Query Expansion through Geographical Feature Types

Nuno Cardoso
University of Lisbon, Faculty of Sciences
ncardoso@xldb.di.fc.ul.pt

Mário J. Silva
University of Lisbon, Faculty of Sciences
mjs@di.fc.ul.pt

ABSTRACT

This paper introduces a new approach for expansion of queries with geographical context. The proposed strategy is based on a query parser that captures geonames and spatial relationships, and maps geographical features and feature types into concepts of a geographical ontology. Different strategies for query expansion, according to the geographical restrictions given by the user, are compared. The proposed method allows a more versatile and focused expansion towards the geographical information need of the user.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval; H.3.4 Systems and Software

General Terms

Algorithms, Design

Keywords

Geographical IR, Query Parsing, Ontologies, Query Expansion, Features, Feature Types, Geographical Relevance

1. INTRODUCTION

Over the past years, Geographical IR has drawn considerable interest in the IR community, with the organization of specific workshops [10, 16], a specific evaluation for Geographic IR – GeoCLEF [7, 8] – and dedicated subtasks for geographical query parsing [20]. The focus of Geographic IR is to understand user queries that contain a geographical scope of interest, by retrieving relevant documents for the desired information need and within the geographical restrictions of the query. This is of special relevance for Web IR, as a significant amount of search queries today have an explicit or implicit geographical scope of interest [9, 12, 18].

Like many Geographical IR systems, our approach works with two different axis of relevance: a *thematic axis*, equivalent to the relevance in classic IR systems, and a *geographical axis*, that measures the , in terms of geographical criteria [3]. This division begins in the user queries, which are typically parsed to separately capture the thematic terms (e.g., *5 star Hotels*), the geographical terms that specify scopes of interest (e.g., *Lisbon*), and sometimes a spatial relationship between the theme and the scope (e.g., *in*, *around* or *next*). In this paper, we represent these query parsing fields as <*what,relation,where*> triplets, where the *what* field represents the thematic terms, the *where* field represents the geographical terms and the *relation* field the spatial relationship terms.

Our Geographic IR system uses text mining tools to capture and disambiguate *geonames* (that is, terms that reference a geographical *feature*) in the document collection [19]. We compute a *geographical weight* that represents the affinity of the document to match the geographical scope of the query, using mined data and other geographical related data such as demographics, coordinates, areas or postal codes. This geographical weight is then combined with a standard IR weight such as Okapi BM25 [17], to generate a final list of results relevant both to the thematic and geographical needs of the user [4, 11].

We ground the geonames on the user queries into unique geographical *concepts*, to define of the queries' scope of interest. However, it may not be easy to capture the geographical concepts in many geographical queries and compute the geographical weights, because the user may not be able to conveniently express a geographical information need in the form of a <*what,relation,where*> triplet:

- A user may not know the name of the location of interest, resorting to an indirect description of its location (e.g., *5 star Hotels in the capital of Portugal*), or related places (e.g., *5 star Hotels near the castle of S. Jorge*). This makes it harder for the Geographic IR systems to determine the scope of the query, requiring a reasoning step to infer it (in the given example, *capital of Portugal* must be converted into the corresponding geographical scope, *Lisbon*, and *castle of S. Jorge* to the proper municipality).

- A user may use an alias of a group of scopes (e.g., *5 star Hotels in the Portuguese islands*, which is comprehensibly easier than naming all the Portuguese islands). In order to find the documents of geographical relevance, the query must be expanded to include the relevant geographical concepts representing islands in Portugal.

GIR '07, Lisbon, Portugal

- Spatial relationships may also contain important information for the definition of a sub-region of interest as a geographical scope (e.g., *Restaurants on the coastlines of Portugal*).

These examples of user queries require additional reasoning in order to capture the geographical concept or concepts that suit the geographical criteria. This reasoning can be performed using a *geographical ontology* through mapping the spatial relationships into relationships of the ontology. We can also observe that the examples above share a similar pattern: the use of *feature types* to represent geographical restrictions while expressing an information need. Feature types designate the categories of geographical concepts that we can recognize, such as *rivers*, *islands*, *airports* or *cathedrals* in the physical domain, and *municipality*, *city* or *country* in the administrative domain [1].

Our main motivation for this work is that by recognizing the feature types on the queries and on the documents, and relating them to the spatial relationships and features of the query, we can *guide the query expansion to search for other geographical concepts based on the available geographical restrictions on the query*, fitting the scope of interest of the query. It is our belief that this approach can deal with more complex geographical queries, improving the retrieved results in terms of a geographical relevance.

