.

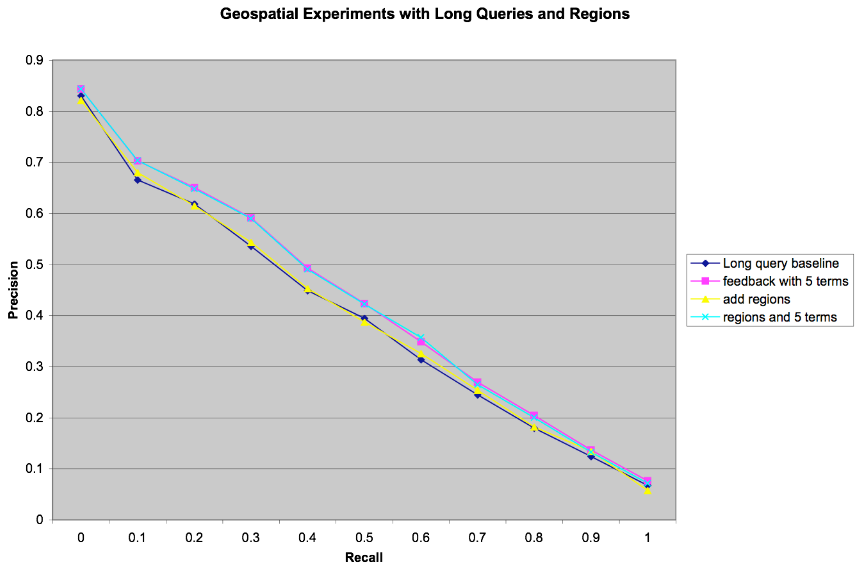

Figure 4.4: Comparison of feedback on long query baseline and on region expanded long queries. Graph comparison of the long query baseline, feedback results assuming 5 documents are relevant and adding 5 terms, adding regions to the baseline and adding 5 terms through feedback to the region expanded queries.

Since doing feedback on region expanded queries had the best results for short queries, Figure 4.5 shows a graph with the short query baseline, the feedback with 10 terms on the baseline, adding regions to the short query baseline and doing feedback with 10 on the short queries with regions. The results shown by the graph are even more disappointing than for long queries. Adding regions to the short queries clearly does worse than the baseline, and even once the feedback is done on those queries retrieval does not go up to the level of just doing feedback.

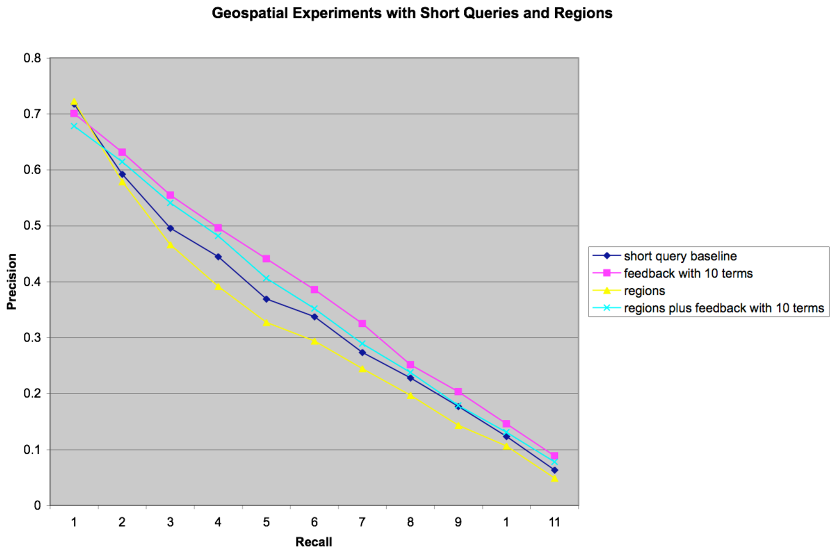

Figure 4.5: Comparison of feedback on short query baseline and on region expanded short queries. Graph comparison of the short query baseline, feedback results assuming 5 documents are relevant and adding 10 terms, adding regions to the baseline and adding 10 terms through feedback to the region expanded queries.

As shown in Figure 4.4 and 4.5, for both short and long queries doing feedback on geographic thesaurus expansion does not improve retrieval over what can be attained by doing feedback on the baseline queries. As mentioned in section 4.3.3.2, doing feedback on top of geographic thesaurus expansion does not significantly improve results over the baseline. In some cases it even hurts results when compared to the baseline. Even the minor improvements, with adding regions and then doing feedback, do not really help much since that improvement is due to the feedback. The feedback alone still does better than feedback after geographic thesaurus expansion.

