TextMESS at GeoCLEF 2008: Result Merging with Fuzzy Borda Ranking

José M. Perea Ortega and L. Alfonso Ureña López
Dpto. de Informática,
Universidad de Jaén, Spain
{jmperea, laurena}@ujaen.es

Davide Buscaldi and Paolo Rosso
Natural Language Engineering Lab,
Dpto. de Sistemas Informáticos y Computación (DSIC),
Universidad Politécnica de Valencia, Spain
{dbuscaldi, prosso}@dsic.upv.es

Abstract

This paper describe the joint participation by the Universidad Politécnica de Valencia and the Universidad of Jaén to the GeoCLEF task. This activity has been carried out within the framework of the Spanish TextMESS project (Intelligent, Interactive and Multilingual Text Mining based on Human Language Technologies). The method employed for the participation is a result merging algorithm based on the fuzzy Borda voting scheme. This method takes as input the two document lists returned by the two systems developed by the participating groups and creates a document list where the documents are ranked according to the fuzzy Borda voting scheme. The results obtained are better than the individual systems, and also ones of the best ones of the task (second as group). However, the best result was obtained with a run which combined the baseline systems. The analysis of the results showed that the best runs were those in which only title and description were used, and unfortunately we chose to submit only a run of this type, with the base systems. The results confirm the effectiveness of the fuzzy Borda scheme for the combination of different systems.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; I.2 [**Artificial Intelligence**]: I.2.3 Uncertainty, "fuzzy," and probabilistic reasoning; I.2.7 Natural Language Processing

General Terms

Measurement, Performance, Experimentation

Keywords

Geographical Information Retrieval, Merging techniques, Fuzzy Borda

1 Introduction

In this paper we describe the joint participation of the groups of the Universidad Politécnica de Valencia and Universidad de Jaén to GeoCLEF 2008. This participation has been carried out within the framework of the Spanish project TextMESS (Spanish acronym for Intelligent, Interactive and Multilingual Text Mining based on Human Language Technologies).

We previously investigated various possibilities for the integration of our systems, focusing on two possible choices:

- Identify some features in the topics that allow to determine which system is going to obtain the best result over a determined topic (*system selection*);
- Combine the output of the different systems in a unique output (*output merging*).

We carried out some preliminary experiments with the GeoCLEF topics from 2005 to 2007 and the systems presented by the two groups in GeoCLEF 2007, in order to check whether the first option was feasible or not. These results proved that it was possible to use bag-of-words features to select the best system for a given topic. However, the experiments carried out with the new systems (those developed for GeoCLEF 2008) did not provide us with the same conclusion. Therefore, we chose to participate with an output merging algorithm based on the fuzzy Borda voting scheme [9, 6]. This method was previously used in the Word Sense Disambiguation task at Semeval[1] with good results [2]. Preliminary experiments with the data from 2005 to 2007 showed that it was possible to achieve an improvement of $\sim 2\%$ in Mean Average Precision (MAP) over the best system.

In Sections 2 and 3 we describe briefly the systems of each group (a more complete description can be found in the corresponding report of the CLEF Working Notes). In Section 4 we describe the fuzzy Borda ranking method, and finally we present the results and a brief discussion.

2 SINAI-GIR System Description

The SINAI-GIR system is made up of five main subsystems: *Translator*, *Collection Preprocessing subsystem*, *Query Analyzer*, *Information Retrieval subsystem* and *Validator*. Each translated query is preprocessed and analyzed by the *Query Analyzer*, identifying their geo-entities and spatial relations and making use of Geonames gazetteer[2]. This module also applies *query reformulation* based on the query parsing subtask [8], generating several independent queries which will be indexed and searched by means of the IR subsystem. On the other hand, the collection is preprocessed by the *Collection Preprocessing* module and finally the documents recovered by the IR subsystem are filtered and re-ranked by means of the *Validator* subsystem. Figure 1 shows the SINAI-GIR system architecture.

