TALP at WePS-3 2010

Daniel Ferrés and Horacio Rodríguez

TALP Research Center, Software Department
Universitat Politècnica de Catalunya
Jordi Girona 1-3, 08043 Barcelona, Spain
{dferres, horacio}@lsi.upc.edu

Abstract. In this paper we present our system and experiments at the Third Web People Search Workshop (WePS-3) task for clustering web people search documents in English. In our experiments we used a simple approach with three algorithms: Lingo, Hierachical Agglomerative Clustering (HAC), and a 2-step HAC algorithm. We also present the results and initial conclusions in the context of the WePS-3 Task 1 for clustering. We obtained best results with HAC and 2-step HAC algorithms.

1 Introduction

The Third Web People Search (WePS-3) workshop is an evaluation Task under the scope of TebleCLEF. Its aim is to evaluate systems which cluster and extract information from web people searches in English. In this paper we present our system at WePS-3 Task 1 for clustering web people search documents in English. We also present the experiments with the WePS-3 development and test data, results and initial conclusions in the context of the WePS-3 Task 1.

1.1 Development and Test Data at WePS-3

The development data we used for WePS-3 is based on the test data of WePS-2 Clustering Task [1]. Test data for WePS-2 is composed of 30 ambiguous names: 10 name sets from the 1990 US Census, 10 from participants in ACL'08 and 10 from Wikipedia. Each name is made of two tokens, a first name and a last name. See more details of the WePS-2 data set in [1]. Around 100 documents have been downloaded from the top ranked search results.

The test data for WePS-3 was composed of 300 person names and 200 web documents for each name. As the WePS organizers did in WePS-2, some person names were obtained from the following sources: US Census (50), Wikipedia (50) and Computer Science Program Committee lists (50). In addition to that, the organizers provided names for which at least one person is an attorney (50), corporate executive (50) or realtor (50). For each name the top 200 web search results from Yahoo! were provided (URL, HTML pages, search snippets and ranking information).

2 System Description

The system architecture has two phases that are performed sequentially: HTML Cleaning and Clustering. The HTML cleaning phase consists in to convert HTML documents into plain text. We used the existing HTMLParser[1] (version 1.6) open-source software to perform this task. For Clustering phase we used several algorithms that are described below: Lingo, Hierarchical Agglomerative Clustering (HAC), and 2-steps HAC.

2.1 Lingo

Lingo is an algorithm that combines Phrase Discovery (detection of topics and phrases) and Latent Semantic Indexing to organize web search results in groups based on their content [2]. The approach of Lingo tries to seek short and clear labels with useful meanings that could cover most of the topics of the input text collection. Lingo gets phrases with semantic content to use them as labels in the clusters, then documents are assigned to the labels to create the groups. Lingo is implemented in the Carrot2 Project[2]. Carrot2 is an Open Source Clustering software that can group automatically small collections of documents or web search results in thematic categories.

Lingo uses the Vector Space Model and Singular Value Decomposition to find the labels of the clusters. It uses 3 methods of Natural Language Processing: Stemming, stop-words, and textual segmentation heuristics. Using stemming and stop-words according Lingo developers is important when we are working with small textual information and some noise (like working with snippets).

The most used parameters for the tuning of the Lingo algorithm are the Cluster assigment threshold and the Cluster candidate label threshold. The Cluster Assignment Threshold (tcA) controls the assignments of documents to the clusters. This threshold is based on the Cosine similarity between a label and a document and its common range is from 0.15 (default) to 0.3. The Cluster Candidate Label Threshold (tcL) controls the number of clusters (labels created). This threshold is based on the Cosine similarity between a candidate cluster label and the basis vectors of the SVD decomposition. This threshold default value is 0.775 and its common value range is from 0.70 to 0.90.