4.4.3 Geographic Thesaurus Expansion on Blind Feedback Expanded Queries

Since doing feedback on top of geographical thesaurus expanded queries did not show any significant improvements over the baseline, a third experiment that switched the order of blind feedback and geographic thesaurus expansion was done. This experiment was done to see what effect geographic thesaurus expansion had on feedback expanded queries. The third experiment used the geographic thesaurus to expand queries that had been through blind feedback. The queries used were ones created earlier by the blind feedback experiments. Both long and short queries were used, all of which were formed assuming that the top 5 documents were relevant. The top 5 terms were added through feedback for long queries and the top 10 terms were added for short queries. In order to add terms from the geographic thesaurus, all words in the queries were looked for in the thesaurus to see if they were countries. Those that were found to be countries were used to select additional words to be added. Section 4.4.3.1 discusses the results of the experiment on long queries. Section 4.4.3.2 discusses the results for short queries.

4.4.3.1 Long queries

Results for doing geographic thesaurus expansion after blind feedback on long queries were disappointing. Table 4.16 shows that expanding feedback queries with administrative districts and regions hurt retrieval when compared to expanding baseline queries with administrative districts and regions. The difference was statistically significant at almost all levels.

	Administrative Districts Regions	5 terms Administrative Districts Regions
Recall		
0	0.8183	0.6299***
0.1	0.6615	0.4925*
0.2	0.6031	0.4925***
0.3	0.5425	0.4099***
0.4	0.4587	0.3459***
0.5	0.3927	0.2900***
0.6	0.3371	0.2195***
0.7	0.2504	0.1551*
0.8	0.1787	0.1024*
0.9	0.1304	0.0622***
1	0.0547	0.0325*
Average Precision	0.3881	0.2590***
# documents retrieved		
5	0.5840	0.4560**
10	0.4760	0.3650***
15	0.4053	0.3173
20	0.3460	0.2860
30	0.2893	0.2400

Table 4.16: Results from expanding feedback on long queries with administrative districts and regions. The table shows the results for doing blind feedback and expansion based on a geographic thesaurus on long query runs where administrative districts and regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. ** indicates a statistically significant difference according to the Signtest. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts and regions added. The third column shows precision values for adding administrative districts and regions to feedback expanded queries.

So, clearly doing thesaurus expansion with administrative districts and regions on top of blind feedback is not helpful. This shows once again that administrative districts and regions are poor words to add. In addition, some terms in the queries were misidentified as countries and thus expanded. So, the process for recognizing countries did not work completely.

When compared to the long query baseline, the administrative district and region expanded feedback queries did statistically significantly worse showing once again that without more selective choosing of geographic thesaurus terms expansion with administrative districts and regions hurts retrieval.

Table 4.17 shows that adding administrative districts to feedback expanded queries hurts retrieval when compared to expanding baseline queries with administrative districts. The difference was statistically significant at all levels except recall level 0.1 and when 15, 20 or 100 documents were retrieved.

Again the results for expanding feedback queries with administrative districts confirm that administrative districts and regions are ineffective terms to add. Perhaps if fewer administrative districts were selected to add and if words were not misidentified as countries the results might improve.

Comparing the results of doing geographic thesaurus expansion with administrative districts on feedback expanded queries to the long query baseline shows that doing expanding feedback queries with administrative districts does statistically significantly worse than the baseline at all recall levels.

	Administrative Districts	5 terms Administrative Districts
Recall		
0	0.8183	0.6259*
0.1	0.6616	0.4899
0.2	0.6024	0.4062***
0.3	0.5418	0.3564***
0.4	0.4581	0.2864***
0.5	0.3914	0.2564***
0.6	0.3354	0.2203*
0.7	0.2474	0.1557*
0.8	0.1766	0.1021*
0.9	0.1296	0.0622***
1	0.0576	0.0338*
Average Precision	0.3877	0.2584***
# documents retrieved		
5	0.5840	0.4480***
10	0.4800	0.3600**
15	0.4027	0.3173
20	0.3480	0.2920
30	0.2867	0.2387*
100	0.1752	0.1396***
200	0.1140	0.0940***
500	0.0569	0.0511***
1000	0.0316	0.0286

Table 4.17: Results from expanding feedback on long queries with administrative districts. The table shows the results for doing blind feedback and expansion based on a geographic thesaurus on long query runs where administrative districts were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. ** indicates a statistically significant difference according to the Signtest. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts added. The third column shows precision values for adding administrative districts to feedback expanded queries.

Table 4.18 shows that adding regions to the feedback expanded queries hurt retrieval when compared to adding regions to the long query baseline. Again the difference was statistically significant for average precision. When compared to the long query baseline, feedback queries expanded with regions did statistically significantly worse at all recall levels.

