Clique-Based Clustering for improving Named Entity Recognition systems

Julien Ah-Pine
Xerox Research Centre Europe
6, chemin de Maupertuis
38240 Meylan, France
julien.ah-pine@xrce.xerox.com

Guillaume Jacquet
Xerox Research Centre Europe
6, chemin de Maupertuis
38240 Meylan, France
guillaume.jacquet@xrce.xerox.com

Abstract

We propose a system which builds, in a semi-supervised manner, a resource that aims at helping a NER system to annotate corpus-specific named entities. This system is based on a distributional approach which uses syntactic dependencies for measuring similarities between named entities. The specificity of the presented method however, is to combine a clique-based approach and a clustering technique that amounts to a soft clustering method. Our experiments show that the resource constructed by using this clique-based clustering system allows to improve different NER systems.

1 Introduction

In Information Extraction domain, named entities (NEs) are one of the most important textual units as they express an important part of the meaning of a document. Named entity recognition (NER) is not a new domain (see MUC[1] and ACE[2] conferences) but some new needs appeared concerning NEs processing. For instance the NE *Oxford* illustrates the different ambiguity types that are interesting to address:

- intra-annotation ambiguity: Wikipedia lists more than 25 cities named *Oxford* in the world
- systematic inter-annotation ambiguity: the name of cities could be used to refer to the university of this city or the football club of this city. This is the case for *Oxford* or *Newcastle*
- non-systematic inter-annotation ambiguity: *Oxford* is also a company unlike *Newcastle*.

The main goal of our system is to act in a complementary way with an existing NER system, in order to enhance its results. We address two kinds of issues: first, we want to detect and correctly annotate corpus-specific NEs[3] that the NER system could have missed; second, we want to correct some wrong annotations provided by the existing NER system due to ambiguity. In section 3, we give some examples of such corrections.

The paper is organized as follows. We present, in section 2, the global architecture of our system and from §2.1 to §2.6, we give details about each of its steps. In section 3, we present the evaluation of our approach when it is combined with other classic NER systems. We show that the resulting hybrid systems perform better with respect to F-measure. In the best case, the latter increased by 4.84 points. Furthermore, we give examples of successful correction of NEs annotation thanks to our approach. Then, in section 4, we discuss about related works. Finally we sum up the main points of this paper in section 5.

2 Description of the system

Given a corpus, the main objectives of our system are: to *detect* potential NEs; to *compute the possible annotations* for each NE and then; to *annotate* each occurrence of these NEs with the *right annotation* by analyzing its local context.

We assume that this corpus dependent approach allows an easier NE annotation. Indeed, even if a NE such as *Oxford* can have many annotation types, it will certainly have less annotation possibilities in a specific corpus.

Figure 1 presents the global architecture of our system. The most important part concerns steps 3 (§2.3) and 4 (§2.4). The aim of these subprocesses is to group NEs which have the same annotation with respect to a given context. On the one hand, clique-based methods (see §2.3 for

[1] http://www-nlpir.nist.gov/related_projects/muc/

[2] http://www.nist.gov/speech/tests/ace

[3] In our definition a corpus-specific NE is the one which does not appear in a classic NEs lexicon. Recent news articles for instance, are often constituted of NEs that are not in a classic NEs lexicon.

Proceedings of the 12th Conference of the European Chapter of the ACL, pages 51–59,
Athens, Greece, 30 March – 3 April 2009. ©2009 Association for Computational Linguistics

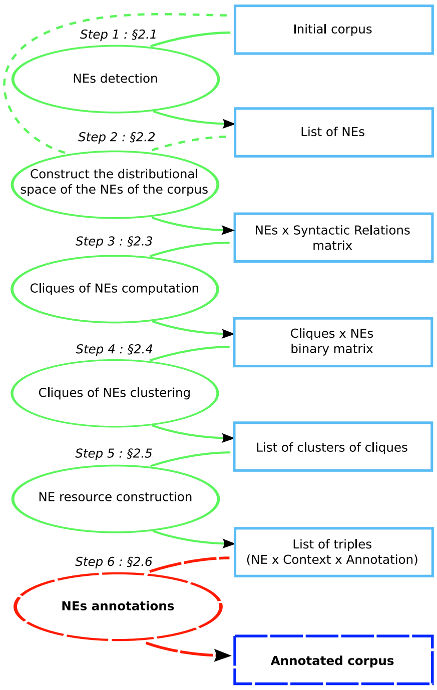

Initial corpus
Step 1 : §2.1
NEs detection
List of NEs
Step 2 : §2.2
Construct the distributional space of the NEs of the corpus
NEs x Syntactic Relations matrix
Step 3 : §2.3
Cliques of NEs computation
Cliques x NEs binary matrix
Step 4 : §2.4
Cliques of NEs clustering
List of clusters of cliques
Step 5 : §2.5
NE resource construction
List of triples (NE x Context x Annotation)
Step 6 : §2.6
NEs annotations
Annotated corpus