To illustrate our current approach, consider the query *5 star Hotels in the Portuguese islands*. We start by parsing the query into a triplet, capturing a feature type (*islands*), a feature (*Portugal*, through its adjective form) and a spatial relationship (*in*). With these geographical restrictions, we reason that the final scope corresponds to a list of geographical concepts of feature type *island* that are sub-regions of the concept *Portugal*.

We then use the geographical ontology to obtain the geographical concepts that fit the geographical restrictions, and perform a geographical expansion that includes all 13 Portuguese islands. The final query is *5 star hotel @ São Miguel, Santa Maria, Formigas, Terceira, Graciosa, São Jorge, Pico, Faial, Flores, Corvo, Madeira, Porto Santo, Desertas, Selvagens*, which is an easier query for the geographical ranking step to process. Now, a document mentioning a 5 star hotel in the Madeira islands can be tracked as being within the desired scope of interest.

The rest of this paper is organized as follows: in Section 2 we detail the query parsing algorithm that captures the triplets of the geographical queries. In Section 3 we describe how we map the spatial relationships into ontological relationships. Section 4 presents our different strategies and algorithms for geographical query expansion, and Section 5 concludes our paper with some final remarks.

2. QUERY PARSING ALGORITHM

The main input to the geographical query parsing and query expansion processes is a geographical ontology of *concepts* and *relationships* between concepts. The geographical concepts are unambiguous entries with unique identifiers, which are associated to places and their metadata, such as alternative and adjective names, demographic data, coordinates or feature types. The relationships represent the types of associations between geographical concepts, such as *part-of* (e.g., *Portugal is* part-of *Europe*) and *adjacent-to* (e.g., *Portugal is* adjacent-to *Spain*).

We capture and disambiguate all geonames on the query, followed by the feature types. After this step, we have enough geographical context to capture the remaining parts, *what* and *relation*. This approach appears to be successful, provided that the ontology contains the necessary geographical information.

The query parsing algorithm is inspired on a previous algorithm developed for the GeoCLEF 2006 task [14]. We now describe the steps involved in extracting the triplets of the user queries.

Pre-processing and capturing geonames

We begin by tokenising and generating word n-grams from the query string. The n-grams are then matched against all names associated to geographical concepts on the ontology. The n-gram comparison follows a greedy match, going from the largest n-gram to the smallest n-gram. This enables capturing *South Africa* before *Africa*, for example. All adjective names, such as *Portuguese*, are also matched and then grounded to the proper geographical concept of the ontology.

The plausibility of these matches is then verified with the help of a gazetteer with manually created context rules. The gazetteer describes contexts where geonames are also used in non-geographical context, such as people names (e.g., *George Washington*) or corporation/foundation names (e.g., *The Seattle Foundation*). When found, the label of the geonames is removed.

Feature type pattern matching

We use context rule patterns to look up for feature types, such as <[*geoname*] [*feature type*]> (e.g., *Portuguese islands*). These rule patterns include all feature type names included in the geographical ontology. When found, the feature types are labeled as such, unless if they are already tagged (e.g. the term *Islands* in the query *Hotels in the Cayman Islands* is not tagged as a feature type, as it belongs to the name of the geographical concept *Cayman Islands*, and therefore it is kept untouched).

Relationship pattern matching

After all geographical terms are labeled, we capture the spatial relationships with the help of spatial relationship rule patterns that take in consideration the context terms and labels, such as <*south of* [*geographical terms*]> or <*between* [*geographical terms 1*] *and* [*geographical terms 2*]>.

The [*geographical terms*] slots represent groups of terms that are labeled as features or feature types. By using this geographical context, there is no need to validate if the captured relationships are really spatial relationships after the match. Initial ad-hoc tests with geographical query parsing showed that we rarely capture wrong spatial relationships with this approach.

In the end, the remaining untagged terms are treated as part of the thematic side of the query, and therefore labeled as such.

3. MAPPING SPATIAL RELATIONSHIPS TO ONTOLOGICAL RELATIONSHIPS

A key process in Geographic IR is the capture of the geonames present in both documents and queries, and their grounding into geographical concepts found on the geographical ontology [6, 13]. Only then we can have disambiguated scopes to calculate the geographical similarity between documents and queries.