	Regions	Regions 5 terms
Recall		
0	0.8215	0.7784
0.1	0.6804	0.6168
0.2	0.6151	0.5429***
0.3	0.5363	0.4688***
0.4	0.4532	0.3835***
0.5	0.3874	0.3359***
0.6	0.3260	0.2825
0.7	0.2540	0.1938***
0.8	0.1821	0.1456
0.9	0.1344	0.0893**
1	0.0580	0.0451
Average Precision	0.3911	0.3373***
# documents retrieved		
5	0.5840	0.5840
10	0.4840	0.4320**
15	0.4053	0.3760
20	0.3600	0.3320
30	0.2933	0.2707
100	0.1716	0.1500***
200	0.1148	0.0982***
500	0.0582	0.0523***
1000	0.0318	0.0290

Table 4.18: Results from expanding feedback on long queries with regions
The table shows the results for blind feedback and expansion based on a geographic thesaurus on long query runs where regions were added to the query. ** indicates a statistically significant difference according to the Signtest. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with regions added. The third column shows precision values for adding regions to feedback expanded queries.

Once again the results show that regions are not good terms to add. They are too general and thus are present in many non-relevant documents as well as in relevant documents. So, geospatial relationships other than regions need to be explored for expanding countries. Perhaps adding the capital city of a country would be a better choice than regions.

Why does expanding with geographic thesaurus after blind feedback hurt retrieval so much? One explanation is that, as mentioned earlier in section 4.4.1.2, administrative districts and regions are not good terms to add. So, when there are more words that could be countries and thus be expanded on by adding new terms from the geographic thesaurus, this makes the unhelpfulness and, in fact, harmfulness of the administrative district and region terms even more apparent than earlier. Another explanation is that some terms in the feedback queries were misidentified as countries and thus expanded with administrative districts and regions from the geographic thesaurus. This means that the algorithm used to identify locations in queries needs to be modified in order to correctly identify countries and other locations.

These results from doing geographic thesaurus expansion on feedback queries show geographic expansion on long feedback queries hurts retrieval. In order for it to possibly help retrieval, words from the geographic thesaurus would need to be added more selectively and the algorithm used to identify locations would need to be modified to identify locations better.

4.4.3.2 Short queries

The results using the short queries were similar to the ones from using long queries. Table 4.19 shows that expanding feedback queries with administrative districts and regions improved retrieval slightly over adding administrative districts and regions to the baseline. However, the difference was not statistically significant.

	Administrative Districts Regions	10 terms Administrative Districts Regions
Recall		
0	0.5981	0.6076
0.1	0.4237	0.4955
0.2	0.3346	0.4066***
0.3	0.2554	0.3569***
0.4	0.2005	0.2808***
0.5	0.1643	0.2437***
0.6	0.1400	0.2000***
0.7	0.1124	0.1288*
0.8	0.0927	0.0922
0.9	0.0808	0.0712
1	0.0302	0.0325
Average Precision	0.2031	0.2492
# documents retrieved		
5	0.3280	0.4320
10	0.2760	0.3440*
15	0.2400	0.2907
20	0.2220	0.2540
30	0.1933	0.2240

Table 4.19: Results from expanding feedback on short queries with administrative districts and regions. The table shows the results for doing blind feedback and expansion based on a geographic thesaurus on short query runs where administrative districts and regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts and regions added. The third column shows precision values for adding administrative districts and regions to feedback expanded queries.

The results for adding administrative districts to feedback queries were much the same as for adding administrative districts and regions, as shown in Table 4.20.

	Administrative Districts	10 terms Administrative Districts
Recall		
0	0.5808	0.5806
0.1	0.3962	0.4885
0.2	0.3234	0.3989
0.3	0.2446	0.3573
0.4	0.1901	0.2826
0.5	0.1552	0.2394*
0.6	0.1380	0.1906
0.7	0.1115	0.1269***
0.8	0.0917	0.0951***
0.9	0.0810	0.0697
1	0.0305	0.0341
Average Precision	0.1948	0.2465
# documents retrieved		
5	0.3200	0.4240
10	0.2800	0.3400
15	0.2373	0.2853
20	0.2200	0.2520
30	0.1893	0.2253

Table 4.20: Results from expanding feedback on short queries with administrative districts. The table shows results for doing blind feedback and expansion based on a geographic thesaurus on short query runs where administrative districts were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with administrative districts added. The third column shows precision values for adding administrative districts to feedback expanded queries.

Table 4.21 shows that expanding feedback queries with regions did slightly better than expanding baseline queries with regions. The difference was not significant for average precision. It was when five documents were retrieved. Showing that at a low recall level the regions do help queries.

	Regions	10 terms Regions
Recall		
0	0.7225	0.6789
0.1	0.5792	0.5980
0.2	0.4662	0.5049
0.3	0.3919	0.4325
0.4	0.3272	0.3644
0.5	0.2941	0.3176
0.6	0.2448	0.2622
0.7	0.1971	0.1913
0.8	0.1432	0.1314
0.9	0.1065	0.1011
1	0.0486	0.0444
Average Precision	0.3053	0.3133
# documents retrieved		
5	0.4480	0.5200*
10	0.3960	0.4160
15	0.3653	0.3520
20	0.3140	0.3020***
30	0.2560	0.2600

Table 4.21: Results from expanding feedback on short queries with regions
The table shows results for doing blind feedback and expansion based on a geographic thesaurus on short query runs where regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the queries with regions added. The third column shows precision values for adding regions to feedback expanded queries.