Figure 1: General description of our system

details on cliques) are interesting as they allow the same NE to be in different cliques. In other words, cliques allow to represent the different possible annotations of a NE. The clique-based approach drawback however, is the over production of cliques which corresponds to an artificial over production of possible annotations for a NE. On the other hand, clustering methods aim at structuring a data set and such techniques can be seen as data compression processes. However, a simple NEs hard clustering doesn't allow a NE to be in several clusters and thus to express its different annotations. Then, our proposal is to combine both methods in a clique-based clustering framework. This combination leads to a soft-clustering approach that we denote CBC system. The following paragraphs, from 2.1 to 2.6, describe the respective steps mentioned in Figure 1.

2.1 Detection of potential Named Entities

Different methods exist for detecting potential NEs. In our system, we used some lexico-syntactic constraints to extract expressions from a corpus because it allows to detect some corpus-specific NEs. In our approach, a potential NE is a noun starting with an upper-case letter or a noun phrase which is (see (Ehrmann and Jacquet, 2007) for similar use):

- a governor argument of an attribute syntactic relation with a noun as governee argument (e.g. *president* $\xrightarrow{attribute}$ *George Bush*)
- a governee argument of a modifier syntactic relation with a noun as a governor argument (e.g. *company* $\xleftarrow{modifier}$ *Coca-Cola*).

The list of potential NEs extracted from the corpus will be denoted $\mathbb{NE}$ and the number of NEs $|\mathbb{NE}|$.

2.2 Distributional space of NEs

The distributional approach aims at evaluating a distance between words based on their syntactic distribution. This method assumes that words which appear in the same contexts are semantically similar (Harris, 1951).

To construct the *distributional space* associated to a corpus, we use a robust parser (in our experiments, we used XIP parser (Aït et al., 2002)) to extract chunks (i.e. nouns, noun phrases, . . .) and syntactic dependencies between these chunks. Given this parser's output, we identify triple instances. Each triple has the form $w_1.R.w_2$ where w_1 and w_2 are chunks and R is a syntactic relation (Lin, 1998), (Kilgarriff et al., 2004).

One triple gives two contexts ($1.w_1.R$ and $2.w_2.R$) and two chunks (w_1 and w_2). Then, we only select chunks w which belong to $\mathbb{NE}$. Each point in the distributional space is a NE and each dimension is a syntactic context. $\mathbb{CT}$ denotes the set of all syntactic contexts and $|\mathbb{CT}|$ represents its cardinal.

We illustrate this construction on the sentence *"provide Albania with food aid"*. We obtain the three following triples (note that *aid* and *food aid* are considered as two different chunks):

provide_VERB•I-OBJ•Albania_NOUN
provide_VERB•PREP_WITH•aid_NOUN
provide_VERB•PREP_WITH•food aid_NP

From these triples, we have the following chunks and contexts[4]:

Chunks:	Contexts:
provide_VERB	**1.provide_VERB.I-OBJ**
Albania_NOUN	1.provide_VERB.PREP_WITH
aid_NOUN	2.Albania_NOUN.I-OBJ
food aid_NP	2.aid_NOUN.PREP_WITH
	2.food aid_NP.PREP_WITH

According to the NEs detection method described previously, we only keep the chunks and contexts which are in **bold** in the above table.

[4] In the context 1.VERB:provide.I-OBJ, the figure 1 means that the verb *provide* is the governor argument of the Indirect OBJect relation.

We also use an heuristic in order to reduce the over production of chunks and contexts: in our experiments for example, each NE and each context should appear more than 10 times in the corpus for being considered.

D is the resulting ($|\mathbb{NE}| \times |\mathbb{CT}|$) NE-Context matrix where $e_i : i = 1, \ldots, |\mathbb{NE}|$ is a NE and $c_j : j = 1, \ldots, |\mathbb{CT}|$ is a syntactic context. Then we have:

$$D(e_i, c_j) = \text{Nb. of occ. of } c_j \text{ associated to } e_i \quad (1)$$

2.3 Cliques of NEs computation

A clique in a graph is a set of pairwise adjacent nodes which is equivalent to a complete subgraph. A maximal clique is a clique that is not a subset of any other clique. Maximal cliques computation was already employed for semantic space representation (Ploux and Victorri, 1998). In this work, cliques of lexical units are used to represent a precise meaning. Similarly, we compute cliques of NEs in order to represent a precise annotation.