2.2 Hierarchical Agglomerative Clustering

The Hierarchical Agglomerative Clustering method used is agglomerative, it starts at the leaves and successively merges clusters together. HAC can be stopped by distance criterion and number of clusters criterion. The Lemur[3] Information Retrieval software includes an implementation of Hierarchial Agglomerative Clustering. The clustering algorithms implemented for Lemur and

[1] HTMLParser. `http://htmlparser.sourceforge.net/`
[2] Carrot2 Project. `http://project.carrot2.org`
[3] Lemur Project. `http://www.lemurproject.org`

used in this paper are described in [3]. These algorithms use cosine similarity in the vector space model as their metric. Stemming is used using the Porter algorithm. The HAC algorithm implemented in Lemur was used in WePS2 with good results [4]. The parameters accepted by Cluster are: 1) Type of cluster to use, either agglomerative or centroid (centroid is agglomerative using mean as a scoring method). 2) The scoring method to use for the agglomerative cluster over documents in a cluster maximum (max), minimum (min), average (avg), mean (mean). 3) The threshold, the minimum score for adding a document to an existing cluster.

2.3 2-step Clustering with Agglomerative Clustering

This is a two step algorithm that consists to cluster the results of an initial clustering process. The process follows these steps: 1) initial clustering with an agglomerative clustering algorithm that produces a set of clusters, 2) merging the content of each cluster in one new document by merging all the documents that pertain to a cluster into a one representative document for the whole cluster, 3) a second clustering step does agglomerative clustering (centroid or agglomerative configurations) over the collection of representative documents for the initial clusters.

3 Development experiments with WePS-3 trial data

For the WePS-3 trial evaluation we designed a set of several experiments that consist in applying different baseline configurations (see Table 1) to the WePS-3 trial data (WePS-2 test data).

The baseline runs were designed changing the parameters of the algorithms and the Clustering method. We did experiments with the three algorithms described before: Lingo, HAC, and 2-step HAC. We present here a set of these experiments. The experiments with Lingo share the same parameters (tcL=0.15, tcA=0.7), and differ in the kind of input to use as source documents. The following four experiments were done: full documents (1), snippets and title (2), context of the person name with 100 and 500 chars (3) (4). The experiments with agglomerative clustering differ with the type of cluster (agglomerative or centroid), type of scoring (minimum or maximum), and threshold. We did the following experiments: (5) aglomerative (agglo) with maximum score (max) and 0.07 as threshold, (6) centroid (cent) with minimum score (min) and 0.20 as threshold, (7) centroid (cent) with max and 0.05 as threshold. The experiments with 2-step clustering were in four types, i) centroid (first step) & centroid (second step) (8), ii) centroid (first step) & agglomerative (second step) (9) iii) agglomerative (first step) & centroid (second step) (10) and iv) agglomerative (first step) & agglomerative (second step): experiments from (11) to (15).

Table 1. Results with WePS-3 trial (development) data using B-Cubed measures

	Macro-averaged Scores			
	F-measures		B-Cubed	
Algorithm & Parameters	alfa=0,5	alfa=0,2	Prec.	Rec.
(10) Agglo(0.07;max)+Centroid(0.20;min)	**0,58**	0,63	0,55	0,67
(12) Agglo(0.07;max)+Agglo(0.15;max)	**0,58**	0,63	0,53	0,70
(11) Agglo (0.07;max)+Agglo(0.20;max)	**0,58**	0,62	0,55	0,66
(13) Agglo (0.07;max)+Agglo(0.10;max)	0,57	0,65	0,49	0,75
(14) Agglo (0.07;max)+Agglo (0.20;min)	0,57	0,60	0,56	0,63
(8) Centroid (0.20;min)+Cent (0.20;min)	0,56	0,67	0,47	0,80
(15) Agglo (0.07;max)+Agglo (0.07;max)	0,55	0,65	0,45	0,79
(7) Centroid (0.05;max)	0,54	0,58	0,54	0,62
(5) Agglo (0.07;max)	0,54	0,55	0,58	0,56
(6) Centroid (0.20;min)	0,53	0,52	0,61	0,52
(9) Centroid/0.07;min)+Agglo(0.03;min)	0,52	0,57	0,49	0,62
(baseline) ALL_IN_ONE	0,53	0,66	0,43	1,00
(baseline) CHEAT_SYS	0,52	0,65	0,43	1,00
(1) Lingo (Full document)	0,45	0,54	0,39	0,64
(2) Lingo (Snippets + Title)	0,43	0,44	0,47	0,46
(3) Lingo (context 500 chars)	0,42	0,42	0,51	0,43
(4) Lingo (context 100 chars)	0,43	0,42	0,53	0,42
(baseline) ONE_IN_ONE	0,34	0,27	1,00	0,24

