EVALUATING THE COREF-PRO CROSS-DOCUMENT COREFERENCE SYSTEM

Nghiem Tri Duc
University tutor: Bernardo Magnini
Company tutor: Roberto Zanoli

ABSTRACT

Cross-document coreference (CDC) plays an important role in text processing. It allows users to extract information about a particular entity from different documents. I evaluated a system for cross-document coreference developed in FBK that, differently for other systems needing a fixed threshold to group names referring to the same entity, tries first to guess the correct number of entities by evaluating the clusters quality, and then to co-refer person names to these entities. The evaluation phase included testing the system on an Italian dataset (Live Memories corpus) and an English one (Web People Search corpus).

1. INTRODUCTION

Nowadays, the development of web technology generates huge data and information in various areas. Particularly, the existence of many community-like web pages such as Facebook, Yahoo 360 (now replaced by Yahoo Profile), Wordpress, Twitter, My Space etc, created giant spaces for users all over the world sharing their mind, their private information. This phenomenon caused the growth in the number of queries related to the person names in some recent years (in order to make friend or find information about specific person). Therefore, the requirement of personal information extraction is increasing significantly. According to a study by Altavista [3], around 17% of the queries contained personal names. Hence, the task of disambiguating personal names across documents (i.e. it is cross-document coreference) plays an important role. Cross-document coreference occurs when the same person, place, event or concept is mentioned in more than one document.

CDC is different to other tasks in text processing:

- CDC is different to name entity recognition (NER), the task that just recognizes proper names (e.g. names of person, organizations, locations, etc.);
- CDC is different to the within document coreference because of the more ambiguity and the complicated phenomenon that do not occur inside one document;
- CDC is also different to entity linking that refers entity mentions in a document to their representation in a knowledge base.

The goal of this internship project is evaluating the CDC system developed in FBK on two different datasets:

- Live Memories benchmark: people in Italian news (L′Adige);
- Web People Search (WePS) dataset: people in English web pages.

To resolve the CDC problem, I executed some different algorithms on the different features sets. Finally, these algorithms were evaluated to find the best configurations of the system.

In this report I present an overview about the cross-document coreference and the evaluation tasks including: the CDC evaluation at WePS-2009 (section 2), the architecture of the COREF-PRO system (section 3), the evaluation metrics (section 4), the experiments and results (section 5).

2. CDC EVALUATION AT WEPS-2009

WePS collected data on web pages using the web search engine API provided by Yahoo. The data is divided in development and test data. The development set consists of 47 ambiguous names. The number of clusters per name has a large variability (from 1 to 91 different people sharing the same name). The test data includes 30 names collected from three different sources (10 names for each source): Wikipedia, ACL08 (Association for Computational Linguistics), and US Census.

Many groups used this dataset to run their experiments and compare the results of the CDC task. In order to simplify the preprocessing task for all users, the datasets were provided in HTML format.

