

Query Parsing Task for GeoCLEF2007 Report

Zhisheng Li¹, Chong Wang², Xing Xie², Wei-Ying Ma²

¹Department of Computer Science, University of Sci. & Tech. of China, Hefei, Anhui, 230026, P.R. China

zsli@mail.ustc.edu.cn

²Microsoft Research Asia, 4F, Sigma Center, No.49, Zhichun Road, Beijing, 100080, P.R. China

{chwang, xingx, wyma}@microsoft.com

Abstract

Geo-query parsing task is a sub-task in GeoCLEF2007 and it is run by Microsoft Research Asia. We have provided a query set of 800,000 real queries (in English) from MSN search. The proposed task requires that, based on the provided query set, the participants first identify the local queries and then analyze the different components for the local queries. We have provided a sample labeled query set of 100 queries for training. There are six valid submissions from six teams. We selected 500 queries to form our evaluation set. Under a rather strict evaluation criterion and metric, the Miracle team achieves the highest F1-score, 0.488, and the highest Recall too, 0.566, while the Ask team achieves the highest Precision, 0.625.

Keywords

Geographical Information Retrieval, Query Parsing, Task Design, Evaluation

1. Introduction

Geographical Information Retrieval (GIR) is becoming more and more important nowadays. GIR is more related to people's daily life than general search because people need to deal with locations all the time, such as dining, traveling and shopping. Many large Internet companies are paying more attention to GIR, such as Microsoft, Google, Yahoo. Among many problems in GIR, query location detection is one of the most important problems. This is the first step when GIR system tries to understand the intentions of the queries. Accurately and effectively detecting and analyzing the locations where search queries are truly about has huge potential impact on increasing search relevance.

A local query often contains some non-location words, too. In general, a local query is usually composed of three components, "what", "relation-type" and "where". The keywords in the "what" component indicate what the user wants to search. The "where" indicates the geographical area that the user wants to know. The "relation-type" indicates the relationship between "what" and "where". For example, for such a query "Restaurant in Beijing, China", "what" is "Restaurant", "where" is "Beijing, China", and "Relation-type" is "IN"; while for another query "Mountains in the south of United States", "what" is "Mountains", "where" is "United States", and "Relation-type" is "SOUTH-OF". How to extract these components from queries is one of the key problems for GIR. If the problem is well solved, the GIR system can handle the different components more efficiently and effectively. Therefore, we conducted such a task for GeoCLEF2007 to test the performance of the query tagging algorithm.

2. Task design

Our goal in this query tagging task is to identify the local query and extract the corresponding three components described above. Moreover, we define three types according to the "what" terms, which are "Yellow page", "Map" and "Information". Here we restrict the local query as the query containing EXPLICIT locations.

In our query set, a common local query structure will be "what" + "geo-relation" + "where". The keywords in the "what" component indicate what users want to search; "where" indicates the geographic area users are interested in; "geo-relation" stands for the relationship between "what" and "where". There also exist non-local queries in our query set which also need to be recognized.

For example, for a local query "Restaurant in Beijing, China", "what" = "Restaurant", "where" = "Beijing, China", and "geo-relation" = "IN". For another query, "Mountains in the south of United States", "what" = "Mountains", "where" = "United States", and "geo-relation" = "SOUTH_OF".

2.1 Tasks Description

- 1) Detect whether the query is a local query or not. A query is defined to be “local” if a query contains at least a “where” component. For example, “pizza in Seattle, WA” is a local query, while “Microsoft software” is a non-local query. For non-local queries, further processing is not needed.
- 2) If the query is local, extract the “where” component and output the corresponding latitude/longitude. For example, in the query “pizza in Seattle, WA”, “Seattle, WA” will be extracted and lat/long value (47.59, -122.33) will be output. Sometimes terms in the “where” component are ambiguous. In this case, the participant should output the lat/long value with the highest confidence. A few queries contain multiple locations, for example, “bus lines from US to Canada”. We try our best to avoid this kind of queries appearing in our query set.
- 3) Extract the “geo-relation” component from the local query and transform it into a pre-defined relation type. A suggested relation type list is shown in Table 1. If the relation type you find is not defined in Table 1, you should categorize it into “UNDEFINED”.

