International Journal of Geographical Information Science
Vol. 00, No. 00, Month 200x, 1–14

An empirical study of the effects of NLP components on Geographic IR performance

Nicola Stokes*, Yi Li, Alistair Moffat, Jiawen Rong

NICTA Victoria Laboratory
Department of Computer Science and Software Engineering,
The University of Melbourne, Victoria 3010, Australia

(*v3 July, 2007*)

Natural Language Processing (NLP) techniques, such as toponym detection and resolution, are an integral part of most Geographic Information Retrieval (GIR) architectures. Without these components, synonym detection, ambiguity resolution and accurate toponym expansion would not be possible. However, there are many important factors affecting the success of an NLP approach to GIR, including toponym detection errors, toponym resolution errors, and query overloading. The aim of this paper is to determine how severe these errors are in state-of-the-art systems, and to what extent they affect GIR performance. We show that a careful choice of weighting schemes in the IR engine can minimize the negative impact of these errors on GIR accuracy. We provide empirical evidence from the GeoCLEF 2005 and 2006 datasets to support our observations.

Keywords: Geographic Information Retrieval, Toponym Detection, Toponym Resolution, Wikipedia.

1 Introduction

Natural Language Processing (NLP) techniques, such as toponym detection and resolution, are an integral part of most Geographic Information Retrieval (GIR) architectures. The recognition and grounding of place names provides important information that is potentially unavailable to a search system based purely on text. Without these NLP components, synonym detection, ambiguity resolution and accurate toponym expansion would not be possible. However, there has been a failure to report the accuracy of these components when they have been used as preprocessing steps within GIR architectures employed in evaluation exercises such as GeoCLEF (Gey *et al.* 2006).

The aim of this paper is to determine how severe these errors are, and to what extent they affect GIR performance. More specifically, we compare the accuracy of two commonly used toponym detection systems and one baseline system on the GeoCLEF data; we evaluate our toponym resolution system on the same subset of hand annotated data; we report the knock-on effects of errors from these NLP components on GIR performance. To facilitate this analysis the paper also introduces our system, which was the only NLP-based GIR system that showed statistical significant improvement over its baseline equivalent on the GeoCLEF 2006 dataset.

This system uses a novel query expansion framework that minimize the effect of these NLP preprocessing errors. Query expansion refers here to the process of automatically adding additional terms to the query (in this instance place names), in an effort to improve the relevance of the retrieved results. Our expansion technique also addresses the issue of query overloading, where non-location query terms are swamped by geographic ones after geospatial expansion. In addition, this paper describes a novel toponym resolution algorithm that leverages geospatial information from Wikipedia to improve performance.

Our results show that off-the-shelf toponym identification systems are under-performing on the GeoCLEF data (F-score < 0.6); that even with state-of-the-art toponym resolution performance (accuracy $> 80\%$) the accumulative effect of the NLP errors on the final annotated corpus means that for our best GIR run, 30% of locations were missed in the indexed collection, and 51% were falsely annotated. In spite of these

*Corresponding author: Nicola Stokes Email: nstokes@csse.unimelb.edu.au

International Journal of Geographical Information Science
ISSN 1365-8816 print / ISSN 1362-3087 online © 200x Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/1365881YYxxxxxxxx

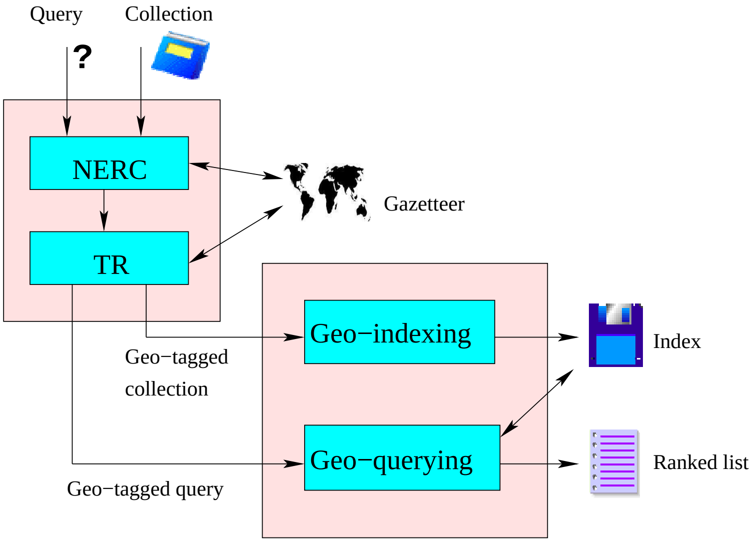

Figure 1. Components of a GIR system.

errors, we achieve a 13% increase over our baseline performance on the 2006 GeoCLEF collection. This suggests that further gains in GIR performance are possible if the accuracy of these NLP components can be improved.

2 GIR architectures and related work

Figure 1 shows the architecture of a typical approach to a GIR system. There are four steps involved in the process: *named entity recognition and classification* (NERC); *toponym resolution* (TR); *geographic indexing*; and *retrieval* (Li *et al.* 2006a). Compared to a traditional IR system, NERC and TR are employed to identify the geographical locations in the collection and queries, so that knowledge of them can be factored into the ranking of the retrieved documents.

In the context of GIR, NERC is concerned with identifying toponyms, by first recognizing name entities and then differentiating between references to the names of places (which we are interested in), and the names of people and organizations (which we are not). Toponym Resolution, also known as *toponym disambiguation*, is the follow-on task of assigning a unique location identifier to each entity labelled as a place name. Toponym resolution is similar to word sense disambiguation, in that the context surrounding the place name in the text is used to determine its exact geographic coordinates. We use the Getty Thesaurus of Geographic Names (`http://www.getty.edu/`) to map disambiguated place names to a unique location identifier.

In this paper, these two steps are collectively denoted as *annotation*. Once the annotation of the collection is complete, the next task is to build a *spatial index* which allows spatial relationships to be exploited during a subsequent *querying* phase. Approaches to GIR have been suggested by a range of authors, and there are also cooperative evaluation exercises, most notably the GeoCLEF geographic retrieval track of the Cross Language Evaluation Forum (CLEF), see Gey *et al.* (2006) for details. We make use of various GeoCLEF resources in our experiments.