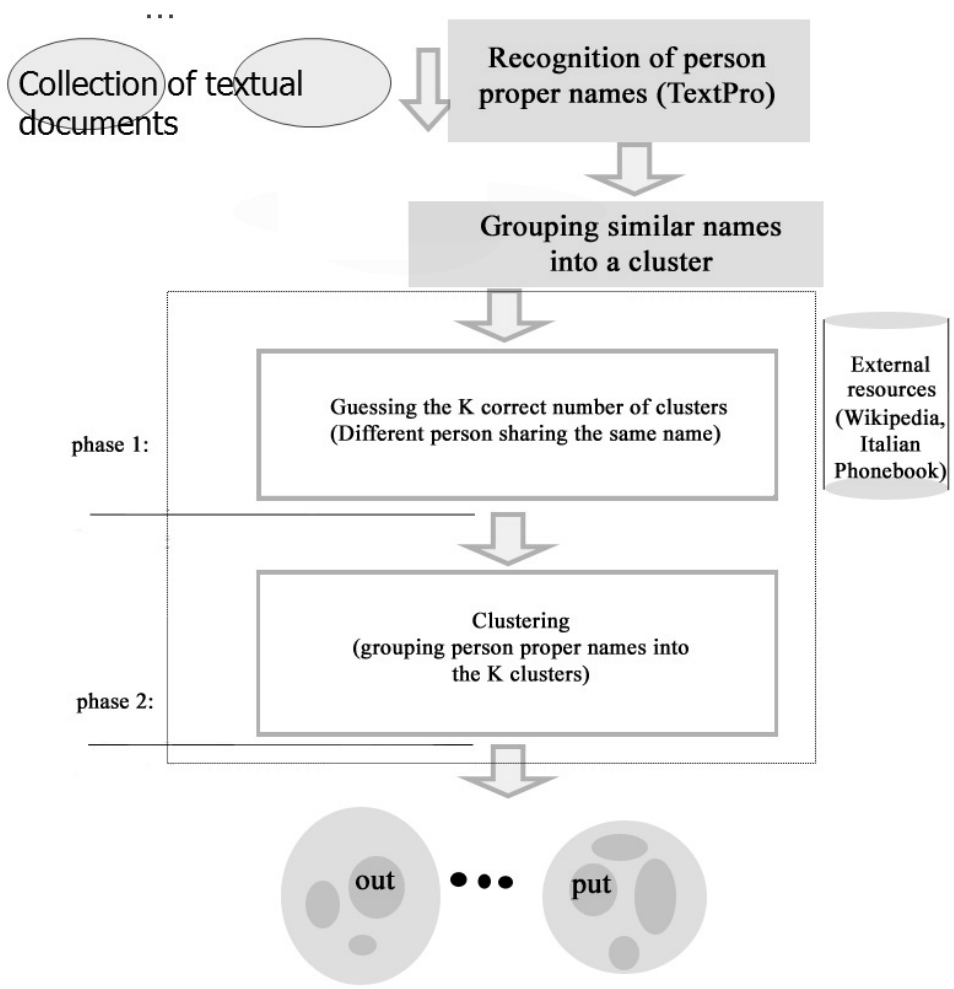

Figure 1. FBK System's architecture.

3. ARCHITECTURE OF THE COREF-PRO SYSTEM

Figure 1 shows the architecture of the COREF-PRO system developed in FBK. The texts from the input sources are pre-processed by different tools. TextPro, as can be seen in the figure 1, extracts the names of entities occurred in the input sources. The pre-processing task then provides some particular features. Each object to be clustered is represented by five features: document topic, the topic assigned automatically (by another system), the key concepts (key information in the document), the name entities co-occurred in document, and the person roles (figure 2). Entity mentions sharing very similar names are, then, assigned to a cluster, and new clusters are created as necessary. These clusters are processed in two phases which are parts of the cross-document coreference task. My project concentrated on the evaluation of the tasks:

- Phase 1: Guessing the correct number of clusters.
  The goal of this phase is to estimate how many clusters should be divided for the data of each name. In other words, it determines how many different people sharing the same name exist in the corpus. This phase is called cluster stopping [5] because it limits the number of clusters that the second phase has to group into. Cluster stopping could use the external resources collected from Wikipedia and Italian Phonebook. These sources provided the number of people sharing the same names for each name in the corpus. They limit the number of clusters for each name to be estimated by this phase.

- Phase 2: Clustering.
  This phase groups objects into the K clusters, where K is estimated by the previous phase. All objects in the same cluster refer to one entity, i.e. the documents

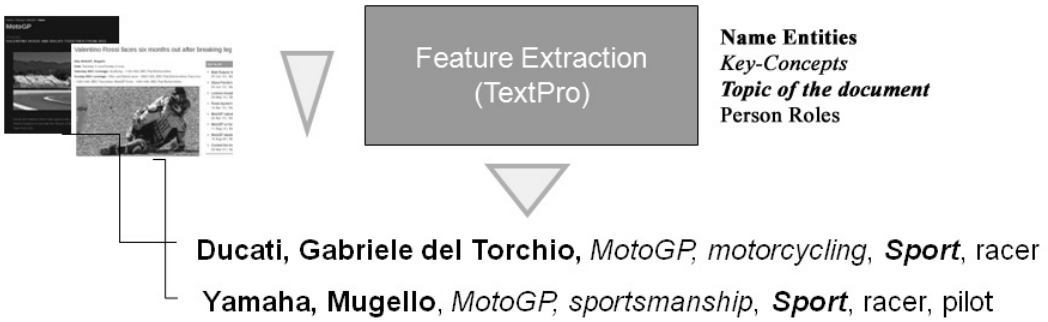

Figure 2. Pre-processing task extracted different features

mentioned the same person.