The main features of each subsystem are:

- **Translator**. We have used SINTRAM (SINai TRAnslation Module), our Machine Translation system which works with different online machine translators and implements several heuristics to combine different translations [5].
- **Collection Preprocessing Subsystem**. During the collection preprocessing, two indexes are generated (*locations* and *keywords* indexes). We apply the Porter *stemmer* [10], the Brill POS tagger [1] and a specific Named Entity Recognizer (NER) as LingPipe[3]. We also discard the English *stop-words*.
- **Query Analyzer**. It is responsible for preprocessing of English queries as well as the generation of different query reformulations.

[1] http://nlp.cs.swarthmore.edu/semeval
[2] http://www.geonames.org
[3] http://alias-i.com/lingpipe

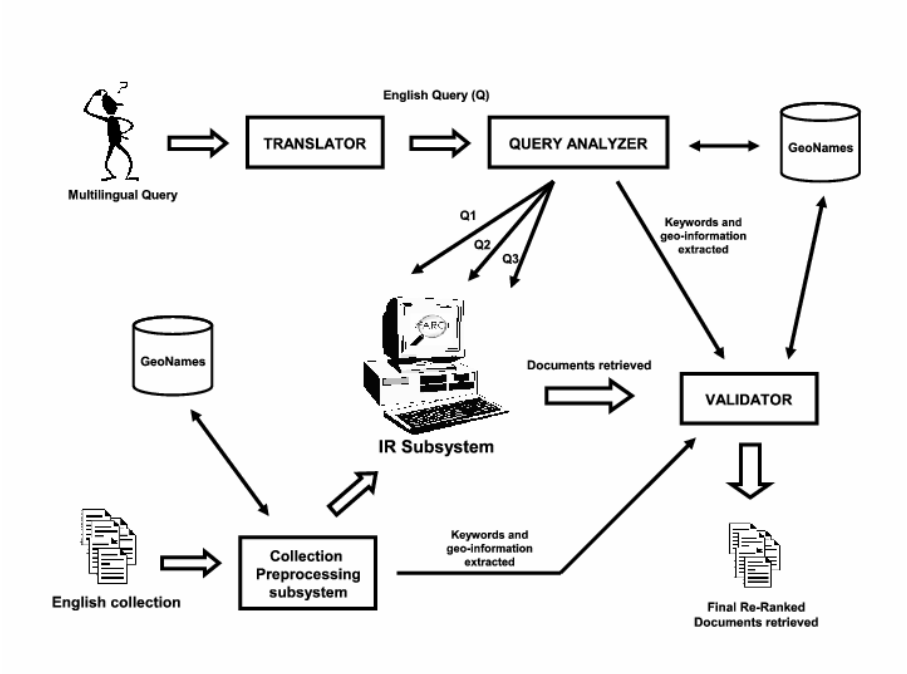

Figure 1: SINAI-GIR system architecture

- **Information Retrieval Subsystem**. As IR index-search engine we have used Lemur[4].
- ***Validator***. The aim of this subsystem is to filter the lists of documents recovered by the IR subsystem, establishing what of them are valid, depending on the locations and the *geo-relations* detected in the query. Another important function is to establish the final ranking of documents, based on manual rules and predefined weights.

3 The UPV GeoWorSE System

The system is built around the Lucene[5] open source search engine, version 2.1. The Stanford NER system based on Conditional Random Fields [4] is used for Named Entity Recognition and classification. The access to WordNet is provided by the MIT Java WordNet Interface [6]. The toponym disambiguator is based on the method presented in [3].

3.1 Indexing

During the indexing phase, the documents are examined in order to find location names (*toponym*) by means of the Stanford NER system. When a toponym is found, the disambiguator determines the correct reference for the toponym. Then, a modified lucene indexer adds to the *geo* index the toponym coordinates (retrieved from GeoWordNet); finally, it stores in the *wn* index the toponym together with its holonyms and synonyms. All document terms are stored in the *text* index. The indices are then used in the search phase, although the *geo* index is not used for search: it is used only to retrieve the coordinates of the toponyms in the document.