A correct interpretation of spatial relationships has an important role in the geographical query expansion. In the query parsing step, we map the spatial relationships found in the queries into the proper relationships contained in the geographical ontology. For the geographical query expansion step, we have a different expansion strategy for each type of spatial relationship, to suit the semantics of the queries. A geographical ontology typically contains *part-of* and *adjacent-to* relationships. For instance, the geographical ontologies that we have created so far with GKB, our geographical knowledge base, includes these two types of relationships [5].

We now detail how the spatial relationships of the queries are matched to the ontology relationships, and then used for driving the geographic query expansion.

part-of relationship

The *in* relationship is the most common spatial relationship found on geographical queries [9], denoting that the user is interested on documents inside the boundaries or from sub-regions of the given scope of interest. The relationships *in*, *of*, *on the*, *at* and other similar spatial relationships are mapped to the *part-of* relationship of the ontology. The geographical query expansion then searches for candidate geographical concepts for expansion that contain a *part-of* relationship to the initial query scope.

adjacency relationship

Spatial relationships in queries denoting proximity, such as *around*, *next to* or *within X km of*, are mapped to the *adjacent-to* relationship in the geographical ontology.

In this particular case, the semantics of the proximity relationships may have distinct interpretations [6]. Consider, for instance, the query *Amusement parks near Lisbon*; is the user interested on amusement parks that are only located on the surroundings of Lisbon, or is he interested in amusement parks both inside Lisbon and in its surroundings? We chose to expand the query scope to all geographical concepts of the same feature type that have an *adjacent-to* relationship with the initial scope, together with geographical scopes found with a *part-of* relationship with the initial scope. In the given example, this translates to all adjacent cities of Lisbon and to geographical concepts that are part of Lisbon.

The geographical query expansion system can also easily support a strict adjacency interpretation of some proximity relationships, such as *in the outskirts* or *on the surroundings*, by mapping them to the *adjacent-to* ontological relationship and searching only for geographical concepts with *adjacent-to* relationship to the initial concept.

Other spatial relationships

We believe that there is no need to support more ontological relationships for the geographical query expansion. We can *combine the* part-of *and* adjacent-to *relationships to fit the semantics for other spatial relationships* of the query with the help of feature types. For instance, we do not need to specify a relationship *on the shores of* in the ontology, as in the spatial relationship of a query like *Vineyards on the shores of Douro river*, because we can infer that the scopes of interest of the query are all geographical concepts of feature type *river* that have a *part-of* or an *adjacent-to* ontology relationship to other geographical concepts of all feature types excluding *ocean*, *river*, *sea* and *lake*.

Another example on how we adjust the geographical query expansion to the spatial relationship of the query and within the available ontology relationships is illustrated by the query *Shipwrecks between Portugal and the USA*. In order to determine the query scope in the form of a group of geographical concepts for a *between* spatial relationship, we generate two groups of geographical concepts: the first contains all expanded geographical concepts that are *adjacent-to* the geographical concept *Portugal*, and the second contains all expanded geographical concepts that are *adjacent-to* the geographical concept *USA*. The final scope correspond to the concepts contained in the intersection of these two groups (e.g., *Atlantic Ocean*).

4. EXPANDING GEOGRAPHIC QUERIES

We classify user queries with geographical content in three types:

1. Queries specifying the scope through features only, that is, queries in the format <*what, relation, feature*> (e.g., *Hotels in Lisbon*).

2. Queries with feature types as the query scope only, that is, queries in the format <*what, relation, feature type*> (e.g., *Hotels in islands*).

3. Queries with both features and feature types as the query scope, that is, queries in the format <*what, relation, feature & feature type*> (e.g., *Hotels in islands of Portugal*).

We now describe how we geographically expand each of these query types.

Queries with references to geographical features only

This type of query is typically found in <*what, relation, feature*> triplets or in <*what, feature*> duplets. When the spatial relationship is not present, it is assumed by default an *in* spatial relationship, directly mapped into a *part-of* ontology relationship. The geographical query expansion starts from the initial geographical concepts captured from the geonames, and searches for additional geographical concepts that are *part-of* the initial geographical concepts.

Since we may have hundreds or even thousands of geographical concepts that can be part of another geographical concept (for example, all features inside Europe), we need some heuristics to control the direction of the geographical expansion. Consider, for example, the geoname *Lisbon*; there are several universities, airports and parks that are part of the *city of Lisbon*, and the *city of Lisbon* itself is part of an administrative division. *Lisbon* is also adjacent to a river, to other cities, and roughly adjacent to seas, beaches, coastlines and mountains in its surroundings. So, when the user issues a query, like *Shopping in or around Lisbon*, the question is which type of geographical concepts would be best to include or to leave excluded in the geographical query expansion.