3.1. Cluster stopping

Firstly, the cluster stopping phase uses clustering to produce *n* cluster results (the cluster result contains 1 cluster, 2 clusters *n* clusters), where *n* is the threshold we set for each name. It can be the number of objects sharing that name (this is the maximum value of *n*) or other value achieved by using external information. The cluster stopping measure determines which result is the best and return the result.

The idea of geometric structure is used for measuring the structure of clustering result. It defines that a good clustering result should subject to: the distance among objects in a cluster is minimized or/and the distance of clusters is maximized. The functions representing these problems are criterion functions. The functions representing the intra-cluster optimization are called internal criterion functions. The functions called external criterion functions represent the inter-cluster optimization problem. The hybrid criterion functions are the combination of both two ways. Figure 3 provides the geometric view of criterion functions.

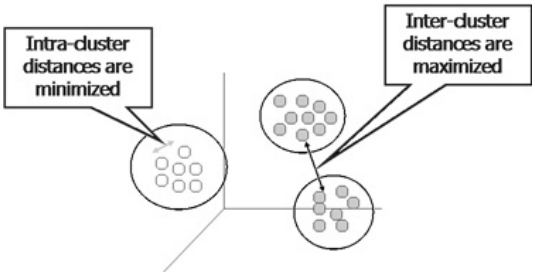

Figure 3. The geometric view of criterion functions.

The cluster stopping measure methods used in this project were implemented by Ted Pedersen and Anagha Kulkarni (2006) [5]. They provided four methods: PK1, PK2, PK3 and Gap statistic. The first three methods find the values of criterion functions as K increases (K is the number of clusters gained in each clustering result, so the value of K is in the interval between 1 and *n* as mentioned in the first step) then try to figure out which value of K makes the criterion function stop increasing significantly. In other words, these cluster stopping measures try to find the knee-point of the criterion function; if it exists, it will be the answer. Figure 4 is an example for the idea of these cluster stopping measures.

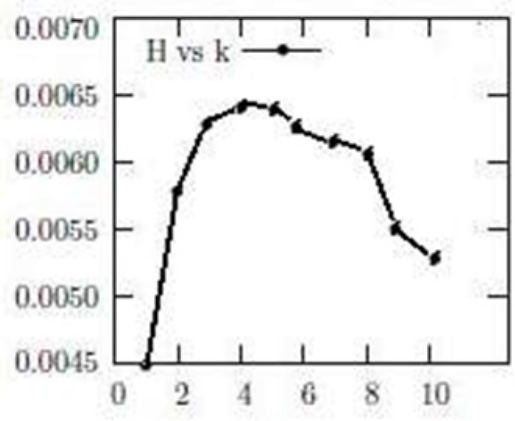

Figure 4. Example of the Hybrid criterion function. In this example, 4 is the best number of clusters.

Gap Statistic [7], however, works in different way. It does not attempt to identify the knee-point of criterion functions. Rather, it generates the sample of reference data that represents the observed data as if it had no meaningful cluster in it and simply made up of noise [5]. After that, the criterion function of the reference data is compared to that of observed data. The value of k is decided when the correlation point in observed data is least like noise (in other words, the biggest Gap between noise data and the observed data). Figure 5 shows the example of Gap statistic.