[4]http://www.lemurproject.org
[5]http://lucene.apache.org/
[6]http://www.mit.edu/~markaf/projects/wordnet/

3.2 Searching

The architecture of the search module is shown in Figure 2.

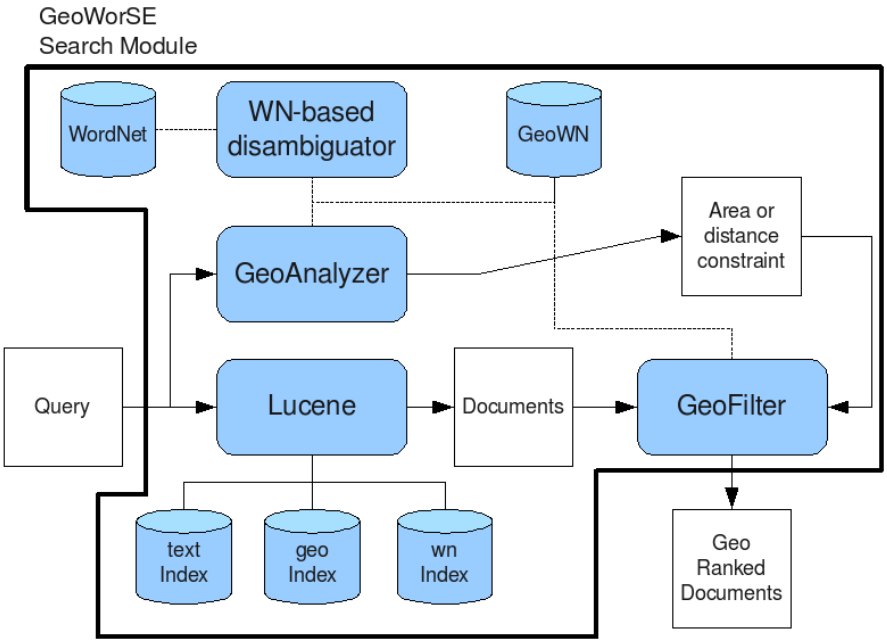

Figure 2: Diagram of the Search module

The topic text is searched by Lucene in the text index. The toponyms extracted by the Stanford NER are searched for in the *wn* index with a weight 0.25 with respect to the content terms. The result of the search is a list of documents ranked using the Lucene's weighting scheme. At the same time, the toponyms are passed to a module named *GeoAnalyzer* that creates a geographical constraint that is used to re-rank the document list. The GeoAnalyzer may return two types of geographical constraints:

- a *distance* constraint, corresponding to a point in the map: the documents that contain locations closer to this point will be ranked higher;
- an *area* constraint, correspoinding to a polygon in the map: the documents that contain locations included in the polygon will be ranked higher. The polygon is obtained by calculating the convex hull of the points associated to the toponyms using the Graham algorithm [7].

WordNet is used by the GeoAnalyzer module in order to extract the meronyms of the toponyms in the topic. These meronyms allow to improve the precision of the area constraint.

The objective of the GeoFilter module is to re-rank the documents retrieved by Lucene, according to geographical information. If the constraint extracted from the topic is a *distance* constraint, the weights of the documents are modified according to the following formula:

$$w(doc) = w_{Lucene}(doc) * (1 + \exp(-\min_{p \in P} d(q, p))) \quad (1)$$

Where w_{Lucene} is the weight returned by Lucene for the document doc, P is the set of points in the document, and q is the point extracted from the topic.

If the constraint extracted from the topic is an *area* constraint, the weights of the documents are modified according to formula 2:

$$w(doc) = w_{Lucene}(doc) * \left(1 + \frac{|P_q|}{|P|}\right) \tag{2}$$

where P_q is the set of points in the document that are contained in the area extracted from the topic.