For example, *Oxford* is an ambiguous NE but a clique such as <*Cambridge, Oxford, Edinburgh University, Edinburgh, Oxford University*> allows to focus on the specific annotation <organization> (see (Ehrmann and Jacquet, 2007) for similar use).

Given the distributional space described in the previous paragraph, we use a probabilistic framework for computing similarities between NEs. The approach that we propose is inspired from the language modeling framework introduced in the information retrieval field (see for example (Lavrenko and Croft, 2003)). Then, we construct cliques of NEs based on these similarities.

2.3.1 Similarity measures between NEs

We first compute the maximum likelihood estimation for a NE e_i to be associated with a context c_j: $P_{ml}(c_j|e_i) = \frac{D(e_i,c_j)}{|e_i|}$, where $|e_i| = \sum_{j=1}^{|\mathbb{CT}|} D(e_i, c_j)$ is the total occurrences of the NE e_i in the corpus.

This leads to sparse data which is not suitable for measuring similarities. In order to counter this problem, we use the Jelinek-Mercer smoothing method: $D'(e_i, c_j) = \lambda P_{ml}(c_j|e_i) + (1 - \lambda)P_{ml}(c_j|\mathbb{CORP})$ where $\mathbb{CORP}$ is the corpus and $P_{ml}(c_j|\mathbb{CORP}) = \frac{\sum_i D(e_i,c_j)}{\sum_{i,j} D(e_i,c_j)}$. In our experiments we took $\lambda = 0.5$.

Given D', we then use the cross-entropy as a similarity measure between NEs. Let us denote by s this similarity matrix, we have:

$$s(e_i, e'_i) = -\sum_{c_j \in \mathbb{CT}} D'(e_i, c_j) \log(D'(e_{i'}, c_j)) \quad (2)$$

2.3.2 From similarity matrix to adjacency matrix

Next, we convert s into an adjacency matrix denoted $\hat{s}$. In a first step, we binarize s as follows. Let us denote $\{e^i_1, \ldots, e^i_{|\mathbb{NE}|}\}$, the list of NEs ranked according to the descending order of their similarity with e_i. Then, $L(e_i)$ is the list of NEs which are considered as the nearest neighbors of e_i according to the following definition:

$$L(e_i) = \quad (3)$$
$$\{e^i_1, ..., e^i_p : \frac{\sum_{i'=1}^{p} s(e_i, e^i_{i'})}{\sum_{i'=1}^{|\mathbb{NE}|} s(e_i, e_{i'})} \leq a; p \leq b\}$$

where $a \in [0, 1]$ and $b \in \{1, \ldots, |\mathbb{NE}|\}$. $L(e_i)$ gathers the most significant nearest neighbors of e_i by choosing the ones which bring the a most relevant similarities providing that the neighborhood's size doesn't exceed b. This approach can be seen as a flexible k-nearest neighbor method. In our experiments we chose $a = 20\%$ and $b = 10$.

Finally, we symmetrize the similarity matrix as follows and we obtain $\hat{s}$:

$$\hat{s}(e_i, e_{i'}) = \begin{cases} 1 & \text{if } e_{i'} \in L(e_i) \text{ or } e_i \in L(e_{i'}) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

2.3.3 Cliques computation

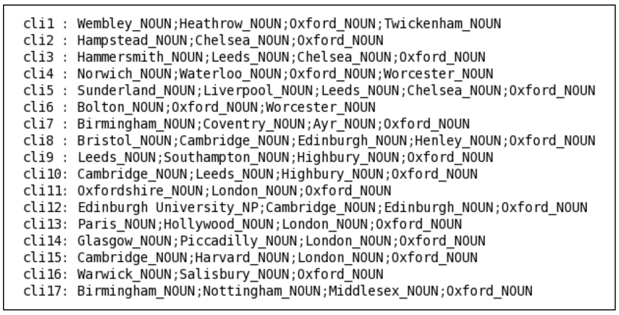
Given $\hat{s}$, the adjacency matrix between NEs, we compute the set of maximal cliques of NEs denoted $\mathbb{CLI}$. Then, we construct the matrix T of general term:

$$T(cli_k, e_i) = \begin{cases} 1 & \text{if } e_i \in cli_k \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where cli_k is an element of $\mathbb{CLI}$. T will be the input matrix for the clustering method.

In the following, we also use cli_k for denoting the vector represented by $(T(cli_k, e_1), \ldots, T(cli_k, e_{|\mathbb{NE}|}))$.