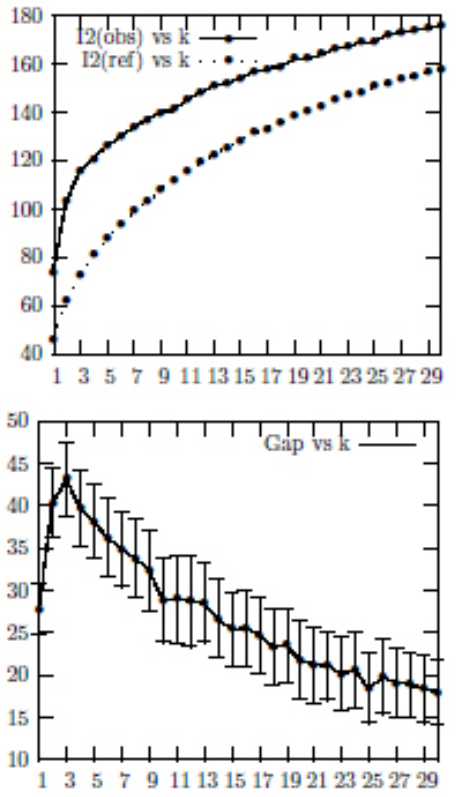

Figure 5. Example of Internal criterion function for the obsevered data and noise data (top) and the Gap between them (bottom). The predicted number of clusters in this case is 3 [5]

3.2. Clustering algorithms

This phase separates the data into *K* clusters, where *K* was given by the cluster stopping step. There are two types of clustering methods using in this project: partition and hierarchy methods [4].

The partition approach attempts to separate the data into smaller subsets subject to the optimization of the criterion functions (locally or globally optimizing). Some of them repeat the dividing until achieving *K* clusters (repeated bisection approach, abbr. rb) whereas the others separate data into *K* clusters by using the initial *K* means ($m1$, $m2$, mK), and then re-calculate these *K* means (each mean is the mean of each cluster) to re-assign objects to the *K* clusters. The repetition is done until the centroids (means) of clusters are stable (K-means).

The hierarchy method, for example agglomerative clustering, firstly assigns objects into numbers of clusters (the number of clusters can be the number of the objects or another number depending on the specific algorithm). Then these clusters are agglomerated (grouped) gradually based on the optimization of the criterion function. The agglomeration loops until having *K* clusters.

4. Evaluation metrics

4.1. Evaluate the cluster stopping phase

Comparing the output of this phase with the correct number of clusters provided by the gold standard is the method being used to evaluate the results. The deviation for the acceptable result was set to 10%. The success method is the one that has the highest acceptable values over the whole data set.

4.2. B-cubed metric

We used the B-Cubed [1], which is the only one that satisfies four intuitive formal constraints on evaluation metrics for the clustering problem [3]. The original B-cubed definition is extended to handle overlapping clustering [3]. For the object (element) i in the data, it has the precision and recall:

$$precision_i = \frac{\text{\#correct elements in output cluster containing i}}{\text{\#elements in the output cluster containing i}}$$

$$recall_i = \frac{\text{\#correct elements in output cluster containing i}}{\text{\# elements in the truth cluster containing i}}$$

$$\text{final precision} = \frac{1}{N} \times \sum_{i=0}^{N} precision_i$$

$$\text{final recall} = \frac{1}{N} \times \sum_{i=0}^{N} recall_i$$

$$\text{Fmean} = \frac{2 \times P \times R}{P + R}$$

According to this definition, we have the results of the example given in the figure 6.
For the entity 6: P = 2/7, R = 2/2
Considering all the entities:
P = (5*5/5 + 2*2/7+ 5*5/7)/12 = 0.76
R = (5*5/5+2*2/2+5*5/5)/12 = 1
Fmean = 2*0.76*1 / (1+0.76)= 0.86

4.3. Baseline

The baseline used in this project is All In One (AIO) baseline. It assigns all objects into a single cluster. Obviously, the effect of this baseline is that the recall is always one. It works very well with the data collected from the news, because most of the news mentions the famous people.