Most GIR systems use NLP techniques in pre-processing stages, as shown in Figure 1. NERC systems such as LingPipe (`http://alias-i.com/lingpipe`) and OpenNLP (`http://opennlp.sourceforge.net`) have been used to detect toponyms, and gazetteers such as Getty Thesaurus and the World Gazetteer (`http://world-gazetteer.com/`) have been used for disambiguation and location-expansion purposes. For example, Geoffrey (2006) uses LingPipe and Wikipedia (`http://www.wikipedia.org`) for query expansion; and Martins *et al.* (2006) describe a similar system which assigns documents to encompassing geographic scopes and then uses these scopes for document retrieval through a ranking function.

However, to date there has been no component-level evaluation of the NLP techniques in this pipeline architecture in the context of the GeoCLEF initiative. This is perhaps surprising, as the success of such an IR architecture potentially depends on obtaining high levels of accuracy from these pre-processing steps. The stumbling block, of course, is the lack of a human geo-annotated version of the GeoCLEF collection. Ideally the entire GeoCLEF collection (169,477 documents), should be manually marked up in this manner. However, human annotation is time-consuming, and consequently too expensive to be performed on such a large dataset.

One of the most significant recent attempts to provide an evaluation dataset for toponym resolution was proposed by Leidner (2006a). That paper describes an XML-based markup language and an annotation markup editor. A total of 946 news documents from the CoNLL 2003 training corpus (Florian *et al.* 2003) were annotated with longitude/latitude coordinates using this tool. While this evaluation dataset is interesting and useful, it only contains newswire articles from Reuters, largely covering international locations and events. On the other hand, the GeoCLEF collection consists of two local news sources: the *Los Angeles Times* and the *Glasgow Herald.* These collections contain a greater number of less well-known local place names that are typically more challenging to disambiguate due to gaps in gazetteer coverage (the "Highlands" region in Scotland is missing from the Getty gazetteer), and significantly more ambiguity ("Lake Forest" in Orange County has 13 alternative locations in the United States).

Our main objectives in this paper are to address this need for a GeoCLEF-specific evaluation dataset, and to report on the effect of NLP component errors on GIR performance. In addition, we outline a new toponym resolution heuristic that uses geographical information in Wikipedia as an additional source of evidence in the disambiguation process. We also report on a novel indexing strategy and ranking metric.

Many GIR systems, for example those of Li *et al.* (2006c) and Andrade and Silva (2006), combine the scores of textual terms and geographic terms using linear combinations of the form:

$$sim = \alpha \times sim_t + (1 - \alpha) \times sim_g \,,$$

where sim_t is the textual relevance score, sim_g is the geo-relevance score, and $0 \leq \alpha \leq 1$ is a combining factor. Many other systems use similar combinations but fail to properly consider expanded geographic locations. This may explain why many GeoCLEF participants found their IR performance suffered when they used geo-term expansion and introduce geo-relevance into the final score. For example, Hughes (2005) reports a Mean Average Precision (MAP) drop after expanding locations to their subordinates, for example, "Europe" to a list of all European countries. The main reason for this is that expanded location terms tend to dominate the query, an issue that we refer to as *query overloading*.

In our experiments we confirm that location expansion is different from general query expansion used by blind relevance feedback, and that the weighting of the non-locational *topic terms* in a geospatial query must be protected.

3 Preprocessing: Linguistic annotation

In order for a GIR system to be sensitive to geospatial information, the query and the document set must be enriched with a linguistic annotation that not only identifies the place name entities in the text, but also grounds them to a single location. As already noted, this annotation process requires two NLP technologies: toponym detection, and toponym resolution. In this section of the paper, we evaluate these two components using a manually annotated subset of the GeoCLEF collection.

The GeoCLEF data consists of articles taken from two local news publications: the Glasgow Herald, and the Los Angeles Times. We employed a linguistics student to identify all of the locations in a subset containing 302 documents. A second pass through the data was then made by one of the authors, and all annotation disputes resolved by discussion with the student. Leidner (2006a), who used three annotators to generate his CoNLL tagged corpus, found that inter-annotator agreement was high. Our annotator was instructed to distinguish between metonymic references to place names; these references are occurrences of place names in text that are not being used in their literal sense. For example, in the sentence "Washington defends its invasion of Iraq", the word "Washington" is referring to a political entity, the US government, whereas "Iraq" is a reference to a geographic entity. The annotation effort resulted in 2,261 place names

Table 1. Toponym annotation statistics for GeoCLEF test collection used to evaluate the toponym detection and resolution components.

	Glasgow Herald	LA Times	Overall
No. of Tagged Documents	106	196	302
No. of Tagged Toponyms	768	1493	2261
No. of Unique Tagged Toponyms	313	391	704

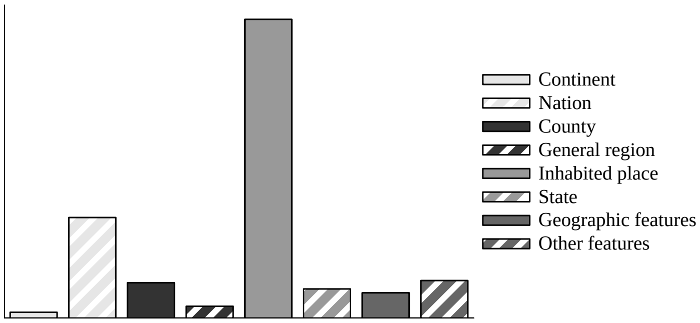

Figure 2. Breakdown of geographic types in the human annotated collection.

being identified and grounded, an average of 7.5 per document. Table 1 provides a breakdown of the annotation. The Getty identifier to which each toponym is bound then allows us to determine the effectiveness of various automatic toponym disambiguation strategies. Figure 2 shows the breakdown of toponym types as defined by the Getty Thesaurus. The most frequent type in our dataset is *inhabited place*; unfortunately Getty uses this label to describe all large and small population centers such as cities, towns and villages; hence, the large frequency count for this type. This annotated collection is available on request from the authors.

The aim of this section is three-fold: to report on the performance of our state-of-the-art toponym resolution and disambiguation systems; to discuss the effects of NERC errors on the disambiguation process; and to report on the quality of the resultant linguistic annotation indexed by the GIR system. This component-level analysis is then used to help explain the observed effects of the annotation errors on GIR performance in Section 5.

3.1 *Toponym detection*

The first step in the annotation process is toponym detection, a task heavily influenced by previous work in Named Entity Detection and Classification (NERC). In the latest official community evaluation at the CoNLL conference, state-of-the-art NERC systems report F-scores of around 90% on newswire data; compared to human performance for this task of 96%–98% (Florian *et al.* 2003). The accuracy of the location classification element of the task is equally impressive, and appears to be no more challenging than the tagging of the other named entity classes. For example, an IBM entry was the top performing system at CoNLL; it achieved F-scores of 91.15%, 84.67% and 93.85% for location, organization and person tagging respectively.