Table 1. Relation-Type

| Example query | Geo-relation |
|-----------------------------|---------------|
| Beijing | NONE |
| in Beijing | IN |
| on the Long Island | ON |
| of Beijing | OF |
| near Beijing | NEAR |
| next to Beijing | |
| in or around Beijing | IN_NEAR |
| in and around Beijing | |
| along the Rhine | ALONG |
| at Beijing | AT |
| from Beijing | FROM |
| to Beijing | TO |
| within d miles of Beijing | DISTANCE |
| north of Beijing | NORTH_OF |
| in the north of Beijing | |
| south of Beijing | SOUTH_OF |
| in the south of Beijing | |
| east of Beijing | EAST_OF |
| in the east of Beijing | |
| west of Beijing | WEST_OF |
| in the west of Beijing | |
| northeast of Beijing | NORTH_EAST_OF |
| in the northeast of Beijing | |
| northwest of Beijing | NORTH_WEST_OF |
| in the northwest of Beijing | |
| southeast of Beijing | SOUTH_EAST_OF |
| in the southeast of Beijing | |
| southwest of Beijing | SOUTH_WEST_OF |

| | |
|-----------------------------|---------------|
| in the southwest of Beijing | |
| north to Beijing | NORTH_TO |
| south to Beijing | SOUTH_TO |
| east to Beijing | EAST_TO |
| west to Beijing | WEST_TO |
| northeast to Beijing | NORTH_EAST_TO |
| northwest to Beijing | NORTH_WEST_TO |
| southeast to Beijing | SOUTH_EAST_TO |

- 4) Extract the “what” component from the local query and categorize it into one of three predefined types, which are listed below:
- Map type**, users are looking for natural points of interests, like river, beach, mountain, monuments, etc.
 - Yellow page type**, users are looking for businesses or organizations, like hotels, restaurants, hospitals, etc.
 - Information type**, users are looking for text information, like news, articles, blogs, etc.

2.2 Data Set

We provided a query data set of 800,000 queries. The queries were selected from MSN search logs collected over fifteen days in Aug. 2006. The queries can be classified as four types according to whether they contain locations or geo-relations or not. Table 2 shows the number of four type queries in the data set. Here “relation” means the types listed in Table 1.

Table 2. Composition of the Data Set

| | Has location terms | No location terms |
|------------------------|--------------------|-------------------|
| Has geo-relation terms | 50,000 | 20,000 |
| No geo-relation terms | 350,000 | 380,000 |

And we provided a sample labeled query set of 100 queries for participants. The format is described in the following section.

2.3 Format

2.3.1 Data Set Format

The query set is provided in XML format. Each query has two attributes: <QUERYNO> and <QUERY>. Examples:

```
<QUERYNO>1</QUERYNO>
<QUERY>Restaurant in Beijing, China</QUERY>
<QUERYNO>2</QUERYNO>
<QUERY>Real estate in Florida</QUERY>
<QUERYNO>3</QUERYNO>
<QUERY>Mountains in the south of United States</QUERY>
```

2.3.2 Training Set and Result

The sample labeled set and the results are in the following format. There are 4 more attributes: <LOCAL>, <WHAT>, <WHAT-TYPE>, <GEO-RELATION> and <WHERE>.

```
<QUERYNO>1</QUERYNO>
<QUERY>Restaurant in Beijing, China</QUERY>
<LOCAL>YES</LOCAL>
<WHAT>Restaurant</WHAT>
```

```

<WHAT-TYPE>Yellow page</WHAT-TYPE>
<GEO-RELATION>IN</GEO-RELATION>
<WHERE>Beijing, China</WHERE>
<LAT-LONG>40.24, 116.42</LAT-LONG>
<QUERYNO>2</QUERYNO>
<QUERY> Lottery in Florida</QUERY>
<LOCAL>YES</LOCAL>
<WHAT>Lottery</WHAT>
<WHAT-TYPE>Information</WHAT-TYPE>
<GEO-RELATION>IN</GEO-RELATION>
<WHERE>Florida, United States</WHERE>
<LAT-LONG>28.38, -81.75</LAT-LONG>

```

If a submission from a team does not contain all queries or all the fields, those absent queries or fields will be treated as errors.