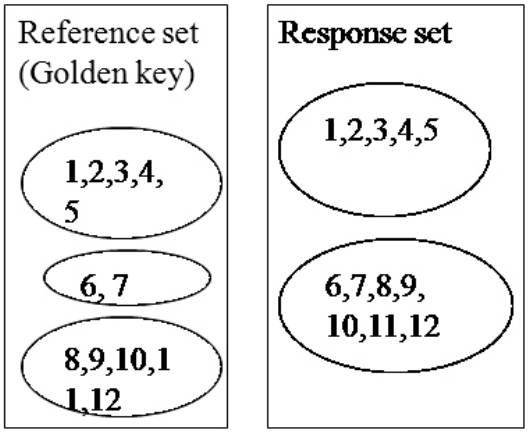

Figure 6. Example to calculate the B-cubed metric.

5. EXPERIMENTS AND RESULTS

5.1. Data

The data set of the Live Memories project was selected from Italian Adige-500K corpus [2]. It contains 209 seed names. Besides, these names are separated into the different categories according to the fame and ambiguity of each name (as can be seen in table 1).

	Not famous	Quite famous regional	Quite famous national	Very famous regional	Very famous national
Very Ambiguos	Paolo Rossi	Elena Marino	Paolo Rossi		Paolo Rossi
Ambiguos	Franco Marini	Vittorio Colombo	Giovanna Marini		Franco Marini
Not Ambiguos	Bruno Kessler	Dante Clauser	Marta Russo	Bruno Kessler	Umberto Eco

Table 1. Example of Live Memories data according with different dimensions.

The fame categories were divided in regional and national level because L′adige′s contents included both a regional and a national section. Besides, the ambiguity categories were evaluated using the number of subscribers in the external source: Pagine Bianche-the Italian telephone dictionary [2].

We used the development set of the Live Memories corpus. It includes 105 ambiguous names belong to 348 entities in 22580 documents. Table 2 shows the distribution of names in different categories mentioned above.

	Not famous	Quite famous regional	Quite famous national	Very famous regional	Very famous national
Very Ambiguos	11	4	4	0	4
Ambiguos	6	2	4	0	2
Not Ambiguos	13	18	14	10	18

Table 2. The distribution of names in the Live Memories development set.

5.2. Experiments

The experiments were designed to evaluate the two phases of the system, in order to find the best configuration for each phase. Besides, the experiments on the two data sets (Live Memories development data set and the test set of WePS-2 system) were run to compare the performances on the two systems.

I did the experiments on the first phase with different configurations (algorithms, criterion functions and cluster stopping measures) and input parameter (the upper bound of number of clusters). I set the input parameter with different values: the number of objects that share the same name (for each name in the development set), the number gained by using the external resources that mentioned in section 3. The simple strategy to achieve the mixed information of these resources can be seen in the table 3 below. With each name, if it appears in wikis list, I use the number provided by Wikipedia. Otherwise, if it appears in Italian Phonebook (IP), the number is taken from this source. If it does not belong to any set, 1 will be returned (high probably it is a rare name).

	WiKi	Not in Wiki
IP	Wi-K	IP-K
Not in IP	Wi-K	1

Table 3. Strategy to consult mixed sources.

In order to find the best clustering algorithm and the criterion functions, I ran phase 2 with the correct number of clusters (counting on the golden key) and with the value

given by phase 1. The best configurations of this phase then were set to run on the WePS-2 data in order to compare two systems. These experiments were performed by Cluto [4], a free application that implemented various clustering algorithms and criterion functions.

Finally, some experiments were executed to determine which feature plays the important role. Each experiment used the different features set that generated by removing one feature from the original set.

5.3. Results

Table 4 below shows the result of the first phases experiments. The best configuration is gained by using the mixed external information and the repeated bisection clustering method running with the internal criterion function, and gap statistic measure.

	Not famous	Quite famous regional	Quite famous national	Very famous regional	Very famous national
All In One	13/30	11/24	12/22	10/10	20/24
Best conf.	15/30	11/24	13/22	10/10	20/24

Table 4. The result of guessing the number of clusters

The first value in each cell is the number of time the system returned the acceptable value. The second value is the number of items (names) in each category. As can be seen from the table, the AIO performed very well. It means that many names in the development data set are not shared by many people (i.e. they belongs to one person). The best performance of COREF-PRO is slightly better than the AIO.