Figure 2 shows some cliques which contain *Oxford* that we can obtain with this method. This figure also illustrates the over production of cliques since at least cli8, cli10 and cli12 can be annotated as <organization>.



```
cli1 : Wembley_NOUN;Heathrow_NOUN;Oxford_NOUN;Twickenham_NOUN
cli2 : Hampstead_NOUN;Chelsea_NOUN;Oxford_NOUN
cli3 : Hammersmith_NOUN;Leeds_NOUN;Chelsea_NOUN;Oxford_NOUN
cli4 : Norwich_NOUN;Waterloo_NOUN;Oxford_NOUN;Worcester_NOUN
cli5 : Sunderland_NOUN;Liverpool_NOUN;Leeds_NOUN;Chelsea_NOUN;Oxford_NOUN
cli6 : Bolton_NOUN;Oxford_NOUN;Worcester_NOUN
cli7 : Birmingham_NOUN;Coventry_NOUN;Ayr_NOUN;Oxford_NOUN
cli8 : Bristol_NOUN;Cambridge_NOUN;Edinburgh_NOUN;Henley_NOUN;Oxford_NOUN
cli9 : Leeds_NOUN;Southampton_NOUN;Highbury_NOUN;Oxford_NOUN
cli10: Cambridge_NOUN;Leeds_NOUN;Highbury_NOUN;Oxford_NOUN
cli11: Oxfordshire_NOUN;London_NOUN;Oxford_NOUN
cli12: Edinburgh University_NP;Cambridge_NOUN;Edinburgh_NOUN;Oxford_NOUN
cli13: Paris_NOUN;Hollywood_NOUN;London_NOUN;Oxford_NOUN
cli14: Glasgow_NOUN;Piccadilly_NOUN;London_NOUN;Oxford_NOUN
cli15: Cambridge_NOUN;Harvard_NOUN;London_NOUN;Oxford_NOUN
cli16: Warwick_NOUN;Salisbury_NOUN;Oxford_NOUN
cli17: Birmingham_NOUN;Nottingham_NOUN;Middlesex_NOUN;Oxford_NOUN
```

Figure 2: Examples of cliques containing *Oxford*

2.4 Cliques clustering

We use a clustering technique in order to group cliques of NEs which are mutually highly similar. The clusters of cliques which contain a NE allow to find the different possible annotations of this NE.

This clustering technique must be able to construct "pure" clusters in order to have precise annotations. In that case, it is desirable to avoid fixing the number of clusters. That's the reason why we propose to use the Relational Analysis approach described below.

2.4.1 The Relational Analysis approach

We propose to apply the *Relational Analysis approach* (RA) which is a clustering model that doesn't require to fix the number of clusters (Michaud and Marcotorchino, 1980), (Bédécarrax and Warnesson, 1989). This approach takes as input a similarity matrix. In our context, since we want to cluster cliques of NEs, the corresponding similarity matrix S between cliques is given by the *dot products* matrix taken from T: $S = T \cdot T'$. The general term of this similarity matrix is: $S(cli_k, cli_{k'}) = S_{kk'} = \langle cli_k, cli_{k'} \rangle$. Then, we want to maximize the following clustering function:

$$\Delta(S,X) = \qquad (6)$$
$$\sum_{k,k'=1}^{|\mathbb{CLI}|} \underbrace{\left(S_{kk'} - \frac{\sum_{(k'',k''')\in\mathbb{S}^+} S_{k''k'''}}{|\mathbb{S}^+|} \right)}_{cont_{kk'}} X_{kk'}$$

where $\mathbb{S}^+ = \{(cli_k, cli_{k'}) : S_{kk'} > 0\}$.

In other words, cli_k and $cli_{k'}$ have more chances to be in the same cluster providing that their similarity measure, $S_{kk'}$, is greater or equal to the mean average of positive similarities.

X is the solution we are looking for. It is a binary relational matrix with general term: $X_{kk'} = 1$, if cli_k is in the same cluster as $cli_{k'}$; and $X_{kk'} = 0$, otherwise. X represents an equivalence relation. Thus, it must respect the following properties:

- binarity: $X_{kk'} \in \{0,1\}; \forall k, k'$,
- reflexivity: $X_{kk} = 1; \forall k$,
- symmetry: $X_{kk'} - X_{k'k} = 0; \forall k, k'$,
- transitivity: $X_{kk'} + X_{k'k''} - X_{kk''} \leq 1; \forall k, k', k''$.

As the objective function is linear with respect to X and as the constraints that X must respect are linear equations, we can solve the clustering problem using an integer linear programming solver. However, this problem is NP-hard. As a result, in practice, we use heuristics for dealing with large data sets.