In general, NERC systems use statistical modeling techniques. They require a training set of annotated data, a set of features by which the problem domain can be defined, and a machine learning technique that uses these features to correctly classify unseen instances of named entities. Commonly considered features include ones based on information in the surrounding context such as part of speech tags, chunk tags, orthographic patterns, and trigger words. Many systems also use external knowledge sources, such as gazetteers, that contain commonly occurring people, places and organization. At the CoNLL evaluation the two most popular learning techniques were the Maximum Entropy Model and the Hidden Markov Model (HMM).

During the course of our investigation into geospatial IR, we have primarily focussed on the toponym disambiguation, spatial indexing and query expansion elements of our system architecture. Given the

Table 2. Performance of NERC systems on a hand-annotated subset of the GeoCLEF collection.

System	Recall	Precision	F-score
G_Lookup	79.71%	47.77%	59.74%
LingPipe	71.75%	45.17%	55.44%
OpenNLP	55.61%	52.07%	53.78%

reported success of NERC systems in the literature, we initially assumed that an off-the-shelf solution would be sufficient. Hence, we were surprised to find that LingPipe, a Hidden Markov Modeling approach, and our NERC system of choice, was under-performing on the GeoCLEF data. To confirm our results we also evaluated the OpenNLP NERC system, a Maximum Entropy approach, and implemented a simple gazetteer lookup approach (called G_Lookup) that had no requirement for training data. Results for these methods are shown in Table 2.

In Table 2, *Recall* is defined as the percentage of correctly detected place names with respect to the total number of human annotated place names in the test collection; while *precision* is defined as the percentage of correctly detected place names with respect to the total number of place names tagged by the system. The *F-score* is the harmonic mean of these metrics. We were surprised to find that the G_Lookup system outperformed LingPipe with respect to recall and precision in this experiment.

A possible reason for the disparity between our results and the results reported in the literature is that these off-the-shelf systems run on models that have been trained on American international news sources. These sources tend to contain frequent occurrences of popular place names such as nations, continents and large population centers like capital cities. Local news sources also make reference to international locations, but in addition contain unseen instances that are specific to the locality of the news source. For example, consider place names such as the lake "Loch nan Clar" in Scotland that has an unusual lexical form; and place names that are more commonly used to refer to other named entities such as "Moss Bank" near Liverpool, or the 17 unique locations called "Blair" in the United States that share their name with the British Prime Minister.

Our G_Lookup approach attempts to find matches between capitalized phrases, that is potential proper noun phrases, in the text and entries in the Getty Thesaurus; hence, its high recall score. The number of falsely detected place names is minimized through the use of a stopword list consisting of common person and organization names, and general terms that find matches in the Gazetteer but are not commonly occurring references to locations. For example, the determiner "an" is also a location in Myanmar. As Table 2 shows, it performs relatively well.

3.2 *Toponym resolution*

The next step in the preprocessing pipeline is toponym resolution, which takes the tagged place names from the toponym detection system and adds additional linguistic information to the data by mapping the place name to a unique Getty location identifier. This *Getty Id* provides a link to geometric information such as the geographic coordinates of the location, its place type, and its hierarchical position in the taxonomy.

Toponym resolution is a special case of general word sense disambiguation (WSD), a fundamental NLP task that determines which of the senses of an ambiguous word is being invoked in a given context. Techniques for WSD rely on lexical, syntactic and semantic cues in the surrounding context to distinguish between the senses of a given term as defined in a resource such as WordNet. Similarly, our approach to toponym resolution uses contextual knowledge in conjunction with two external knowledge sources – the Wikipedia online encyclopedia and the Getty Thesaurus – to determine the locations corresponding to occurrences of place names.

Recently the WSD community (McCarthy *et al.* 2004) has focussed on the automatic acquisition of predominant sense information from large domain specific corpora. It is well known that the distribution of word senses is skewed according to the domain of interest. For example, in the biomedical domain the predominant sense of the word "cell" is biological, rather than a reference to a mobile phone or prison accommodation. It has also been shown that a naive WSD approach that simply assigns a term its most frequently occurring sense is as effective as WSD approaches that do not utilize sense tagged training data (McCarthy *et al.* 2004).

Toponym disambiguation techniques have used population statistics to infer dominant sense information (that is, large population centers are more likely to be referred to in news articles). The experiments presented in this paper show that World Gazetteer population data lacks coverage, and is a weaker heuristic than is the information provided by Wikipedia. For example, consider the location "Gaza" in the Palestinian territories. The World Gazetteer has no available population statistics for this location. Searching the Getty gazetteer we find that there are five locations that share this place name, three of which are inhabited places in the United States. Without any predominant sense information, each of these locations is deemed to be equally likely.

Wikipedia (`www.wikipedia.org`) is an online encyclopedia that allows collaborative aggregation of information. Our hypothesis is that the most likely candidate location for a place name will be the Wikipedia page that has the most frequent occurrences of that place name. We use the Wikipedia Geonames webservice (`www.geonames.org`) to find dominant sense information. Geonames is a more useful resource than Wikipedia's search engine for two reasons: firstly, it filters out all non-geographic results from the ranked list; and secondly, it provides geographic coordinates for each location article retrieved which allows us to flag the predominant location of each place name in the Getty Thesaurus. To do this, we search for the candidate in Getty with the closest geographic proximity to the specified coordinates in the Wikipedia page, taking places only if the distance between the two coordinates is less than 20km. After querying the geonames webservice, 9744 Getty Ids were flagged as predominant locations in the Getty taxonomy.

Our TR algorithm works by incrementally applying disambiguation heuristics to the toponyms annotated by the NERC system. For a detailed literature review of commonly used heuristic for TR we refer the reader to (Leidner 2006b). In the first pass over the data, all toponyms that have candidate senses in Getty representing a large land mass (such as a nation or continent) are assigned that Getty Id. The next two passes use information in the local context to determine the correct sense of the toponym. The algorithm first searches for geographic trigger words such as "State", "County", "Mountain" or "River" and then assigns the Getty Id of the candidate sense with the Getty defined geographic type that matches the trigger word. Next, the algorithm identifies adjacent toponyms using a path traversal algorithm that finds the shortest path between their candidate locations in the Getty hierarchy. For example, although a hierarchical relationship exists between Washington, D.C. and Seattle through their shared holonym "United States", the shortest path between these two place names is between the candidates "Washington State" and "Seattle city".