The results for the three runs expanding feedback queries using terms from the geographic thesaurus show that regions are still the best thing to add to the queries. This is likely because they include fewer words than the administrative districts and so do not add as many weak terms, that can harm retrieval.

Compared to the baseline, adding administrative districts and regions to feedback queries hurt retrieval; the difference was statistically significant. Compared to the short baseline, adding administrative districts to feedback expanded queries hurt retrieval. The difference was statistically significant. Adding regions to feedback queries also statistically significantly hurt retrieval.

As with long queries, the short queries demonstrate the fact that administrative districts and regions are poor words to add to the queries as discussed in section 4.4.1.2. Again adding bad terms hurts retrieval and the short queries because they have fewer terms and have more room for improvement but also more room for the queries to be harmed by the new terms. Additionally, the short queries did not have as large a problem with misidentified countries as long queries did because they had fewer terms that could be misidentified.

4.4.4. Disambiguating “near” using Longitude and Latitude

The fourth geospatial experiment looked at using longitude and latitude to determine the distance between locations. The goal was to use distance to define or disambiguate what is meant by “near” and to select only those terms meeting that definition in the hope that the distance could then be used to determine which places should be considered near and added to the query. Unfortunately, I only had longitude and latitude data for cities. Most of the queries do not mention cities by name. If feedback was used on the queries, a few cities got added. So, for these experiments, queries from blind feedback were used to expand.

Recall	Long Query Baseline	Feedback with 5 terms	Add cities within 5 miles
0.0	0.8307	0.8438	0.1478***
0.1	0.6658	0.7030	0.0674***
0.2	0.6185	0.6505	0.0640***
0.3	0.5363	0.5917*	0.0522***
0.4	0.4491	0.4930*	0.0327***
0.5	0.3948	0.4235	0.0287***
0.6	0.3137	0.3489	0.0179***
0.7	0.2453	0.2700	0.0101***
0.8	0.1799	0.2049	0.0075***
0.9	0.1243	0.1371*	0.0061***
1.0	0.0676	0.0767*	0.0056***
Average Precision:	0.3868	0.4175*	0.0312***
	Precision		
Top x documents			
5 docs	0.5760	0.6240	0.0560***
10 docs	0.4600	0.4920	0.0480***
15 docs	0.4053	0.4347	0.0400***
20 docs	0.3580	0.3840	0.0360***
30 docs	0.2987	0.3147	0.0307***
100 docs	0.1716	0.1804*	0.0184***
200 docs	0.1134	0.1158	0.0164***
500 docs	0.0558	0.0572	0.0132***
1000 docs	0.0310	0.0317	0.0097***

Table 4.22: Results from disambiguating “near” on long queries. The table showing results for blind feedback and longitude latitude based expansion on long query runs where regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the long query baseline. The third column shows precision values for feedback with 5 terms for long queries. The fourth column shows precision values for adding cities within 5 miles.

The distance formula used was distance = $(69.1 * (\text{latitude2} – \text{latitude1}))^{2} + (53.0 * (\text{longitude2} – \text{longitude1}))^{2}$. This gave the approximate distance in miles. So there is room for error in the calculation. An approximate distance seemed reasonable to me because I was trying to tell what cities were within a certain distance of another city. This meant that it was not as important to have the exact distance as it might be in other cases where the distance between places was needed. However, there are more precise distance formulas that could be used if it was necessary to have a precise distance.

Recall	Short Query Baseline	Feedback with 10 terms	Add cities within 5 miles
0.0	0.7180	0.7006	0.3847***
0.1	0.5926	0.6317	0.2328***
0.2	0.4958	0.5549	0.1666***
0.3	0.4450	0.4965	0.1344***
0.4	0.3693	0.4413*	0.1186***
0.5	0.3375	0.3857*	0.1059***
0.6	0.2735	0.3254	0.0705***
0.7	0.2281	0.2514	0.0565***
0.8	0.1770	0.2034*	0.0365***
0.9	0.1236	0.1461*	0.0319***
1.0	0.0633	0.0884*	0.0283***
Average Precision:	0.3337	0.3675*	0.1090***
Top x documents	Precision		
5 docs	0.4480	0.4960	0.1680***
10 docs	0.4320	0.4320	0.1440***
15 docs	0.3813	0.3920	0.1200***
20 docs	0.3260	0.3520	0.1040***
30 docs	0.2653	0.2973*	0.0813***
100 docs	0.1564	0.1700	0.0404***
200 docs	0.1094	0.1148*	0.0318***
500 docs	0.0561	0.0573*	0.0178***
1000 docs	0.0308	0.0316	0.0133***

Table 4.23: Results from disambiguating “near” on short queries. The table shows the results for doing blind feedback and longitude latitude based expansion on short query runs where regions were added to the query. * indicates a statistically significant difference according to the Wilcoxon test. *** indicates a statistically significant difference according to both the Wilcoxon test and the Signtest. The first column shows recall levels. The second column shows precision values for the short query baseline. The third column shows precision values for feedback with 10 terms for short queries. The fourth column shows precision values for adding cities within 5 miles.