2.4.2 The Relational Analysis heuristic

The presented heuristic is quite similar to another algorithm described in (Hartigan, 1975) known as the "leader" algorithm. But unlike this last approach which is based upon euclidean distances and inertial criteria, the RA heuristic aims at maximizing the criterion given in (6). A sketch of this heuristic is given in Algorithm 1, (see (Marcotorchino and Michaud, 1981) for further details).

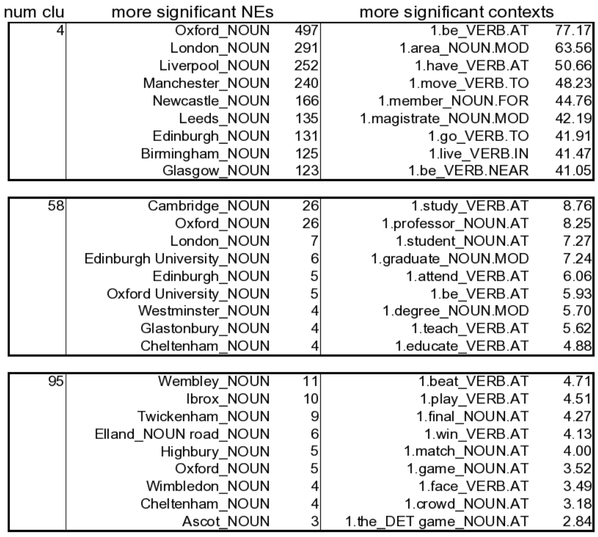
Algorithm 1 RA heuristic

```
Require: $nbitr$ = number of iterations; $\kappa_{\max}$ = maximal number of clusters; $S$ the similarity matrix
  $m \leftarrow \frac{\sum_{(k,k')\in\mathbb{S}^+} S_{kk'}}{|\mathbb{S}^+|}$
  Take the first clique $cli_k$ as the first element of the first cluster
  $\kappa = 1$ where $\kappa$ is the current number of cluster
  for $q = 1$ to $nbitr$ do
    for $k = 1$ to $|\mathbb{CLI}|$ do
      for $l = 1$ to $\kappa$ do
        Compute the contribution of clique $cli_k$ with cluster $clu_l$: $cont_l = \sum_{cli_{k'} \in clu_l} (S_{kk'} - m)$
      end for
      $clu_{l^*}$ is the cluster id which has the highest contribution with clique $cli_k$ and $cont_{l^*}$ is the corresponding contribution value
      if $(cont_{l^*} < (S_{kk} - m)) \wedge (\kappa < \kappa_{\max})$ then
        Create a new cluster where clique $cli_k$ is the first element and $\kappa \leftarrow \kappa + 1$
      else
        Assign clique $cli_k$ to cluster $clu_{l^*}$
        if the cluster where was taken $cli_k$ before its new assignment, is empty then
          $\kappa \leftarrow \kappa - 1$
        end if
      end if
    end for
  end for
```

We have to provide a number of iterations or/and a delta threshold in order to have an approximate solution in a reasonable processing time. Besides, it is also required a maximum number of clusters but since we don't want to fix this parameter, we put by default $\kappa_{\max} = |\mathbb{CLI}|$.

Basically, this heuristic has a $O(nbitr \times \kappa_{\max} \times |\mathbb{CLI}|)$ computation cost. In general terms, we can assume that $nbitr << |\mathbb{CLI}|$, but not $\kappa_{\max} << |\mathbb{CLI}|$. Thus, in the worst case, the algorithm has a $O(\kappa_{\max} \times |\mathbb{CLI}|)$ computation cost.

Figure 3 gives some examples of clusters of cliques[5] obtained using the RA approach.