At this point in the disambiguation process there will still be a number of ambiguous toponyms in the text; hence, without sufficient contextual information the algorithm must resort to a back-off or best guess disambiguation strategy. We investigated two such strategies, based on the Wikipedia predominant location information, and the World Gazetteer's population statistics. As a final step, for all remaining ambiguous place names we apply a candidate sense ranking technique that assigns probability scores to location candidates based on their depth in the Getty hierarchy, that is, lower level locations in the hierarchy are smaller geographic entities (for example, a village versus a state) and so are considered less significant.

We evaluated the performance of all of these heuristics on the hand-annotated GeoCLEF dataset. Our best performing combination of heuristics were applied in the order described, with the Wikipedia predominant location information followed by the significance score. The results of this run, labeled $Wiki_{ALL}$ are shown in Table 3. All of the runs in this table use the G_Lookup NERC output, which is our best performing toponym detection system. The Pop_{ALL} run uses the same heuristics except replaces the Wikipedia best guess information with the World Gazetteer population statistics; while the $Wiki_{ONLY}$ is a simple baseline that just assigns all detected toponyms with their predominant location.

Precision scores are frequently reported in WSD experiments; however, they can be somewhat misleading. Precision is the ratio of the number of correct disambiguations to the total number of attempted disambiguations. For example, if the system returns just one tagged term and it is disambiguated correctly then precision is 100%. Instead, we define two recall measures:

Table 3. Performance of our Toponym Resolution systems on the GeoCLEF subset described in Table 1.

System	$Recall_{NERC}$	$Recall_{Human}$
$Wiki_{ALL}$	80.48%	64.15%
Pop_{ALL}	70.33%	56.06%
$Wiki_{ONLY}$	70.55%	56.23%

Table 4. Performance of NERC systems, and NERC systems with added errors (missed locations and falsely detected locations), on a subset of the GeoCLEF data

System	Recall	Precision	F-score
G_Lookup	79.71%	47.77%	59.74%
OpenNLP	55.61%	52.07%	53.78%
LingPipe	71.75%	45.17%	55.44%
LingPipe_R35%	35.94%	45.27%	40.07%
LingPipe_P25%	73.96%	25.87%	38.33%

- $Recall_{NERC}$ is the number of correctly disambiguated toponyms returned by the disambiguator divided by the total number of toponyms correctly detected by the NERC system.
- $Recall_{Human}$ is the number of correctly disambiguated toponyms returned by the disambiguator divided by the total number of human tagged toponyms in the gold standard collection.

The distinction between these metrics is important: $Recall_{NERC}$ tells us how well the disambiguator is performing on the correctly NERC–tagged toponyms in the test collection, while $Recall_{Human}$ factors in the *missed* location errors from the NERC system. Although the $Wiki_{ALL}$ system correctly disambiguates 80.48% of the toponyms identified by the NERC system, only 64.15% of the total number of toponyms in the collection have actually been tagged correctly. This shows us that NERC-missed locations have a significant knock-on-effect on the performance of the disambiguator.

3.3 *Geo-Annotation accuracy*

So far we have presented the toponym detection and disambiguation results using our hand annotated GeoCLEF subset. We have also briefly discussed the effects of NERC missed locations on disambiguation performance. We now examine the *overall* geo-annotation accuracy of the GeoCLEF collection when the accuracy of the NERC and Toponym Disambiguation components are varied. Table 2 showed that the LingPipe, OpenNLP, and G_Lookup toponym detection systems yield a range of NERC precision and recall values. However, we also want to investigate the effect of lower NERC recall and precision values on both disambiguation accuracy and GIR performance. To do this we created a tagged corpus, LingPipe_R35%, that has roughly half LingPipe's original NERC recall score, by removing every second tagged toponym in the original LingPipe NERC output. We then generated a second corpus, LingPipe_P25%, with half LingPipe's original precision score, by changing the tag of an evenly distributed number of person/organization entities making them locations instead. These error addition strategies were employed to keep the original LingPipe precision static, in the case of LingPipe_R35%; and recall static, in the case of LingPipe_P25%. However, neither of these procedures are 100% effective since there is a chance that when tags are either added or removed, correct toponyms will end up being detected, even though we are attempting to increase our errors. However, the characteristics of our three LingPipe annotated datasets, shown in Table 4, indicate that we have been fairly successful as there is only a slight decrease in recall from the original LingPipe dataset to LingPipe_P25% and a similar small precision decrease for LingPipe_R35%.

Table 5 provides a breakdown of the annotation accuracy of these NERC corpora after the $Wiki_{ALL}$ disambiguation algorithm has assigned Getty Ids to all the labelled toponyms. The table's columns represent the types of detection errors that the GIR system will find in the annotated collections:

- Missed_Tag: the percentage of human annotated toponyms that have not been annotated in the *machine tagged* collection. This metric is equivalent to one minus the Recall of the NERC system (as shown in

Table 5. Breakdown of annotation statistics for the $Wiki_{ALL}$ disambiguation algorithm when the NERC performance is varied.

NERC Input	Missed_Tag	False_Tag	NERC FA_Dis	Incorrect_Dis	Ambig_Dis
G_Lookup	20.29%	61.53%	49.15%	8.11%	4.27%
LingPipe	28.25%	57.99%	44.97%	10.29%	2.74%
OpenNLP	44.39%	51.29%	35.30%	12.08%	3.91%
LingPipe_R35%	64.06%	57.27%	44.10%	10.14%	3.02%
LingPipe_P25%	26.04%	65.50%	51.92%	8.60%	4.99%

Table 5) and takes into account the missed location errors made by the NERC system.

- False_Tag: the percentage of incorrectly disambiguated and annotated terms in the machine tagged collection. These false annotations arise from three different pipeline errors:

  - NERC FA_Dis: when falsely detected locations are tagged by the NERC system, two outcomes are possible: the falsely detected location is filtered out by the disambiguation process because it isn't in the Getty Thesaurus; or the falsely detected location has an entry in the Gazetteer and so gets annotated by the disambiguator. The first scenario is a good outcome; however, the second is bad, because it adds annotations errors to the collection. The metric NERC FA_Dis is the percentage of falsely detected locations that are annotated in the machine tagged collection.
  - Incorrect_Dis: the number of correctly NERC detected toponyms that are incorrectly disambiguated and annotated by the disambiguator in the machine tagged collection.
  - Ambig_Dis: the percentage of correctly and incorrectly NERC detected toponyms that are assigned more than one equally likely location candidate in the machine tagged collection.