Topic	P	R	Fmean
Agglo + i	0.93	0.82	0.86
All In One	0.85	1.0	0.9
Best conf.	0.96	0.93	0.94

Topic	P	R	Fmean
gap +rb + i	0.86	0.97	0.89
All In One	0.85	1.0	0.9
Best conf.	0.86	1.0	0.9

Table 5. Result of experiment on phase 2 when parsing the correct number of clusters (top) and the number of clusters given by the first phase (bottom).

The best configuration for the second phase is using UPGMA [6] algorithm.UPGMA (Unweighted Pair Group Method with Arithmetic Mean) is a simple agglomerative algorithm. The algorithm examines the structure present in a pair wise distance matrix (or a similarity matrix) to construct a rooted tree. At each step, the nearest two clusters are combined into a higher-level cluster. The distance between any two clusters A and B is taken to be the average of all distances between pairs of objects "x" in A and "y" in B, that is, the mean distance among elements of two clusters. In the top table (table 5), the performance of this configuration is significantly better than the others. This experiment was done by using the correct number of clusters; hence the gap is clear. But in the second table, the gaps between them are small. It is because of the accumulation of errors of both two phases. The AIO baseline performed really well on this data set because of many ambiguous names, and the phenomenon of many documents mentioning one name i.e. famous name (more than 22000 documents mentioned just around 350 entities). It is shown more clearly in table 7.

Although using the correct number of clusters (table 5 top), the best configuration made mistakes. It shows that CDC is really a difficult task. For example, with the name: Paolo Rossi, suppose that we have some people sharing this name: one is football player, one is comedian. If the football player attended in theatre, his name could be mentioned as the comedian in articles. Therefore, the system grouped him in the comedian cluster, i.e. the mistake was made.

The system performed on this data set better than on the WePS-2 test set (table 6). Table 6 presents some results of two datasets. The first two rows are the result on Live Memories development set. The others are the results provided by different participants performances on WePS-2. That test set just contains only 30 names and the average ambiguity of those names is higher than the Live Memories Development set. The other reason is that the feature used on this test set. It contains most of the words collected in web pages. That is why it increased the difficulty for the system to measure the similarity among objects. Besides, the experiments on the WePS-2 used the best configuration achieved by experiments on the Live Memories development set. Obviously, the AIO baseline just worked well with the famous names or the non-ambiguous names, because in these cases the data set tends to mention one single person. As can be seen from the table 7, our best configuration had better performances than the baseline on the very ambiguous and not famous categories.

The last group of experiments attempt to find most important feature. After removing each feature in each experiment, the results of these experiments are similar. It shows that each feature plays a similar important role in the system. The same situation occurred in WePS-2, the NER feature did not play the important role in the web people search

	Not famous			Quite famous regional			Quite famous national			Very famous regional			Very famous national		
	P	R	F-mean	P	R	F-mean	P	R	F-mean	P	R	F-mean	P	R	F-mean
Not ambiguos	0.81	1	0.88	0.92	1	0.95	0.93	1	0.95	0.99	1	1	0.99	1	0.99
	0.86	0.66	0.73	0.95	0.85	0.89	0.96	0.94	0.95	0.99	0.96	0.97	0.99	0.97	0.98
	0.82	1	0.88	0.92	1	0.95	0.93	1	0.95	0.99	1	1	0.99	1	0.99
Ambiguos	0.72	1	0.82	0.68	1	0.76	0.68	1	0.76				1	1	1
	0.85	0.69	0.75	0.89	0.73	0.79	0.86	0.65	0.72	empty	empty	empty	1	1	1
	0.76	1	0.85	0.68	0.98	0.75	0.7	0.98	0.78				1	1	1
Very ambiguos	0.72	1	0.82	0.68	1	0.76	0.68	1	0.76				1	1	1
	0.85	0.69	0.75	0.89	0.73	0.79	0.86	0.65	0.72	empty	empty	empty	1	1	1
	0.76	1	0.85	0.68	0.98	0.75	0.7	0.98	0.78				1	1	1

Table 7. Result of the whole system on each category of names. The first line of each row is the performance of All In One Baseline. The second line is the performance of the best clustering with the correct number of clusters, the last one is the performance of the whole system (combination of phase 1 and phase 2).