3. Evaluation

The contest is open to any party planning to attend GeoCLEF 2007. A person can participate in only one group. Multiple submissions are allowed before the deadline, but we only evaluated the last submissions.

The participants take the responsibility of obtaining any permission to use any algorithms/tools/data that are intellectual property of third party.

3.1 Evaluation Set

To evaluate the performance of the submission, we choose a set of queries from the query set to form an evaluation set. However, if all the queries are chosen randomly, there will be several problems as follows. 1) There are some typos in the queries, e.g. “beuty”; 2) The query is ambiguous and difficult to understand. For example, “Cambridge”, “daa files”; 3) Many geo-relations don’t appear very often, e.g. “NORTH_EAST_TO”, “NORTH_OF”, so it is difficult to include this kind of cases in the evaluation set if the queries are chosen randomly. So we choose the following steps to construct the final evaluation set to cover as many different types as possible.

- 1) Choose 800 queries randomly from the query set.
- 2) Remove the typos and the ambiguous queries from the 800 ones manually.
- 3) Select the queries with special geo-relations from the remainder queries in the query set manually and add them to the evaluation set.
- 4) Select 500 queries for the final evaluation set.

3.2 Distribution

Figure 1 shows the distribution of the evaluation set. The three types of queries, including map, information and yellow page, consist of the local queries which occupy 61.4%.

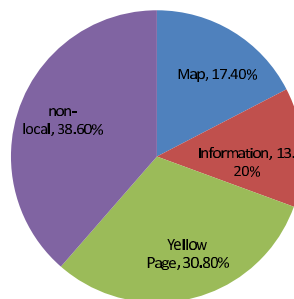


Figure 1. Distribution of the evaluation set

3.3 Labeling approach

3.3.1 Labeling tool

To accelerate the labeling efficiency, we design a labeling tool. With its help we can easily identify each part of the query. Figure 2 shows the interface of the tool.