For both short and long queries, the expansion using longitude and latitude to determine distance was disastrous. Table 4.22 shows the results for long queries and Table 4.23 shows the results for short queries. In both cases, adding all cities within 5 miles of the original city hurt retrieval. This difference was statistically significant at all recall points according to the Wilcoxon test and the Signtest.

Below I discuss briefly why the longitude and latitude experiment did so poorly (in addition, see Chapter 5 for a discussion of future work that could be done with longitude and latitude). One explanation for the poor results is that city names were not correctly identified. My algorithm for recognizing location names did not work. It incorrectly identified cities. It did better with countries as seen from the experiments discussed in sections 4.3.1, 4.3.2, but in experiments discussed in section 4.3.3 the system had trouble identifying countries when there were lots of terms being looked at and it could not identify cities in the queries. For instance, looking at the "Shark Attacks off California and Australia" query (See Figure 3.3) a human reader would notice that there are no cities among the query words and yet the short query when expanded had added 2075 words. This is likely because many of the terms in the query can be found in my geographic thesaurus to be the name of a city, even if that is not the most common usage of the term. This caused the program to add many terms (for the short queries between 27- 4353 terms per a query, with an average of 1023 per a query) to the query even when most of the words a human reader would recognize as not being helpful. So, one problem is that the names of cities cannot be identified correctly. One way to improve this would be to modify my algorithm for recognizing location. Currently, the algorithm matches terms in the query to terms in the geographic thesaurus. One way this might be improved would be by using a part of speech tagger to tag the queries and then only check if words tagged as proper nouns are locations.

A second problem is that when a city name was identified, either correctly or incorrectly, there were found to be a lot of cities within 5 miles of that city. This means that even if the program had only identified cities correctly, it would still likely be adding too many terms. So, a way needs to be found to restrict how many cities are actually added from those that are identified as being within a certain distance.

Originally I had planned to also experiment with other distances for “near” such as 15, 25, 30, 100 miles etc. Due to the terrible results for 5 miles, I decided not to enlarge the distance because that would have caused even more terms to be added making the results more disastrous. In addition, the experiments done with longitude and latitude are preliminary. I had longitude and latitude data for cities, but more longitude/latitude data is needed as well as a better way to locate cities.

4.5 Summary of Results

To summarize the results of the experiments described in this chapter, blind feedback significantly improved retrieval over the baseline when 5 documents were used and 5 terms added for long queries and 10 terms added for short queries. For re-weighting it was found that Rocchio is the best scheme to use. The geographic thesaurus expansions did not improve retrieval overall. There were cases, as in section 4.4.1.1, where they did not hurt retrieval, but in most cases they failed to improve retrieval over the baseline. This is likely because administrative districts added too many terms to the queries and regions were too general to help retrieval. When using longitude and latitude to define “near,” it was found that the algorithm used to identify locations misidentified many terms in the queries as being locations.

The next chapter presents a summary of the experimental results, the general trends shown by them, conclusions to be drawn from the experimental results, and suggestions for future work in GIR.

CHAPTER 5: CONCLUSIONS AND FUTURE WORK

Information Retrieval (IR) aims to find and retrieve information that fulfills a user's need. In recent years, the amount of information available on the internet has grown at a rapid rate. Still with all of this information available there is a need for better methods to search and find the information one is actually interested in at any one point in time. This thesis examined the area of Geographical Information Retrieval (GIR). The goal of GIR is to exploit geographical information in queries and documents in order to improve retrieval effectiveness. My goal was to use query expansion, re-weighting and geographical information available on the internet to automatically disambiguate query terms in order to improve retrieval effectiveness.

This chapter describes the conclusions that can be drawn from the experiments run and some ideas for future work. Section 5.1, summarizes the experimental results and outlines general trends the results show. Section 5.3 offers conclusions based on the work done. Lastly, in section 5.2, suggestions for future work in GIR are offered.

5.1 Summary of Results

My experiments show that GIR is not as a simple as adding geospatial terms. It is more complex and needs to take into account many different factors such as length of queries, how much geospatial data are available and how geographic entities are identified and used.

The blind feedback experiments showed that using long queries formed from all fields of the query was most effective. For the long queries, the best results were when 5 documents were assumed to be relevant and 5 terms were added (Section 4.2.2). For short queries formed from only the title field, on the other hand, assuming 5 documents were relevant and adding 10 terms gave the best results (Section 4.2.2). For the re-weighting experiments, as I expected, the queries re-weighted and expanded using Rocchio achieved the best results.