num clu	more significant NEs		more significant contexts	
4	Oxford_NOUN	497	1.be_VERB.AT	77.17
	London_NOUN	291	1.area_NOUN.MOD	63.56
	Liverpool_NOUN	252	1.have_VERB.AT	50.66
	Manchester_NOUN	240	1.move_VERB.TO	48.23
	Newcastle_NOUN	166	1.member_NOUN.FOR	44.76
	Leeds_NOUN	135	1.magistrate_NOUN.MOD	42.19
	Edinburgh_NOUN	131	1.go_VERB.TO	41.91
	Birmingham_NOUN	125	1.live_VERB.IN	41.47
	Glasgow_NOUN	123	1.be_VERB.NEAR	41.05
58	Cambridge_NOUN	26	1.study_VERB.AT	8.76
	Oxford_NOUN	26	1.professor_NOUN.AT	8.25
	London_NOUN	7	1.student_NOUN.AT	7.27
	Edinburgh University_NOUN	6	1.graduate_NOUN.MOD	7.24
	Edinburgh_NOUN	5	1.attend_VERB.AT	6.06
	Oxford University_NOUN	5	1.be_VERB.AT	5.93
	Westminster_NOUN	4	1.degree_NOUN.MOD	5.70
	Glastonbury_NOUN	4	1.teach_VERB.AT	5.62
	Cheltenham_NOUN	4	1.educate_VERB.AT	4.88
95	Wembley_NOUN	11	1.beat_VERB.AT	4.71
	Ibrox_NOUN	10	1.play_VERB.AT	4.51
	Twickenham_NOUN	9	1.final_NOUN.AT	4.27
	Elland_NOUN road_NOUN	6	1.win_VERB.AT	4.13
	Highbury_NOUN	5	1.match_NOUN.AT	4.00
	Oxford_NOUN	5	1.game_NOUN.AT	3.52
	Wimbledon_NOUN	4	1.face_VERB.AT	3.49
	Cheltenham_NOUN	4	1.crowd_NOUN.AT	3.18
	Ascot_NOUN	3	1.the_DET game_NOUN.AT	2.84

Figure 3: Examples of clusters of cliques (only the NEs are represented) and their associated contexts

2.5 NE resource construction using the CBC system's outputs

Now, we want to exploit the clusters of cliques in order to annotate NE occurrences. Then, we need to construct a NE resource where for each pair (NE x syntactic context) we have an annotation. To this end, we need first, to assign a cluster to each pair (NE x syntactic context) (§2.5.1) and second, to assign each cluster an annotation (§2.5.2).

2.5.1 Cluster assignment to each pair (NE x syntactic context)

For each cluster clu_l we provide a score $F_c(c_j, clu_l)$ for each context c_j and a score $F_e(e_i, clu_l)$ for each NE e_i. These scores[6] are given by:

$$F_c(c_j, clu_l) = \sum_{e_i \in clu_l} \frac{D(e_i, c_j)}{\sum_{i=1}^{|\mathbb{NE}|} D(e_i, c_j)} \sum_{e_i \in clu_l} \mathbf{1}_{\{D(e_i,c_j) \neq 0\}} \tag{7}$$

where $\mathbf{1}_{\{P\}}$ equals 1 if P is true and 0 otherwise.

$$F_e(e_i, clu_l) = \#(clu_l, e_i) \tag{8}$$

Given a NE e_i and a syntactic context c_j, we now introduce the contextual cluster assignment matrix $A_{ctxt}(e_i, c_j)$ as follows: $A_{ctxt}(e_i, c_j) = clu^*$ where: $clu^* = \text{Argmax}_{\{clu_l : clu_l \ni e_i; F_e(e_i, clu_l) > 1\}} F_c(c_j, clu_l)$.

In other words, clu^* is the cluster for which we find more than one occurrence of e_i and the highest score related to the context c_j.

Furthermore, we compute a default cluster assignment matrix A_{def}, which does not depend on the local context: $A_{def}(e_i) = clu^\bullet$ where: $clu^\bullet = \text{Argmax}_{\{clu_l : clu_l \ni \{cli_k : cli_k \ni e_i\}\}} |cli_k|$.

In other words, $clu^\bullet$ is the cluster containing the biggest clique cli_k containing e_i.

2.5.2 Clusters annotation

So far, the different steps that we have introduced were unsupervised. In this paragraph, our aim is to give a correct annotation to each cluster (hence, to all NEs in this cluster). To this end, we need some annotation seeds and we propose two different semi-supervised approaches (regarding the classification given in (Nadeau and Sekine, 2007)). The first one is the manual annotation of some clusters. The second one proposes an automatic cluster annotation and assumes that we have some NEs that are already annotated.

Manual annotation of clusters This method is fastidious but it is the best way to match the corpus data with a specific guidelines for annotating NEs. It also allows to identify new types of annotation. We used the ACE2007 guidelines for manually annotating each cluster. However, our CBC system leads to a high number of clusters of cliques and we can't annotate each of them. Fortunately, it also leads to a distribution of the clusters' size (number of cliques by cluster) which is

[5] We only represent the NEs and their frequency in the cluster which corresponds to the number of cliques which contain the NEs. Furthermore, we represent the most relevant contexts for this cluster according to equation (7) introduced in the following.

[6] For data fusion tasks in information retrieval field, the scoring method in equation (7) is denoted CombMNZ (Fox and Shaw, 1994). Other scoring approaches can be used see for example (Cucchiarelli and Velardi, 2001).

similar to a Zipf distribution. Consequently, in our experiments, if we annotate the 100 biggest clusters, we annotate around eighty percent of the detected NEs (see §3).