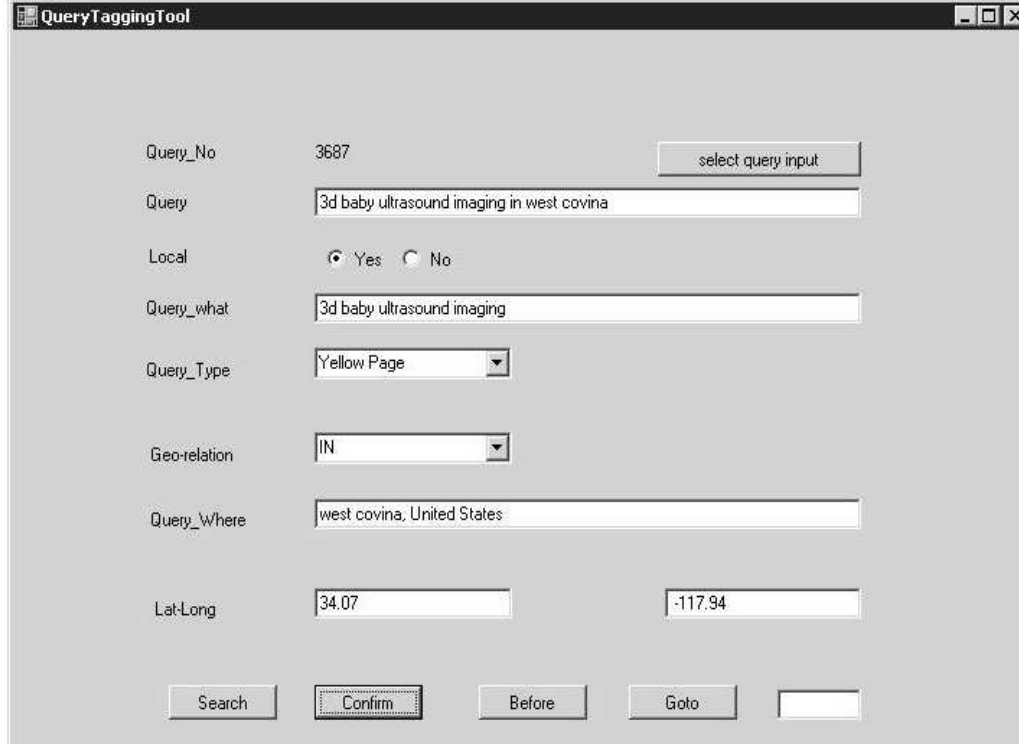


Figure 2. Interface of the Query Labeling tool

3.3.2 Label process

Two experts identify the six fields of these queries according to the task description including the <LOCAL>, <WHAT>, <WHERE>, <GEO-RELATION>, <WHAT-TYPE>, <LAT-LONG> of the location in the query. For the <LOCAL> field, we label it as “local” only if the query contains explicit locations. For the <WHAT> field, we keep all the terms in the query after extracting the <GEO-RELATION> and the <WHERE> fields. For example, the <WHAT> field of the query “ambassador suite hotel in Atlanta” is “ambassador suite hotel”. For <WHAT-TYPE> field, we define three types: Map type, Yellow page type, and Information type, which have been described above.

For the <WHERE> field, if the locations are ambiguous, we choose the location with the highest confidence score. The format is “location name + its upper location name”, e.g. “Atlanta, United States”. Meanwhile, we label the latitude and longitude of the location point. This value is only for reference here, because the lat-long values from different participants may vary greatly especially for the “big” locations, e.g. “Asia”, “Canada”.

3.4 Evaluation Method

To evaluate the performance of the query tagging task for the participants, we do the following three steps. First we pre-process the submissions of the participants to solve problems such as the format errors or data absence. Then we choose the subset from submissions with the same query number as in the evaluation set. Here we don’t use automatic checking since the format of the <WHERE> field is not unique. Three experts checked all the submissions independently and reach a final decision through discussion.

3.4.1 Criterion

We consider the following criteria in the evaluation process:

- 1) the <LOCAL> field should be the same as the answer;
- 2) the terms in the <WHAT> field should be the same as the answer;
- 3) the <WHAT-TYPE> and <GEO-RELATION> should be the same as the answer;

- 4) the <WHERE> field should contain the locations in the original query, no matter its upper location is correct or not;
- 5) we ignore the <LAT-LONG> field in this evaluation;

If one record in the submission meets the entire above criterions, it is correct, otherwise wrong.

3.4.2 Metric

We evaluate the submissions based on several evaluation metrics, including Precision, Recall, and F1-score. The participants do not know which queries will be used for evaluation. Here are the set of measures we use to evaluate results submitted by the participants:

$$Precision = \frac{Correct_tagged_query_num}{all_tagged_query_num}$$

$$Recall = \frac{Correct_tagged_query_num}{all_local_query_num}$$

$$F1_score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Figure 3. Precision, Recall and F1

4. Results

We only give the results for the local query. We can see that the Miracle team achieves the highest F1-score, 0.488, and the highest Recall too, 0.566, while the Ask team achieves the highest Precision, 0.625. Table 3 shows the results of all participants.

Table 3. Results of all Participants (only for the local query)

| Team | Precision | Recall | F1 |
|---------|--------------|--------------|--------------|
| Ask | 0.625 | 0.258 | 0.365 |
| Csusm | 0.201 | 0.197 | 0.199 |
| Linguit | 0.112 | 0.038 | 0.057 |
| Miracle | 0.428 | 0.566 | 0.488 |
| Talp | 0.222 | 0.249 | 0.235 |
| Xldb | 0.096 | 0.08 | 0.088 |

5. Discussions

In total, six teams participated in this query tagging task. We find several problems (technical) during the evaluations.