We believe that the use of associated *feature types* is key to the solution. We can reason about how to guide the geographical expansion with the help of the thematic part of the query by analysing the feature types contained in the returned documents. We illustrate this heuristic with the GeoCLEF 2006 query *Shipwrecks in Atlantic*. The thematic part of the query is submitted to our IR system (that is, *Shipwrecks*) and an initial result set is obtained, based solely on textual ranking. Our intuition is that it is more likely that documents about shipwrecks contain geonames that can be grounded to concepts with feature types like *oceans*, *seas*, *islands* or *beaches*, instead of feature types like *airports*, *municipalities* or *highlands*. To determine which feature types should be more promising for the geographical query expansion, we analyse the frequency of feature types on the top ranked documents from the initial result set. Algorithm 1 describes the adopted geographical query expansion method for this type of queries.

```
Require: O = a geographic ontology
Require: GQ = the geographical query
Define: ORS = ontological relationship
{WHAT,SPATREL,WHERE} = extract_elements(GQ);
if SPATREL != null
    ORS = map_relationship(SPATREL);
else
    ORS = part-of;
Define: RS = {};
RS = retrieve_BM25(WHAT);
MFFT = get_most_frequent_feature_type(RS);
Define: GC = a geographical concept in O;
Define: GQEL = geographical query expansion list;
Define: LF = List of features in the query;
LF = get_features(WHERE);
foreach feature F in LF do
    foreach GC that is ORS to F
        if GC.feature_type == MFFT then
            GQEL.append(GC);
        end if
    end for
end for
return({WHAT,ORS,GQEL});
```

Algorithm 1: Query expansion with features only.

Queries with references to feature types only

This type of queries do not have geonames that can be grounded immediately into geographical features to serve as basis for hierarchical query expansion, but have a reference to the feature type for the relevant scope of interest instead.

In some cases, the query does not even have a thematic part, but only feature types. An example is the following query taken from the GeoCLEF 2006 evaluation, *cities near volcanoes*. One can argue that *cities* may be considered the thematic part of the query instead of a feature type, but this approach forces the retrieval of documents with the term *cities* (or a morphological variant), taking a chance on drifting query expansions [15], and probably bypassing documents like this (fictional) one:

> (...) *Indonesian authorities have started the evacuation of Yogyakarta, with is 18 miles only of the Merapi, that bursted yesterday into several eruptions* (...)

Our system can also handle queries with no thematic part properly. For these queries, the text ranking is not invoked, and the final result set is ranked only by the geographical weighting module. There is a significant amount of queries only concerned with a geographical information need, e.g. *Countries next to France*, or *Islands on the Indic Ocean*, where the textual similarity schemes do not help in finding relevant documents.

Since we captured two feature types (*cities* and *volcanoes*) that are unrelated to any feature and are connected with a *near* spatial relationship, the geographical query expansion searches for all geographical concepts of feature type *city* found in the ontology that have an *adjacent-to* relationship to at least one geographical concept of feature type *volcano*. The document retrieval is based solely on geographical indexes, and the ranking is calculated with the help of geographical similarity heuristics [2, 11]. Algorithm 2 describes the geographical query expansion method for queries with feature types only.

```
Require: O = a geographic ontology
Require: GQ = the geographical query
Define: ORS = ontological relationship
{WHAT,SPATREL,WHERE} = extract_elements(GQ);
if SPATREL != null
    ORS = map_relationship(SPATREL);
else
    ORS = part-of;
Define: GC = a geographical concept in O;
Define: GQEL = geographical query expansion list;
Define: LFT = List of feature types;
LFT = get_feature_types(WHERE);
foreach GC in O do
    if GC.feature_type contained in LFT then
        GQEL.append(GC);
    end if
end for
return({WHAT,ORS,GQEL});
```

Algorithm 2: Query expansion with feature types only.

Queries with references to features and feature types

This type of queries fit a <*what, relation, feature & feature type*> triplet. One example is the query taken from the GeoCLEF 2007 evaluation, *sea traffic around Portuguese islands*, which contains a feature that can be easily grounded into the corresponding geographical concept (*Portugal*) and a feature type (*island*).