Automatic annotation of clusters We suppose in this context that many NEs in $\mathbb{NE}$ are already annotated. Thus, under this assumption, we have in each cluster provided by the CBC system, both annotated and non-annotated NEs. Our goal is to exploit the available annotations for *refining the annotation* of a cluster by implicitly taking into account the syntactic contexts and for *propagating the available annotations* to NEs which have no annotation.

Given a cluster clu_l of cliques, $\#(clu_l, e_i)$ is the weight of the NE e_i in this cluster: it is the number of cliques in clu_l that contain e_i. For all annotations a_p in the set of all possible annotations $\mathbb{AN}$, we compute its associated score in cluster clu_l: it is the sum of the weights of NEs in clu_l that is annotated a_p.

Then, if the maximal annotation score is greater than a simple majority (half) of the total votes[7], we assign the corresponding annotation to the cluster. We precise that the annotation <none>[8] is processed in the same way as any other annotations. Thus, a cluster can be globally annotated <none>. The limit of this automatic approach is that it doesn't allow to annotate new NE types than the ones already available.

In the following, we will denote by $\mathbf{A_{clu}}(clu_l)$ the annotation of the cluster clu_l.

The cluster annotation matrix $\mathbf{A_{clu}}$ associated to the contextual cluster assignment matrix A_{ctxt} and the default cluster assignment matrix A_{def} introduced previously will be called the CBC system's NE resource (or shortly the NE resource).

2.6 NEs annotation processes using the NE resource

In this paragraph, we describe how, given the CBC system's NE resource, we annotate occurrences of NEs in the studied corpus with respect to its local context. We precise that for an occurrence of a NE e_i its associated local context is the set of syntactical dependencies c_j in which e_i is involved.

2.6.1 NEs annotation process for the CBC system

Given a NE occurrence and its local context we can use $A_{ctxt}(e_i, c_j)$ and $A_{def}(e_i)$ in order to get the default annotation $\mathbf{A_{clu}}(A_{def}(e_i))$ and the list of contextual annotations $\{\mathbf{A_{clu}}(A_{ctxt}(e_i, c_j))\}_j$.

Then for annotating this NE occurrence using our NE resource, we apply the following rules:

- if the list of contextual annotations $\{\mathbf{A_{clu}}(A_{ctxt}(e_i, c_j))\}_j$ is conflictual, we annotate the NE occurrence as <none>,
- if the list of contextual annotations is non-conflictual, then we use the corresponding annotation to annotate the NE occurrence
- if the list of contextual annotations is empty, we use the default annotation $\mathbf{A_{clu}}(A_{def}(e_i))$.

The NE resource plus the annotation process described in this paragraph lead to a NER system based on the CBC system. This NER system will be called CBC-NER system and it will be tested in our experiments both alone and as a complementary resource.

2.6.2 NEs annotation process for an hybrid system

We place ourselves into an hybrid situation where we have two NER systems (NER 1 + NER 2) which provide two different lists of annotated NEs. We want to combine these two systems when annotating NEs occurrences.

Therefore, we resolve any conflicts by applying the following rules:

- If the same NE occurrence has two different annotations from the two systems then there are two cases. If one of the two system is CBC-NER system then we take its annotation; otherwise we take the annotation provided by the NER system which gave the best precision.
- If a NE occurrence is included in another one we only keep the biggest one and its annotation. For example, if *Jacques Chirac* is annotated <person> by one system and *Chirac* by <person> by the other system, then we only keep the first annotation.
- If two NE occurrences are contiguous and have the same annotation, we merge the two NEs in one NE occurrence.

3 Experiments

The system described in this paper rather target corpus-specific NE annotation. Therefore, our ex-

[7] The total votes number is given by $\sum_{e_i \in clu_l} \#(clu_l, e_i)$.

[8] The NEs which don't have any annotation.

periments will deal with a corpus of recent news articles (see (Shinyama and Sekine, 2004) for motivations regarding our corpus choice) rather than well-known annotated corpora. Our corpus is constituted of news in English published on the web during two weeks in June 2008. This corpus is constituted of around 300,000 words (10Mb) which doesn't represent a very large corpus. These texts were taken from various press sources and they involve different themes (sports, technology, …). We extracted randomly a subset of articles and manually annotated 916 NEs (in our experiments, we deal with three types of annotation namely <person>, <organization> and <location>). This subset constitutes our test set.