4 Test experiments with WePS-3 test data

For the WePS-3 evaluation with the test data we designed a set of five experiments that consist in applying different baseline configurations to the development set data(see Table 2). The first run (TALP_1) uses agglomerative clustering and the second run (TALP_2) uses a 2-step clustering approach with both Agglomerative clustering algorithms of Lemur. The first step does agglomerative clustering and the second step does again agglomerative clustering with the output of the first step. The third run (TALP_3) applies the algorithm Lingo for clustering. The fourth run (TALP_4) uses the centroid algorithm from the Lemur. The fifth run (TALP_5) used a 2 step clustering, the first step applies the centroid algorithm of lemur and the second step applies agglomerative clustering.

Table 2. Results with the WePS-3 Test data Task evaluated with BCubed mesures.

run	Algorithm	Parameters.	avgPrec.	avgRec.	avgF-m.(0,5)
TALP_1	Agglo	(t=0.10;max)	0.56	0.41	0.42
TALP_2	Agglo + Agglo	(t=0.10;max)	0.38	0.70	0.43
TALP_3	Lingo	(tcl=0.15;tca=0.7)	0.40	0.49	0.39
TALP_4	Centroid	(t=0.10;max)	0.60	0.41	0.43
TALP_5	**Centroid + Agglo**	(t=0.10;max)	0.40	0.66	**0.44**

5 Conclusions

This is our first attempt to deal with Web Person Search Clustering at WePS clustering task. We have used three clustering algorithms (Lingo, HAC, and 2-step HAC) to perform the task of clustering web people search in the context of the WePS-3 Task-1. In the preprocessing of documents we detected that HTML filtering is a crucial step to avoid noise. It is convenient to avoid noise from input documents to achieve better results, specially in the Lingo algorithm. Input Noise as broken sentences and random strings could have afected the results of the clustering algorithms, specially Lingo and its cluster labels. We achieved best results with the 2-step HAC and Agglomerative Clustering which deliver better performance than Lingo. We used limited NLP processing only in with lingo, the other runs used Porter Stemmer before indexing and clustering. Further improvements include the use of NLP techniques for Part-of-Speech Tagging, Named Entity Recognition and Classification, and Information Extraction.

Acknowledgments

This work has been supported by the Spanish Research Dept. (KNOW 2, TIN2009-14715-C04-04). Daniel Ferrés is supported by the EBW II Project, which is financed by the European Commission within the framework of the Erasmus Mundus Programme. TALP Research Center is recognized as a Quality Research Group (2001 SGR 00254) by DURSI, the Research Department of the Catalan Government.

References

1. Artiles, J., Gonzalo, J., Sekine, S.: WePS 2 Evaluation Campaign: Overview of the Web People Search Clustering Task. In: 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference. (2009)
2. Osinski, S., Weiss, D.: A Concept-Driven Algorithm for Clustering Search Results. IEEE Intelligent Systems **20**(3) (2005) 48–54
3. Steinbach, M., Karypis, G., Kumar, V.: A Comparison of Document Clustering Techniques. In Grobelnik, M., Mladenic, D., Milic-Frayling, N., eds.: KDD-2000 Workshop on Text Mining, August 20, Boston, MA (2000) 109–111
4. Balog, K., He, J., Hofmann, K., Jijkoun, V., Monz, C., Tsagkias, M., Weerkamp, W., de Rijke, M.: The University of Amsterdam at WePS2. In: 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference. (2009)