If we ignore the feature type, we can mistakenly blindly expand the query scope to all geographical concepts that are sub-regions of Portugal, including continental regions. So, we perform a selective query expansion taking in consideration two types of restriction: feature type (*island*) and the spatial relationship (*in*). If the feature type is associated to the geographical concept (e.g., *Cuba island*), the feature type only enforces the disambiguation of the geographical concept. If it is not associated (e.g., *Portugal island*, and there is no such island with that name on the ontology), the geographical query expansion is driven to search for all geographical concepts of feature type *island* that contain a *part-of* relationship to the geographical concept *Portugal*.

This approach is also valid for queries with features and feature types, but with no thematic part (ex: *Portuguese islands*). These are handled without the contribution of the textual ranking module, relying only on the geographical indexes for the retrieval and in the geographical ranking module for sorting the documents. Algorithm 3 describes the geographical query expansion method for queries with features and feature types.

```
Require: O = a geographic ontology
Require: GQ = the geographical query
Define: ORS = ontological relationship
{WHAT,SPATREL,WHERE} = extract_elements(GQ);
if SPATREL != null
    ORS = map_relationship(SPATREL);
else
    ORS = part-of;
Define: GC = a geographical concept in O;
Define: GQEL = geographical query expansion list;
Define: LF = List of features;
Define: LFT = List of feature types;
LF = get_features(WHERE);
LFT = get_feature_types(WHERE);
foreach feature F in LF do
    foreach GC that is ORS to F
        if GC.feature_type contained in LFT then
            GQEL.append(GC);
    end if
end for
return({WHAT,ORS,GQEL});
```

Algorithm 3: query expansion with features and feature types.

5. FINAL REMARKS

We presented our new approach for geographical query parsing and geographical query expansion for Geographic IR, based on the use of *feature types* for grounding geonames and expanding geographical features.

The query parsing module captures spatial relationships, feature types and features on the query separately, so that the geographical query expansion module can readjust its expansion strategy according to the semantics of the query. With a correct matching of the features into ontology concepts, and spatial relationships into ontology relationships, we direct the geographical query expansion process into the intended scope of interest.

Our work on geographical query expansion with feature types is only at the beginning, as there are many other interesting geographical reasoning opportunities to be explored. For instance, we can reason implicit knowledge over the relationships between feature types, which might be helpful when cross-expanding from the physical and administrative domains (e.g., by knowing that oceans only contain islands, we can guide the expansion step of a query like *Shipwrecks in Atlantic* to include islands in the Atlantic Ocean on the query scope).

Our geographical query expansion module uses direct relationships between geographical concepts only. In other words, it does not infer that if A is *part-of* B and B is *part-of* C, then A is *part-of* C. We believe that the reasoning capabilities of the geographical query expansion module may improve considerably with such inferences and with the help of feature types. Yet, we are also aware that this kind of inference must be carefully made, as it may include undesirable concepts (e.g., in the query *cities near the Atlantic*, we can reason that the Atlantic is *adjacent-to* Portugal, but not all cities of Portugal are adjacent to the Atlantic).

Our first quantitative evaluation results will be obtained once the analysis of the runs we submitted to GeoCLEF 2007 is completed.

Acknowledgements

This work was jointly funded by the Portuguese government and the European Union (FEDER and FSE) under contract ref. POSC/339/1.3/C/NAC (Linguateca), and partially supported by grants SFRH/BD/29817/2006 and POSI/SRI/40193/2001 (GREASE) from FCT (Portugal), co-financed by POSI. We would like to thank the other members of the GREASE project, in particular Marcirio Chaves and Catarina Rodrigues, for developing the GKB and managing the geographical data, Joana Campos for developing the text mining tool, and David Cruz for developing the geographical ranking module and implementing the geographical similarity heuristics.

6. REFERENCES

[1] ISO 19109:2005 - Geographic Information - Rules for Application Schema, 2005. Available at `http://www.iso.org/`.

[2] L. Andrade and M. J. Silva. Relevance Ranking for Geographic IR. In *Proceedings of the 3rd ACM Workshop on Geographical Information Retrieval, GIR'2006*, Seattle, Washington, USA, 10th August 2006.

[3] G. Cai. GeoVSM: An Integrated Retrieval Model for Geographic Information. In *Proceedings of the 2nd International Conference on Geographic Information Science, GIScience'02*, pages 65–79, Boulder, Colorado, USA, 25–28th September 2002. Springer.