- 1) Fail to classify the local queries. Some local queries are classified as non-local by a few teams, so the recall for the local queries drops significantly.
- 2) The <WHAT> field is not complete. Some terms in the query are missing. For example, “apartments to rent in Cyprus”, the <WHAT> field should be “apartments to rent”, but some participants just output “apartments”. And “homer Alaska real estate”, <WHAT> field should be “homer real estate”, not “homer” or “real estate”.
- 3) Fail to classify the <WHAT-TYPE>. Especially for the “Yellow Page” and “Information”, a few of teams classify the “Yellow Page” queries as “Information”. Frankly speaking, sometimes it’s really hard to differentiate “Yellow Page” from “Information”, because of the ambiguity. For example, “Kansas state government”, if you want to know about the information about state government, it can be classified as “Information”, if you want to find the location, it can be classified as “Yellow Page”. Moreover, some teams don’t output the <WHAT-TYPE> for the local queries. Though their extraction precision for other fields is quite high, we have no choice but to label them as wrong cases.
- 4) Fail to identify <GEO-RELATION> field correctly. Most of the teams can recognize the geo-relation “IN”, but for the others, like “SOUTH_OF”, “SOUTH_WEST_OF”, “NORTH_WEST_OF”, few teams can identify them correctly. For example, “bank west of nevada”, the <GEO-RELATION> should be “WEST-OF”.

- 5) Fail to find the correct <WHERE>. A few of teams fail to extract the locations from the queries and label them as non-local queries. We guess the reason is that their gazetteer is not big enough. Moreover, some teams fail to disambiguate the locations and don't output the locations with the highest confidence scores.
- 6) Although we don't consider the <LAT-LONG> field this time, we find some participants don't output <LAT-LONG> at all. Maybe they don't have such information.

Most teams have employed a sophisticated gazetteer for location extraction, containing millions of geographical references. Their approaches for analyzing and classifying queries were mainly based on pre-defined rules. The system from the Miracle team, which achieved the best F1, was composed of three modules, namely geo-entity identifier, query analyzer and a two-level multi-classifier. Other systems followed similar designs. Generally speaking, the performance for most teams is not high. We list several possible reasons as follows:

- 1) New task. This query tagging task is totally new for the participants. And the time left is a bit short from the very beginning, just 2 months.
- 2) Critical standard. Our criterions to judge the results are quite critical. Some teams are good at the extraction but fail to identify the <WHAT-TYPE>. The overall performance is thus affected.
- 3) Queries are ambiguous. We find three kinds of ambiguity here.
 - a. Local/Non-Local Ambiguity: Some queries, like "airport", "space needle", are defined as non-local here because they don't contain explicit locations.
 - b. Yellow Page/Information Ambiguity: Due to the lack of background knowledge, it's hard to say some queries, like "Atlanta medical", are Yellow Page or Information. But we defined it as Yellow Page in this task.
 - c. Location Ambiguity: Some locations are ambiguous, like "Washington", "Columbus". We just choose the locations with the highest confidence based on our algorithm.
- 4) Culture understanding problem. When we were labeling the queries, we found that we lacked the necessary background of culture of western countries. In such situation, the labeling process may still contain some errors, even if we tried our best to avoid them.

6. Conclusions

In this report, we summarized the configuration and the results of the new query parsing task in GeoCLEF2007. The main purpose for organizing this task is to gather researchers who have similar interests. We first discussed the motivation of this task. Then we described the task design and evaluation methods. We also reported the evaluation results for all the participants.

This is the first time for us to organize such a task and we have learned a lot of lessons from it. In general, the task was conducted smoothly but it can be improved in many aspects, such as evaluation measure and data set preparation. We also plan to include more challenging parsing tasks and multilingual queries in the future.