Note that Ambig_Dis errors, like Incorrect_Dis errors are attributed to the disambiguator only. Our disambiguator assigns a toponym a list of candidate locations with probability scores if it is unable to fully disambiguate it, and so there will also be a percentage of tagged, but ambiguous toponyms in the collection. Our GIR system has been implemented to handle these probabilities in its term weighting formula. These multi-candidate tagged toponyms may have a positive effect on GIR performance, if the list of candidate locations contains the correct candidate. However, we assume these annotation instances are incorrect, and add them to the False Tag error.

Table 5 shows that G_Lookup – the best NERC and disambiguation run – has a relatively low percentage of missed annotations (Missed_Tag), but a high percentage of falsely annotated terms (False_Tag). This trade-off is apparent in all the collections. The breakdown of false tag errors indicates that falsely detected locations in the NERC phase of the pipeline contribute to these errors more than disambiguation errors do. This trend is also observable in the other collections, as their NERC FA_Dis percentages far exceed their Incorrect_Dis percentages.

Comparing the LingPipe and LingPipe_R35 datasets, we can see that by removing NERC tagged locations we have, not surprisingly, significantly increased the percentage of missed annotations (from 30.23% to 64.06%). For LingPipe_P25 we should see an opposite increase in false annotation errors; however, comparing False_Tags percentages with LingPipe we can see that this is not the case. There is only a slight difference in the percentage of false annotations, because the filtering effect of the disambiguation process has removed many of these erroneous annotated locations because they have no entry in the Getty gazetteer. Section 5 explores whether a collection with a high percentage of missed annotations (LingPipe_R35) or a high percentage of incorrect annotations (LingPipe_P35) is worse in terms of overall GIR performance.

The last table in this section, Table 6, provides a breakdown of the annotation accuracy of the tagged collection when the performance of the disambiguator is varied. The four annotated corpora listed in the table were generated by replacing a certain percentage of Getty Ids with a random Getty Id. The introduction of these disambiguation errors was controlled so that each of the corpora contains roughly $Recall_{NERC}$ scores of 60% (Dis_60%), 40% (Dis_40%), 20% (Dis_20%), and 0% (Dis_0%). The same definitions apply to the $Recall_{NERC}$, $Recall_{Human}$, False_Tags and Missed_Tags metrics as in Section 3.2 and Table 5.

Table 6 shows a steady drop in disambiguation performance ($Recall_{Human}$). By comparing the False_Tag scores between Table 5 and Table 6, it is observed that when disambiguation accuracy falls below 80%, it

Table 6. Breakdown of annotation statistics on G_Lookup NERC output when disambiguation performance is varied.

NERC Input	$Recall_{NERC}$	$Recall_{Human}$	Missed_Tags	False_Tags
G_Lookup	80.48%	64.15%	20.29%	61.53%
Dis_60%	61.01%	48.63%	20.29%	70.84%
Dis_40%	40.54%	32.32%	20.29%	80.62%
Dis_20%	20.13%	16.05%	20.29%	90.38%
Dis_0%	0.00%	0.00%	20.29%	100.00%

has a much greater impact on False_Tag scores than the addition of any of the NERC errors did. However, misses remain stable at 20.29%, since all collections are processed by the same NERC system, G_Lookup.

4 Geographic information retrieval

To include geographic information into index and query, specific data structures representing space are used. Our ultimate goal, in this paper, is to determine the extent to which detection and resolution errors affect overall GIR effectiveness. This section briefly describes the GIR system used in our experiments, which operates by adding *textual geo-terms* into the IR index.

4.1 *Hierarchical-based geo-terms and geographic indexing*

A textual geo-term in the geographic index is a special textual term which has a hierarchical format derived from the place name hierarchy of the gazetteer that is being used (Li *et al.* 2006a). For example, the synthetic index term "`@OC-7000490-7006238-7001933`" represents "Melbourne" (7001933) within "Victoria" within "Australia" within "Oceania" (OC). As a shorthand notation to ease the remaining discussion, the location names themselves are used, rather than numeric indicators. For example, "`@OC-7000490-7006238`" is referred to as "`@VICTORIA`".

The standard vocabulary of the GIR system is extended by including locational references as index terms, to various degrees of granularity. Using the geo-terms, a location can be expanded to all its descendants and neighbors, because they share the same prefix and are listed contiguously in the vocabulary.

When tagged locations are encountered during indexing, both the text form of that location (for example, the word "Melbourne") and the geo-term derived from it (for example, "`@MELBOURNE`") are indexed. By adding hierarchical-based geo-terms into the index as if they were "words", we do not need to introduce spatial data structures.

4.2 *Geographic query expansion*

Once locations are indexed in this way, expansion techniques can be used to augment any geo-terms identified in queries. To use this additional information to help retrieve documents when the query is also spatial, Li *et al.* (2006a) describe two types of geo-query expansion: *downward expansion* and *upward expansion*.

Downward expansion extends the influence of a geo-term to some or all of its descendants in the hierarchical gazetteer structure, to encompass locations that are part of, or subregions of, the location specified in the query. For example, in the query "flooding Australia", the geo-term "`@AUS`" can be expanded to seven geo-terms (indicated by the arrows in Figure 3a). Directional operation can be combined with downward expansion as a filter when the query has such spatial relations as "in_south", "in_northeast", by comparing longitudes and latitudes to retain only the expansions that satisfy the spatial relationship.

Upward expansion extends the influence of a geo-term to some or all of its ancestors, and then possibly downward again into other siblings of the original node. This facilitates the expansion of geo-terms in the query to their nearby locations. For example, in the query "hotel Melbourne", the hotels not only within Melbourne but also near it might be of interest (Figure 3b). As with downward expansion, directional operations can be added as a filter.