The geospatial experiments I did showed that one has to be careful with what geospatial information is used because some kinds of terms (e.g., regions) might hurt retrieval more than they help (Section 4.4). More research needs to be done in order to discover which types of geospatial relationships are useful and how these relationships should be used.

The results of my blind feedback, re-weighting and geospatial experiments illustrate several general trends: the benefits and costs of long and short queries, what impacts feedback effectiveness, the need for more research on thesaurus expansion, and the difficulty of identifying geographic entities correctly.

Regarding queries, the longer the query is to start with the more guidance and context it gives to the system, yielding better results. In addition, the effect of adding poor terms to long queries is less damaging because long queries are better specified. So there is less room for degrading the query. For example, in my experiments expanding long queries with geospatial information from the geographic thesaurus slightly improved the queries, while expanding the short queries with the geographic thesaurus hurt retrieval (Section 4.4.1). Unfortunately, since the long queries are already well specified, it is harder to find "good" terms to add and improve the queries than it is for short queries. Short queries, on the other hand, are less well specified and so are less effective for retrieval. However, since they are less precisely specified, they have more room for improvement through expansion. However, the choice of expansion terms is critical since short queries are more sensitive to the negative impact of poor expansion terms (Section 4.4.1.2). The expanded queries did worse than the baseline. Even when feedback was done on top of the geographic thesaurus expansion only adding regions did slightly better than the baseline (Sections 4.4.3.2, 4.4.3.3). Additionally, shorter queries are more realistic. A user is more likely to type 2 or 3 words than several sentences. So, shorter queries should be a priority because they are the majority of what users use.

In terms of what impacts feedback effectiveness, the first thing, as mentioned above, is the query length. Longer more clearly specified queries are affected less by feedback than shorter queries. In my blind feedback experiments, the short queries did statistically significantly better when 10 terms were added as opposed to the long queries, which did statistically significantly better when 5 terms were added (Section 4.2). One explanation for this is that since the short queries were less effective, there was more room for improvement.

Furthermore, the number of relevant documents used and the number of terms added impacted retrieval effectiveness. The fewer documents used, the more likely those documents are actually relevant and the fewer terms added the more likely those terms will be useful for retrieval. This was seen first in my blind feedback experiments where the best results were obtained using fewer relevant documents and adding fewer terms (Section 4.2). This is related to the actual number of relevant documents per query. If a query has only 3 relevant documents in the entire collection, feedback will not be able to help it as much. While if a query has 100 relevant documents, feedback may be more useful since it is more likely to be looking at relevant documents. As described in section 3.1, most of my queries did not have many relevant documents. Fourteen out of the 25 queries had fewer than 30 relevant documents. Due to relatively few relevant documents available for each query, it was more difficult for blind feedback to improve the queries, because some of the documents that feedback looked at for additional terms would have been non-relevant.

A third trend shown by my experiments is that administrative districts and regions are not good terms to add to queries (Sections 4.4.1, 4.4.3, 4.4.4). Regions were too general and were in more non-relevant documents than relevant ones. For example, for the query “Shark Attacks Off California and Australia” the geographic thesaurus added the region “Western Europe / Americas.” “Western Europe/ Americas” is a larger area, one in which California and Australia are a small part. There are likely many documents that mention Western Europe or

America but are not related to California or Australia, much less to shark attacks. So, regions are poor terms to add to the queries.

Adding administrative districts to queries performed even more poorly than adding regions (Sections 4.4.1, 4.4.2, 4.4.4). This is probably because there were many more administrative districts added to queries than there were regions. Regions tended to be short 1-3 words, while administrative districts were a long list of every administrative district in the country. So, adding administrative districts added too many terms. Most of the administrative terms, again, were not useful. For instance, in the "Shark Attacks Off California and Australia" example, adding the administrative districts of Australia was not particularly helpful. Some of those districts probably have no shark attack within them and so would not be mentioned in relevant documents. Still, some administrative districts might be useful. So, more research could possibly narrow down how administrative districts are selected to be added to the query. This would show whether adding administrative districts in general produces poor results or if just adding too many of them is bad. (See Section 5.2 for more on possible future work.)

A fourth trend is that it is difficult to identify geographic entities. In my experiments, countries were identified fairly well using my geographic thesaurus (Section 4.4.1), but one can imagine problems that might occur in identifying countries. For instance, if the name of a country has changed over time or a country has multiple names, this may not be reflected in the database. Cities pose an even larger problem. Cities were incorrectly identified by my query processor

(Section 4.4.5). It misidentified many words in the queries as cities, even when there were no city names in the query; this hurt retrieval. There are so many cities in the world and many have the same name or names that happen to be more commonly used as non-location words. This means a word can be identified as a city even when in the context of the query it is not meant as a city. So, having a large amount of geospatial information in one's thesaurus means one will be able to recognize more geospatial entities, but some of these entities will be meant as words not geospatial entities. Even if one does not misidentify words and geospatial entities, one might still miss other geospatial terms if they are not in the thesaurus.