4 Fuzzy Borda Merging

4.1 Fuzzy Borda count

In the classical (discrete) Borda count each expert gives a mark to each alternative, according to the number of alternatives worse than it. The fuzzy variant [9, 6] allows the experts to show numerically how much some alternatives are preferred to the others, evaluating their preference intensities from 0 to 1.

Let $R^1, R^2, \ldots, R^m$ be the fuzzy preference relations of m experts over n alternatives $x_1, x_2, \ldots, x_n$. Each expert k expresses its preferences by means of a matrix of preference intensities:

$$\begin{pmatrix} r_{11}^k & r_{12}^k & \ldots & r_{1n}^k \\ r_{21}^k & r_{22}^k & \ldots & r_{2n}^k \\ \ldots & \ldots & \ldots & \ldots \\ r_{n1}^k & r_{n2}^k & \ldots & r_{nn}^k \end{pmatrix}$$

where each $r_{ij}^k = \mu_{R^k}(x_i, x_j)$, with $\mu_{R^k} : X \times X \to [0, 1]$ is the membership function of R^k. The number $r_{ij}^k \in [0, 1]$ is considered as the degree of confidence with which the expert k prefers x_i to x_j. The final value assigned by the expert k to each alternative x_i is the sum by row of the entries greater than 0.5 in the preference matrix, or, formally:

$$r_k(x_i) = \sum_{j=1, r_{ij}^k > 0.5}^{n} r_{ij}^k \tag{3}$$

The threshold 0.5 ensure the relation $R^k$to be an ordinary preference relation [6].

The fuzzy Borda count for an alternative x_i is obtained as the sum of the values assigned by each expert:

$$\mathbf{r}(x_i) = \sum_{k=1}^{m} r_k(x_i) \tag{4}$$

4.2 Application of Fuzzy Borda count to Result Merging

In our approach each system is an expert: therefore, there are two preference matrices. The size of these matrices is variable: the reason is that the document list is not the same for the two systems. Therefore, the size of a preference matrix is $N_t \times N_t$, where N_t is the number of unique documents retrieved by the two systems (i.e. the number of documents that appear at least in one of the lists returned by the systems) for topic t.

The systems ranks the document with weights that are not in the same range. Therefore, the output weights $w_1, w_2, \ldots, w_n$ of each expert k are transformed to fuzzy confidence values by means of the following transformation:

$$r_{ij}^k = \frac{w_i}{w_i + w_j} \tag{5}$$

This transformation ensure that the preference values are in the range $[0, 1]$. In order to adapt the fuzzy Borda count to the merging of the results of IR systems, we had to determine the preference values in all the cases where one of the systems does not retrieve a document that has been retrieved by the other one. We decided to set the preference values of these documents to 0.5. This corresponds to the idea that the expert is presented an option on which it cannot express a preference.

5 Results

We submitted a total of 9 runs. In Table 1 we show the detail of each run in terms of the two systems combined and the topic fields used.

Table 1: Details of the setup of the TextMESS runs.

run ID	fields	UPV run ID	Jaen run ID
TMESS01	TDN	NLEL0802	EXP3
TMESS02	TDN	NLEL0802	EXP2
TMESS03	TDN	NLEL0802	EXP5
TMESS04	TDN	NLEL0803	EXP3
TMESS05	TDN	NLEL0803	EXP2
TMESS06	TDN	NLEL0803	EXP5
TMESS07A	TD	NLEL0804	EXP1
TMESS08	TDN	NLEL0505	EXP5
TMESS09	TDN	NLEL0807	EXP5

Table 2: Description of the runs of each system.

run ID	description
Valencia	
NLEL0802	base system (only text index, no wordnet, no map filtering)
NLEL0803	2007 system (no map filtering)
NLEL0804	base system, title and description only
NLEL0505	new system, all indices and map filtering enabled
NLEL0807	new system, text index and map filtering
Jaen	
EXP1	base system, title and description only
EXP2	base system, all fields
EXP3	"fusion" system (query reformulations)
EXP5	filtering system (rule-based)

In Table 3 we show the Mean Average Precision (MAP) obtained for each run, together with the MAP obtained by its composing runs.