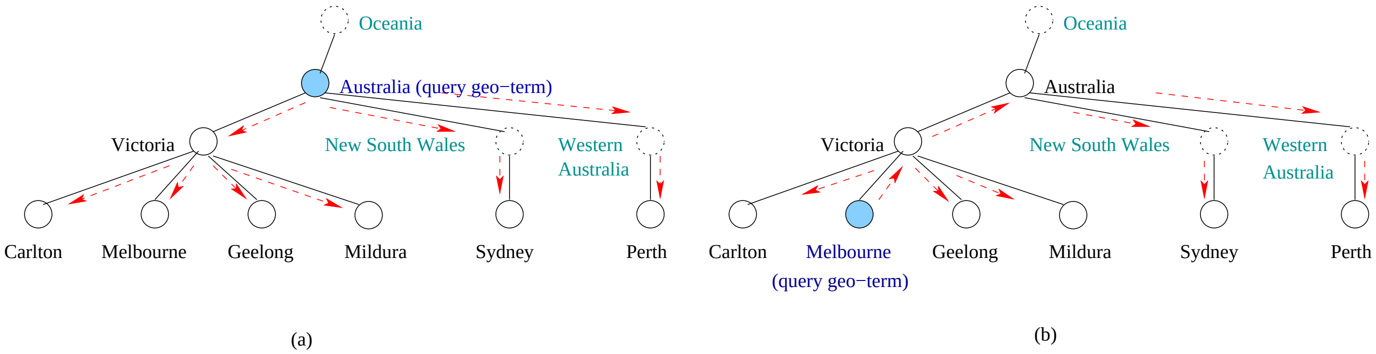

Figure 3. Query expansion: (a) downward from `@AUSTRALIA`; and (b) upward from `@MELBOURNE`. The collection is assumed to contain references to all of the locations indicated by the solid circles. Dashed circles represent locations in the hierarchy that do not appear in the collection.

4.3 *Similarity computation*

When calculating the similarity score, we also need to consider how to add the influence from those expanded geo-terms to the final score. Our first approach is to add up the similarity scores of all the terms and geo-terms, including expanded geo-terms as follows:

$$sim(Q, D_d) = sim_t(Q, D_d) + sim_g(Q, D_d) + sim_{exp}(Q, D_d) \qquad (1)$$

where $sim_t(Q, D_d)$ is the text similarity score which only accounts for terms that are explicitly mentioned in the query, and $sim_g(Q, D_d)$ is the geographic similarity score contributed by geo-terms referred by text location terms, and $sim_{exp}(Q, D_d)$ is the score contributed by geo-terms from down-expansion and up-expansion. They can be evolved from the traditional metric. Our system uses the Okapi similarity computation (Walker *et al.* 1997):

$$\begin{aligned} sim_t(Q, D_d) &= \sum_{t \in Q_t} r_{d,t} \cdot w_t \cdot r_{q,t} && (2)\\ sim_g(Q, D_d) &= \sum_{t \in Q_g} r_{d,t} \cdot w_t \cdot r_{q,t} && (3)\\ sim_{exp}(Q, D_d) &= sim_{down}(Q, D_d) + sim_{up}(Q, D_d) \\ &= \sum_{t \in Q_{down}} \alpha_t \cdot r_{d,t} \cdot w'_t \cdot r'_{qt} + \sum_{t \in Q_{up}} \beta_t \cdot r_{d,t} \cdot w'_t \cdot r'_{qt} \,. && (4) \end{aligned}$$

where

$$\begin{aligned} r_{d,t} &= \frac{(k_1 + 1) \cdot f_{d,t}}{k_1 \cdot [(1-b) + b \cdot \frac{W_d}{avrW_d}] + f_{d,t}} \\ w_t &= \log \frac{N - f_t + 0.5}{f_t + 0.5} \\ r_{q,t} &= \frac{(k_3 + 1) \cdot f_{q,t}}{k_3 + f_{q,t}} \\ w'_t &= \log \frac{N - \max(f_t, f_{t_q}) + 0.5}{\max(f_t, f_{t_q}) + 0.5} && (5) \end{aligned}$$

$$r'_{q,t} = \frac{(k_3 + 1) \cdot f_{q,t_q}}{k_3 + f_{q,t_q}} \,. \tag{6}$$

In these expressions,

- k_1 and b are constants, and usually set to 1.2 and 0.75 respectively, with k_3 set to ∞;
- W_d is the length of the document d in bytes;
- $avrW_d$ is the average document length in the entire collection;
- N is the total number of documents in the collection;
- f_t is the number of documents in which term t occurs;
- $f_{x,t}$ is the frequency of term t in either a document d or query q;
- Q_t and Q_g are aggregations of all query textual terms and geo-terms; Q_{down} and Q_{up} are aggregations of all down or up-expanded geo-terms from the q; and
- α_t and β_t are values less than 1 which can be explained as the expanded geo-terms' similarity to the source query geo-term t_q (for details of how to calculate this similarity, see Li (2007)).

When expanded geo-terms are being processed and w'_t is being calculated, the native f_t and source term f_{t_q} are compared, and the bigger one is used, to normalize the source geo-term and all expanded geo-terms (Equation 5). Similarly, we use the source geo-term's query term frequency f_{q,t_q} for those expanded geo-terms (Equation 6).

4.4 *Query overloading and normalization*

A problem with the simple approach described in the previous section is that with the query expansion, a geo-term might be expanded to thousands of its subordinates and neighbors, potentially swamping all of the non-geo topic terms in the query. Another problem is that even if the influence of the expanded geo-terms is discounted, adding geo-terms makes location terms in a query contribute twice to the final score, because both the location's textual terms and geo-terms are considered. We call these problems *query overloading*.

Our system uses a novel technique in order to strike a balance between textual terms and expanded geo-terms in a query (Li 2007). Query terms are divided into two types: *concept terms* t_c, and *location terms* t_l. For example in the query "wine Australia", "wine" is a concept term, and "Australia" is a location term. The final score for any particular document is generated by summing both concept and locations term similarities:

$$sim(Q, D_d) = sim_c(Q, D_d) + sim_l(Q, D_d) \tag{7}$$

where $sim_c(Q, D_d)$ is the concept similarity score, and $sim_l(Q, D_d)$ is the locational similarity score. The concept similarity score can be denoted as:

$$sim_c(Q, D_d) = \sum_{t \in Q_c} sim_t(Q, D_d) = \sum_{t \in Q_c} r_{d,t} \cdot w_t \cdot r_{qt} \tag{8}$$

where Q_c is the aggregation of all textual concept terms. The location similarity score can be denoted as:

$$\begin{aligned} sim_l(Q, D_d) &= \sum_{t \in Q_l} sim_t(Q, D_d) \\ &= \sum_{t \in Q_l} Norm_t\Big(sim_{text}(Q, D_d), \\ &\qquad sim_{geo1}(Q, D_d), \ldots, sim_{geoT}(Q, D_d)\Big) \end{aligned} \tag{9}$$

where Q_l is the aggregation of all location terms, T is the number of all corresponding geo-terms of a location $t \in Q_l$ including expanded geo-terms, and $Norm_t()$ is a normalization function which adjusts the similarity scores of a location's textual terms, geo-term and its expanded geo-terms.