[4] N. Cardoso, B. Martins, M. Chaves, L. Andrade, and M. J. Silva. The XLDB Group at GeoCLEF 2005. In C. Peters et al., editor, *Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evalution Forum*, volume 4022 of *LNCS*, pages 997–1006. Springer, 2006.

[5] M. Chaves, M. J. Silva, and B. Martins. A Geographic Knowledge Base for Semantic Web Applications. In C. A. Heuser, editor, *Proceedings of the 20th Brazilian Symposium on Databases*, pages 40–54, Uberlândia, Minas Gerais, Brazil, 3–7th October 2005.

[6] G. Fu, C. B. Jones, and A. I. Abdelmoty. Ontology-Based Spatial Query Expansion in Information Retrieval. In *OTM Conferences (2)*, volume 3761 of *LNCS*, pages 1466–1482. Springer, 2005.

[7] F. Gey, R. Larson, M. Sanderson, K. Bischoff, T. Mandl, C. Womser-Hacker, D. Santos, P. Rocha, G. M. D. Nunzio, and N. Ferro. GeoCLEF 2006: the

CLEF 2006 Cross-Language Geographic Information Retrieval Track Overview. In A. Nardi, C. Peters, and J. L. Vicedo, editors, *Cross Language Evaluation Forum: Working Notes for the CLEF 2006 Workshop*, Alicante, Spain, 20–22nd September 2006.

[8] F. Gey, R. Larson, M. Sanderson, H. Joho, P. Clough, and V. Petras. GeoCLEF: the CLEF 2005 Cross-Language Geographic Information Retrieval Track Overview. In C. Peters et al., editor, *Accessing Multilingual Information Repositories: 6th Workshop of the Cross-Language Evalution Forum*, volume 4022 of *LNCS*, pages 908–919. Springer, 2006.

[9] J. Kohler. Analysing Search Engine Queries for the Use of Geographic Terms. Master's thesis, University of Sheffield, 2003.

[10] A. Kornai and B. Sundheim, editors. *HLT-NAACL 2003 Workshop: Analysis of Geographic References.* ACL, Edmonton, Canada, 31st May 2003.

[11] B. Martins, N. Cardoso, M. Chaves, L. Andrade, and M. J. Silva. The University of Lisbon at GeoCLEF 2006. In A. Nardi, C. Peters, and J. L. Vicedo, editors, *Cross Language Evaluation Forum: Working Notes for the CLEF 2006 Workshop*, Alicante, Spain, 20–22nd September 2006.

[12] B. Martins and M. J. Silva. A Statistical Study of the Tumba! Corpus. Technical report, Faculty of Sciences, University of Lisbon, May 2004. TR DI/FCUL TR 4-4.

[13] B. Martins and M. J. Silva. A Graph-Ranking Algorithm for Geo-Referencing Documents. In *Proceedings of the 5th IEEE International Conference on Data Mining, ICDM'2005*, pages 741–744, Houston, Texas, USA, 27–30th November 2005.

[14] B. Martins, M. J. Silva, S. Freitas, and A. P. Afonso. Handling Locations in Search Engine Queries. In *Proceedings of the 3rd ACM Workshop on Geographical Information Retrieval, GIR'2006*, Seattle, Washington, USA, 10th August 2006.

[15] M. Mitra, A. Singhal, and C. Buckley. Improving Automatic Query Expansion. In *Proceedings of the 21st Annual International ACM Conference on Research and Development in Information Retrieval, SIGIR'1998*, pages 206–214, Melbourne, Australia.

[16] R. Purves and C. Jones. Workshop on Geographic Information Retrieval. *Computers, Environment and Urban Systems*, 30(4):375–377, 2006.

[17] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In D. K. Harman, editor, *Proceedings of the 3rd Text REtrieval Conference*, pages 109–126. NIST Special Publication 500–225, 1995.

[18] M. Sanderson and J. Kohler. Analyzing Geographic Queries. In *Proceedings of the 1st ACM Workshop on Geographic Information Retrieval, GIR'2004*, Sheffield, UK, 2004.

[19] M. J. Silva, B. Martins, M. S. Chaves, N. Cardoso, and A. P. Afonso. Adding Geographic Scopes to Web Resources. *CEUS - Computers, Environment and Urban Systems*, 30(4):378–399, July 2006.

[20] X. Xie. Query Parsing Task Proposal for GeoCLEF 2007. `http://ir.shef.ac.uk/geoclef/2007/Query-Parsing.htm`.