The obtained results show that the use of the fuzzy Borda merging method always allows to improve the results of the best system. The improvement is greater if the two systems have a similar performance (see TMESS06) and the UPV system does not use map filtering. This behaviour is not observed when the UPV system uses map filtering. We suppose that a key feature for obtaining greater improvements by means of fuzzy Borda is that the systems share as few as characteristics as possible.

6 Conclusions and Further Work

We combined two different systems by means of the fuzzy Borda voting scheme. The implemented method allowed to improve the results of the combined systems, although the improvement was limited. We suppose that the best results with the fuzzy Borda merging can be obtained if the two systems share the same level of accuracy. Further work will be aimed to verify this hypothesis and to the integration of more than two systems.

Table 3: Results obtained for the submitted runs.

run ID	MAP	UPV run MAP	Jaen run MAP	system diff.	improvement
TMESS01	0.226	0.201	0.225	0.024	0.1%
TMESS02	0.227	0.201	0.226	0.025	0.1%
TMESS03	0.219	0.201	0.212	0.011	0.8%
TMESS04	0.234	0.216	0.225	0.009	0.9%
TMESS05	0.235	0.216	0.226	0.01	0.9%
TMESS06	0.226	0.216	0.212	0.004	1.4%
TMESS07A	0.286	0.224	0.284	0.06	0.2%
TMESS08	0.216	0.203	0.212	0.009	0.4%
TMESS09	0.213	0.202	0.212	0.01	0.1%

Acknowledgements

We would like to thank the TIMOM (TIN2006-15265-C06-03) and TIN2006-15265-C06-04 research projects for partially supporting this work.

References

[1] Eric Brill. A simple rule-based part-of-speech tagger. In *Proceedings of the third Conference on Applied Natural Language Processing (ANLP'92)*, pages 152–155, Trento, Italy, 1992.

[2] Davide Buscaldi and Paolo Rosso. Upv-wsd : Combining different wsd methods by means of fuzzy borda voting. In *Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 434–437. ACL, 2007.

[3] Davide Buscaldi and Paolo Rosso. A conceptual density-based approach for the disambiguation of toponyms. *International Journal of Geographical Information Systems*, 22(3):301–313, 2008.

[4] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370, U. of Michigan - Ann Arbor, 2005. ACL.

[5] Miguel A. García-Cumbreras, L. Alfonso Ureña López, Fernando Martínez Santiago, and José M. Perea Ortega. BRUJA System. The University of Jaén at the Spanish task of QA@CLEF 2006. In *Lecture Notes in Computer Science*, volume 4730 of LNCS Series, pages 328–338. Springer-Verlag, 2007.

[6] José Luis García Lapresta and Miguel Martínez Panero. Borda Count Versus Approval Voting: A Fuzzy Approach. *Public Choice*, 112(1-2):167–184, 2002.

[7] Ronald L. Graham. An efficient algorith for determining the convex hull of a finite planar set. *Information Processing Letters*, 1(4):132–133, 1972.

[8] Zhisheng Li, Chong Wanga, Xing Xie, and Wei-Ying Ma. Query Parsing Task for GeoCLEF 2007 Report. In *Proceedings of the Cross Language Evaluation Forum (CLEF 2007)*, 2007.

[9] Hannu Nurmi. Resolving Group Choice Paradoxes Using Probabilistic and Fuzzy Concepts. *Group Decision and Negotiation*, 10(2):177–199, 2001.

[10] M.F. Porter. An algorithm for suffix stripping. In *Program 14*, pages 130–137, 1980.