5.2 Conclusion

My hypothesis was that using geospatial information for query expansion and re-weighting of terms would improve retrieval effectiveness because the expansion would more clearly specify the query and address issues of language ambiguity and vocabulary mismatch. The results show that blind feedback without geospatial information improves retrieval effectiveness. Adding administrative districts, regions or cities within a certain distance of another city can improve specific queries, but overall does not improve retrieval and in many cases can hurt retrieval. The results did not prove the hypothesis, but they suggest that more work must be done. The experiments started with the simplest approach to adding geospatial information. This a good place to start because now the results presented in this thesis have shown that more sophisticated analysis is necessary. This means that research can move to developing more complex methods for GIR, knowing that the simplest does not work. My research suggests several ways that geospatial information could be used and highlights the complexity of GIR, pointing out future avenues to explore in the use of geospatial information in IR.

5.3 Future Work

One area that clearly needs improvement as indicated by my work is identifying cities or other geospatial entities correctly. My method consisted of matching the query term to a term within my geospatial thesaurus. This worked adequately for recognizing country names, but did not work for recognizing city names (Section 4.4.5) because the method confused words that were not meant as cities with cities. Thus my system would be improved if I added another way, other than simply matching, for checking if terms were geospatial entities. Perhaps the context of queries could be taken into account as well. So, if there was a country mentioned in the query then only cities in that country would be searched for matches to query terms.

A second problem is finding good terms to add. The words I chose to add to queries did not improve retrieval overall (Sections 4.4.1, 4.4.3, 4.4.4). I added regions and administrative districts. Regions were too general and thus added terms that were in more non-relevant documents than in relevant documents. Administrative districts, on the other hand, added too many terms. So more research needs to be done on what other kinds of terms could be added. For instance, perhaps if a query uses the word or mentions something that is associated with the ocean as in the query “Shark Attacks Off California and Australia,” where shark clearly refers to somewhere with an ocean, then only administrative districts near an ocean would be added. So, in this example, administrative districts on the coast of California or Australia could be added, instead of adding all administrative districts of California and Australia.

More research is needed to find a way to decrease the number of terms added when expanding based on distance. As noticed in my experiments, adding all of the cities within 5 miles of a location mentioned in the query added too many terms and was detrimental to retrieval (Section 4.4.5). Some way of limiting terms is needed. The simplest way to do this would be to add only the n closest places, but that might miss some of the most useful terms that could be added. Another method could be considering additional information along with the distance information when choosing terms to add. For example, if a system had population information, the system could add only terms within a certain distance that had above a certain population or the top n populous cities within m miles.

So, one way that GIR could be improved would be to discover what sorts of terms are good to add and what kind of geospatial relationships are useful. Also what can theses terms and relationships be used for? Are some terms or relationships good to add or are they more useful for finding other terms to add even if they are not necessarily good terms to add? Additionally, how many terms should be added to queries for GIR? Another avenue for improving GIR would be to explore exactly what types of queries should be considered GIR. This would help to get a clearer idea of what makes GIR different than regular IR. This knowledge could then be used to gain an idea of what methods might improve GIR and what types of terms and relationships might be useful to GIR.

BIBLIOGRAPHY

[1] Kerstin Bischoff, Thomas Mandl, and Christa Womser-Hacker. Blind Relevance Feedback and Named Entity based Query Expansion for Geographical Retrieval at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[2] Davide Buscaldi, Paolo Rosso, and Emilio Sanchia Arnal. A WordNet-based Query Expansion Method for Geographical Information Retrieval. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[3] David Buscaldi, Paolo Rosso, and Emilio Sanchis. WordNet-based Index Terms Expansion for Geographical Information Retreival. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[4] J. Callen, W. B. Croft, and S. M. Harding. The INQUERY retrieval system. In A. Min Tjoa and I. Tamos (Eds.), *Proceedings of the 3rd International Conference on Database and Expert Systems Applictions,* pp. 78-83. Berlin: Springer-Verlag, 1992.