Li (2007) suggests the use of a normalization regime based around a *geometric progression* of weights. Assume that the T similarity scores from terms belonging to the same location (text, geo-term or expanded geo-terms) after re-ranking into descending order are: sim_1, sim_2, ..., sim_T. Then the final score is given by:

$$Norm(sim_1, \ldots, sim_T) = sim_1 + \frac{sim_2}{a} + \cdots + \frac{sim_T}{a^{T-1}} \tag{10}$$

where $a > 1$.

According to Equation 10, after normalization, the final score $sim_l(Q, D_d)$ of location l will satisfy $sim_l(Q, D_d) \in [sim_1, a/(a-1)sim_1)$. For example, if $a = 3$, the final score of $sim_l(Q, D_d)$ will be between sim_1 and $1.5sim_1$.

The experiments below verify that without query normalization, retrieval performance using query expansion is significantly worse than a traditional IR baseline run.

5 NLP errors and GIR performance

The previous section described the final component in our GIR pipeline: the GIR engine. In this section, our aim is to draw some conclusions on the effects of various NLP errors on GIR performance. Section 3.3 listed nine distinct annotated GeoCLEF corpora that will facilitate this analysis. We now introduce the four IR runs discussed in this section, all of which are built around a modified Zettair retrieval engine (Zettair 2006):

- `unannotated text baseline` is a baseline "bag-of-words" IR approach that indexes an unannotated version of the collection and makes no provision of any sort for geographic references in the query.
- `geo` is a GIR run that queries an annotated version of the collection using all query text terms, the geo-terms (Getty Ids), and the expanded geo-terms. By including location text terms in the query, this run reduces the negative effects of NLP errors such as missed annotations and incorrectly disambiguated toponyms. This run uses the query normalization strategy discussed in the previous section.
- `geo_notxt` is a similar run to `geo` run; but text location terms have been removed from the query, and only their corresponding Getty Id numbers are searched for in the document index.
- `geo_nonorm` is equivalent to the `geo` run except it leaves out the query normalization step.

Tables 7 and 8 show the Mean Average Precision (MAP) scores for these runs on the annotated collections, using the title and description fields in the GeoCLEF 2005 and 2006 queries. The most impressive result from this table is the consistent effectiveness of the query normalization strategy. On all annotated collections and GeoCLEF queries (2005/2006) the average MAP scores of the `geo` run exceed the corresponding `geo_nonorm` scores. In addition, the MAP score variation for each of the `geo` runs on the (relatively) high accuracy annotated corpora and low accuracy corpora is smaller than one might expect (with the exception of a disambiguation accuracy of 0%). This shows that the normalization strategy is effective in reducing the impact of falsely annotated geo terms by ensuring that, even when expanded, they do not swamp the similarity computation. Of course this means, that in some instances using a normalization strategy may dampen the positive impact of geo-term expansion on "appropriate" queries.

Another way of mitigating NLP errors, this time *missed locations* as well as falsely annotated ones, is to include the location text terms in the query as is the case for the `geo` run, which shows consistently better MAP scores than the `geo_notxt` run. However, the rest of our discussion will focus on the results of `geo_notxt` since these runs are more sensitive to annotation errors in the various corpora, and will give us a better understanding of the impact of NLP errors on the GIR performance. All of the following statistical significance test results were performed using a paired one-sided Wilcoxon signed-rank test.

We will first look at the effect on performance when NERC errors are varied. The G_Lookup, LingPipe, OpenNLP and LingPipe_R35 collections have similar NERC precision but different NERC recall values.

Table 7. Retrieval effectiveness when NERC precision and recall are varied, measured as MAP scores for GIR runs using the GeoCLEF 2005/2006 collections.

	NERC recall	NERC precision	GeoCLEF05 data (MAP)			GeoCLEF06 data (MAP)		
			`geo`	`geo_notxt`	`geo_nonorm`	`geo`	`geo_notxt`	`geo_nonorm`
G_Lookup	79.71%	47.77%	0.3557	0.3549	0.2205	0.2606	0.2523	0.2067
LingPipe	71.75%	45.17%	0.3545	0.3443	0.2422	0.2504	0.2368	0.1998
LingPipe_P25	73.96%	25.87%	0.3558	0.3444	0.2349	0.2547	0.2426	0.2000
LingPipe_R35	35.94%	45.27%	0.3416	0.2929	0.2460	0.2461	0.2131	0.2076
OpenNLP	55.61%	52.07%	0.3614	0.3395	0.2819	0.2618	0.2442	0.2230
unannotated text baseline				0.3528			0.2313	

Table 8. Retrieval effectiveness when Toponym Disambiguation accuracy is varied, measured as MAP scores for GIR runs using the GeoCLEF 2005/2006 collections.

		GeoCLEF05 data (MAP)			GeoCLEF06 data (MAP)		
	$Recall_{NERC}$	`geo`	`geo_notxt`	`geo_nonorm`	`geo`	`geo_notxt`	`geo_nonorm`
G_Lookup	80.48%	0.3557	0.3549	0.2205	0.2606	0.2523	0.2067
Dis_60%	61.01%	0.3495	0.3492	0.2400	0.2616	0.2486	0.2097
Dis_40%	40.54%	0.3421	0.3127	0.2411	0.2492	0.2251	0.2030
Dis_20%	20.13%	0.3205	0.2703	0.2443	0.2234	0.1852	0.1897
Dis_0%	00.00%	0.2639	0.1487	0.2507	0.2095	0.0957	0.1649

Looking at the 2005 results for `geo_notxt` in Table 7, we can see that there is a statistically significant difference of 0.0514 (P-value < 0.001) between MAP scores when the recall is halved (LingPipe versus LingPipe_R35), whereas there is only a insignificant difference of 0.01 (P-value $= 0.4$) when NERC precision is halved (LingPipe versus LingPipe_P25). This is an interesting result which concurs with our observation in Section 3.3 that a large number of incorrectly tagged locations added to the LingPipe_P25 corpus have been filtered out by the disambiguation process. Unfortunately, this observation is not evident from the corresponding 2006 results for these runs. Hence, we can only conclude that for the 2005 collection, low NERC recall has a greater impact on retrieval effectiveness than low NERC precision does.