	P	R	Fmean
Live Memories dev set: correct #clusters	0.96	0.93	0.94
Live Memories dev set: two phases	0.86	1.0	0.9
BEST-HAC-TOKENS	0.89	0.83	0.85
BEST-HAC-BIGRAMS	0.91	0.81	0.85
polyUHK	0.87	0.79	0.82
UVA 1	0.85	0.80	0.81
ITC-UT 1	0.93	0.73	0.81
XMEDIA 3	0.82	0.66	0.72
WePS-2 test set: correct #clusters	0.73	0.81	0.76
UCI 2	0.66	0.84	0.71
LANZHOU 1	0.80	0.66	0.70
FICO 3	0.85	0.62	0.70
UMD 4	0.94	0.60	0.70
HAC-BIGRAMS	0.95	0.55	0.67
UGUELPH 1	0.54	0.93	0.63
CASIANED 4	0.65	0.75	0.63
HACK TOKENS	0.95	0.48	0.59
AUG 4	0.73	0.58	0.57
WePS-2 test set: two phases	0.44	1.0	0.53
All In One	0.43	1	0.53
One In One	1	0.24	0.34
BUAP 1	0.89	0.25	0.33

Table 6. Results on two datasets.

data after all [3].

6. CONCLUSIONS

One of the most important motivations of this project is to find the best algorithms, methods on the Live Memories corpus. As the results of the evaluation tasks, the best configurations on the Live Memories development corpus were provided: internal criterion function, and gap statistic measure for the cluster stopping phase; UPGMA algorithm for the clustering phase. These configurations then can be used to annotate the whole corpus of the Live Memories project.

7. ACKNOWLEDGEMENTS

A special thanks to Roberto Zanoli, and Bernardo Magnini, my tutors, who helped, trained and supported me so much during the internship in FBK. Thanks to Alberto Lavelli for the suggestions that made my work better. Thanks to Chritian Girardi for the corpus and your supported works to me. This work has been partially supported by the Live Memories project (www.livememories.org), funded by the Autonomous Province of Trento (Italy) under the call Major Projects 2006.

References

[1] Amit Bagga, Breck Baldwin. Entity-Based Cross-Document corefencing Using the Vector Space Model.

In *Proceedings of the 17th international conference on Computational linguistics*, volume 1, pages 79–85, 1998.

[2] Bentivogli L., Girardi C. and Pianta E. Creating a Gold Standard for Person Cross-Document Coreference Resolution in Italian News. In *Proceedings of LREC 2008 Workshop on Resources and Evaluation for Identity Matching, Entity Resolution and Entity Management*, Marrakech, Morocco, 2008.

[3] Javier Artiles, Julio Gonzalo and Satoshi Sekine. WePS 2 Evaluation Campaign: Overview of the Web People Search Clustering Task. In *2nd Web People Search Evaluation Workshop (WePS 2009)*, 2009.

[4] George Karypis. Cluto a Clustering Toolkit. Technical report 2003, available at `http://glaros.dtc.umn.edu/gkhome/views/cluto`.

[5] Ted Pedersen and Anagha Kulkarni. Automatic Cluster Stopping with Criterion Functions and the Gap Statistic. In *Proceedings of the Demonstration Session of the Human Language Technology Conference and the Sixth Annual Meeting of the North American Chapter of the Association for Computational Linguistic*, New York City, NY., 2006.

[6] R. Sokal and C. Michener. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, pages 1409–1438, 1958.

[7] Robert Tibshirani, Guenther Walther and Trevor Hastie. Estimating the number of clusters in a data set via the Gap Statistic. *Journal of the Royal Statistics Society (Series B)*, pages 411–423, 2001.