[5] Nuno Cardoso, Bruno Martins, Marcirio Silveira Chaves, Leonardo Andrade, and Mario J. Silva. The XLDB Group at GeoCLEF 2005. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[6] O. Ferrandez, Z. Kozareve, A. Toral, E. Noguera, A. Montoyo, R. Munoz, and Fernando Llopis. University of Alicante at GeoCLEF 2005. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[7] Daniel Ferres, Alicia Ageno, and Horacio Rodriguez. The GeoTALP-IR System at GeoCLEF-2005: Experiments Using a QA-based IR System, Linguistic Analysis, and a Geographical Thesaurus. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[8] David Ferres and Horacio Rodriguez. TALP at GeoCLEF-2006: Experiments Using JIRS and Lucene with the ADL Feature Type Thesaurus. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[9] Manuel Garcia-Vega, Miguel A. Garcia-Cumbreras, L. Alfonso Urena-Lopez, and Jose M. Oerea-Ortega. SINAI at GeoCLEF 2006: Expanding the Topics with Geographical Information and Thesaurus. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[10] Manuel Garcia-Vega, Miguel A. Garcia-Cumbreras, L. Alfonso Urena-Lopez, Jose M. Oerea-Ortega, and F. Javier Ariza-Lopez. R2D2 at GeoCLEF 2006: a Mixed Approach. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[11] Andogah Geoffrey. GIR Experimentation. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[12] Fredric Gey, Ray Larson, Mark Sanderson, Kerstin Bischoff, Thomas Mandl, Christa Womser-Hacker, Diana Santos and Paulo Rocha. GeoCLEF 2006: the CLEF 2006 Cross-Language Geographic Information Retrieval Track Overview. In CLEF2006 Notebook paper (CDROM) Springer, 2006.

[13] Fredric Gey, Ray Larson, Mark Sanderson, Hideo Joho, Paul Clough, and Vivian Petras. GeoCLEF: the CLEF 2005 Cross-Language Geographic Information Retrieval Track Overview. In *Cross-Language Evaluation Forum: CLEF 2005.* Springer (Lecture Notes in Computer Science LNCS 4022), 2006.

[14] Fredric Gey and Vivien Petras. Berkeley2 at GeoCLEF: Cross-Language Geographic Information Retrieval of German and English Documents. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[15] Rocio Guillen. CSUSM Experiments in GeoCLEF2005: Monolingual and Bilingual Tasks. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[16] Rocio Guillen. Monolingual and Bilingual Experiments in GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[17] Claudia Hauff, Dolf Trieschnigg, and Henning Rode. University of Twente at GeoCLEF 2006: geofiltered document retrieval. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[18] You-Heng Hu and Linlin Ge. UNSW at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[19] Baden Hughes. NICTA i2d2 at GeoCLEF 2005. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[20] Thorsten Joachims. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of {ICML}-97, 14th International Conference on Machine Learning, p. 143-151, 1997.*

[21] Andras Kornai. MetaCarta at GeoCLEF 2005. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[22] Sara Lana-Serrano, Jose M. Goni-Menoyo, and Jose C. Gonzalez-Cristobal. Miracle's 2005 Approach to Geographical Information Retrieval. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[23] Sara Lana-Serrano, Jose M. Goni-Menoyo, and Jose C. Gonzalez-Cristobal. Report of MIRACLE team for Geographical IR in CLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[24] Ray R. Larson. Chesire II at GeoCLEF: Fusion and Query Expansion for GIR. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[25] Ray R. Larson and Fredric C. Gey. GeoCLEF Text Retrieval and Manual Expansion Approaches. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[26] Jochen L. Leidner. Preliminary Experiments with Geo-Filtering Predicates for Geographic IR. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[27] Johannes Leveling, Sven Hartrumpf, and Dirk Veiel. University of Hagen at GeoCLEF 2005: Using Semantic Networks for Interpreting Geographical Queries In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2005*, 2005.

[28] Johannes Leveling and Dirk Veiel. University of Hagen at GeoCLEF 2006: Experiments with Metonymy Recognition in Documents. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[29] Yi Li, Nicola Stokes, Lawrence Cavedon, and Alistair Moffat. NICTA I2D2 Group at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[30] Zhisheng Li, Chong Wang, Xing Xie, and Wei-Ying Ma. MSRA Columbus at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[31] Bruno Martins, Nuno Cardoso, Marcirio Silveira Chaves, Leonardo Andrade and Mario J. Silva. The University of Lisbon at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[32] Simon Overall, Joao Magalhaes, and Stefan Ruger. Place Disambiguation with Co-occurrence Models. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[33] J. J. Rocchio, Jr. “Relevance Feedback in Information Retrieval” in Salton, Gerard (ed.) The SMART Retrieval System, experiments in automatic document processing. Englewood Cliffs, NJ: Prentice –Hall, Inc, 1971, pp. 313-323.

[34] Miguel E. Ruiz, Stuart Shapiro, June Abbas, Silvia B. Southwick and David Mark. UB at GeoCLEF 2006. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[35] Karen Sparck Jones and Peter Willett. *Readings in Information Retrieval.* San Francisco, CA: Morgan Kaufmann Publishing, Inc., 1997.

[36] A, Toral, O. Ferrandez, E. Noguera, Z. Kozareva, A. Montoyo and R. Munoz. Geographic IR Helped by Structured Geospatial Knowledge Resources. In *Working Notes in Cross-Language Evaluation Forum (CLEF) 2006*, 2006.

[37] Jinxi Xu and W. Bruce Croft. 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.

[38] Geographic Names Database.
http://gnswww.nga.mil/geonames/GNS/index.jsp