Our second set of GIR experimental results, presented in Table 8 looks at the impact of disambiguation accuracy on GIR performance. Five collections were used in this experiment: G_Lookup, Dis_60%, Dis_40%, Dis_20% and Dis_0%, which have disambiguation accuracies ($Recall_{NERC}$) ranging between 80% and 0%. Looking at G_Lookup and Dis_40%, both the 2005 and 2006 `geo_notxt` runs on these corpora show statistically significant decreases (P-value < 0.001 and P-value $= 0.002$ respectively) in MAP scores when disambiguation accuracy is reduced from 80% to 40%. Hence, we can conclude that halving the disambiguation accuracy (G_Lookup versus Dis_40%) appears to have a greater negative effect on MAP score than halving NERC Recall (LingPipe versus LingPipe_R35) or Precision does (LingPipe versus LingPipe_P25).

However, the most significant finding of all our experiments is that our `geo` run (OpenNLP) outperforms our `unannotated text baseline` on the GeoCLEF 2005 and 2006 topics (a 2.44% and 13.19% gain respectively). Table 5 shows that the OpenNLP annotated collection has the lowest false tag rate of all our collections, which indicates that our GIR system is more robust at dealing with missed locations than false alarms. However, only the OpenNLP 2006 run's gain in MAP score is statistically significant, compared to the baseline. In spite of this, we consider the result interesting since all NLP-based geographic runs submitted by other GeoCLEF participants failed to outperform their baseline runs in the 2005 and 2006 evaluations. This result also suggests that further gains in GIR performance are possible if the accuracy of the toponym detection and resolution NLP components can be improved. However, even with 100% annotation accuracy, it may be that significant gains in effectiveness are unlikely to arise using our current methods.

More specifically, we have observed that many geographical-based queries require more than just geospatial expansion to beat baseline performance (Li *et al.* 2006a,b). Consider the GeoCLEF 2006 query "Snowstorms in North America", the documents that weren't retrieved by the baseline run contain the lexical variation "snow storm", a non-geo expansion term. This observation may explain why GeoCLEF participants who used pseudo-relevance feedback instead of NLP-based geo expansion outperformed all other runs at the 2005 workshop (Petras and Gey 2005).

In addition, the techniques described in this paper only adequately deal with simple queries that require basic geospatial reasoning, such as "cities within a 100km of Frankfurt Am Main". In contrast, consider the query "Ivy League Universities in New England". It is not adequate to simply expand "New England" to its various states and cities, or indeed to expand the concept "Ivy League Universities" to all 8 members. More sophisticated reasoning methods and novel geographical ontologies derived from encyclopedic resources are required for queries where the expansion term set of the non-geo query concept is being influenced by the geo concept and its geospatial operator.

In conclusion then, the contribution of this paper to GIR research is as follows. We have shown that:

- Off-the-shelf NERC taggers are underperforming and require re-training for the GeoCLEF dataset.
- Predominant sense information mined from Wikipedia can improve location resolution performance.
- Query term normalization improves GIR performance by ensuring that the weighting of the non-locational concept terms in a geospatial query is protected. It has also been shown to be effective at dampening the negative impact of NLP errors in the geo-annotated collection on GIR performance.
- Significant gains in GIR will only be made if all query concepts (not just geospatial ones) are expanded, which in many cases will require more sophisticated geospatial reasoning techniques and the development of novel ontological resources that associate many different entities types with geographical coordinates.

References

Andrade, L. and Silva, M.J., 2006, Relevance Ranking for Geographic IR. In: Purves and Jones (2006).

Florian, R., Ittycheriah, A., Jing, H. and Zhang, T., 2003, Named Entity Recognition through Classifier Combination. In *Proceedings of the CoNLL-2003*, pp. 168–171.

Geoffrey, A., 2006, GIR Experimentation. In: Peters (2006).

Gey, F., Larson, R. and et al., 2006, GeoCLEF 2006: the CLEF 2006 Cross-Language Geographic Information Retrieval Track Overview. In: Peters (2006).

Hughes, B., 2005, NICTA i2d2 at GeoCLEF 2005. In: Peters (2005).

Leidner, J.L., 2006a, An Evaluation Dataset for the Toponym Resolution Task. *Computers, Environment and Urban Systems 30(4): 400-417. Special Issue on Geographic Information Retrieval*, **30**, 400–417.

Leidner, J.L., 2006b, Toponym Resolution: A First Large-Scale Comparative Evaluation. Technical report, Research Report EDI-INF-RR-0839 (July 2006), School of Informatics, University of Edinburgh.

Li, Y., Moffat, A., Stokes, N. and Cavedon, L., 2006a, Exploring Probabilistic Toponym Resolution for Geographical Information Retrieval. In: Purves and Jones (2006).

Li, Y., Stokes, N., Cavedon, L. and Moffat, A., 2006b, NICTA I2D2 Group at GeoCLEF 2006. In: Peters (2006).

Li, Y., "Probabilistic Toponym Resolution and Geographic Indexing and Querying", Masters thesis, The University of Melbourne 2007.

Li, Z., Wang, C., Xie, X., Wang, X. and Ma, W., 2006c, Indexing implicit locations for geographical information retrieval. In: Purves and Jones (2006).

Martins, B., Cardoso, N., Chaves, M.S., Andrade, L. and Silva, J., 2006, The University of Lisbon at GeoCLEF 2006. In: Peters (2006).

McCarthy, D., Koeling, R., Weeds, J. and Carroll, J., 2004, Finding predominant senses in untagged text. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pp. 280–287.

Peters, C. (Ed.), *Working Notes for the CLEF 2005 Workshop*, Austria, http://www.clef-campaign.org/2005/working_notes/CLEF2005WN-Contents.htm%l, 2005.

Peters, C. (Ed.), *Working Notes for the CLEF 2006 Workshop*, Spain, http://clef.isti.cnr.it/2006/working_notes/CLEF2006WN-Contents.html, 2006.

Petras, V. and Gey, F., 2005, Berkeley2 at GeoCLEF: Cross-Language Geographic Information Retrieval of German and English Documents. In: Peters (2005).

Purves, R. and Jones, C. (Eds), *Workshop on Geographic Information Retrieval, SIGIR 2006*, Seattle, USA, http://www.geo.unizh.ch/~rsp/gir06/papers, 2006.

Walker, S., Robertson, S., Boughanem, M., Jones, G. and Jones, K.S., 1997, Okapi at TREC-6 Automatic ad hoc, VLC, routing, filtering and QSDR. In *Proceedings of the Sixth Text Retrieval Conference (TREC 6).*

Zettair, "The Zettair Search Engine", http://www.seg.rmit.edu.au/zettair/, 2006.