In our experiments, first, we applied the XIP parser (Aït et al., 2002) to the whole corpus in order to construct the frequency matrix D given by (1). Next, we computed the similarity matrix between NEs according to (2) in order to obtain $\hat{s}$ defined by (4). Using the latter, we computed cliques of NEs that allow us to obtain the assignment matrix T given by (5). Then we applied the clustering heuristic described in Algorithm 1. At this stage, we want to build the NE resource using the clusters of cliques. Therefore, as described in §2.5, we applied two kinds of clusters annotations: the manual and the automatic processes. For the first one, we manually annotated the 100 biggest clusters of cliques. For the second one, we exploited the annotations provided by XIP NER (Brun and Hagège, 2004) and we propagated these annotations to the different clusters (see §2.5.2).

The different materials that we obtained constitute the CBC system's NE resource. Our aim now is to exploit this resource and to show that it allows to improve the performances of different classic NER systems.

The different NER systems that we tested are the following ones:

- CBC-NER system M (in short CBC M) based on the CBC system's NE resource using the manual cluster annotation (line 1 in Table 1),
- CBC-NER system A (in short CBC A) based on the CBC system's NE resource using the automatic cluster annotation (line 1 in Table 1),
- XIP NER or in short XIP (Brun and Hagège, 2004) (line 2 in Table 1),
- Stanford NER (or in short Stanford) associated to the following model provided by the tool and which was trained on different news corpora (CoNLL, MUC6, MUC7 and ACE): ner-eng-ie.crf-3-all2008-distsim.ser.gz (Finkel et al., 2005) (line 3 in Table 1),
- GATE NER or in short GATE (Cunningham et al., 2002) (line 4 in Table 1),
- and several hybrid systems which are given by the combination of pairs taken among the set of the three last-mentioned NER systems (lines 5 to 7 in Table 1). Notice that these baseline hybrid systems use the annotation combination process described in §2.6.1.

	Systems	Prec.	Rec.	F-me.
1	*CBC-NER system M*	*71.67*	*23.47*	*35.36*
	CBC-NER system A	*70.66*	*32.86*	*44.86*
2	*XIP NER*	*77.77*	*56.55*	*65.48*
	XIP + CBC M	78.41	60.26	**68.15**
	XIP + CBC A	76.31	60.48	67.48
3	*Stanford NER*	*67.94*	*68.01*	*67.97*
	Stanford + CBC M	69.40	71.07	70.23
	Stanford + CBC A	70.09	72.93	**71.48**
4	*GATE NER*	*63.30*	*56.88*	*59.92*
	GATE + CBC M	66.43	61.79	64.03
	GATE + CBC A	66.51	63.10	**64.76**
5	*Stanford + XIP*	*72.85*	*75.87*	*74.33*
	Stanford + XIP + CBC M	72.94	77.70	75.24
	Stanford + XIP + CBC A	73.55	78.93	**76.15**
6	*GATE + XIP*	*69.38*	*66.04*	*67.67*
	GATE + XIP + CBC M	69.62	67.79	68.69
	GATE + XIP + CBC A	69.87	69.10	**69.48**
7	*GATE + Stanford*	*63.12*	*69.32*	*66.07*
	GATE + Stanford + CBC M	65.09	72.05	68.39
	GATE + Stanford + CBC A	65.66	73.25	**69.25**

Table 1: Results given by different hybrid NER systems and coupled with the CBC-NER system

In Table 1 we first reported in each line, the results given by each system when they are applied alone (figures in italics). These performances represent our baselines. Second, we tested for each baseline system, an extended hybrid system that integrates the CBC-NER systems (with respect to the combination process detailed in §2.6.2).

The first two lines of Table 1 show that the two CBC-NER systems alone lead to rather poor results. However, our aim is to show that the CBC-NER system is, despite its low performances alone, complementary to other basic NER systems. In other words, we want to show that the exploitation of the CBC system's NE resource is beneficial and non-redundant compared to other baseline NER systems.

This is actually what we obtained in Table 1 as for each line from 2 to 7, the extended hybrid systems that integrate the CBC-NER systems (M or

A) always perform better than the baseline either in terms of precision[9] or recall. For each line, we put in bold the best performance according to the F-measure.

These results allow us to show that the NE resource built using the CBC system is *complementary* to any baseline NER systems and that it allows to improve the results of the latter.

In order to illustrate why the CBC-NER systems are beneficial, we give below some examples taken from the test corpus for which the CBC system A had allowed to improve the performances by respectively disambiguating or correcting a wrong annotation or detecting corpus-specific NEs.

First, in the sentence "From the start, his parents, *Lourdes* and Hemery, were with him.", the baseline hybrid system Stanford + XIP annotated the ambiguous NE "*Lourdes*" as <location> whereas Stanford + XIP + CBC A gave the correct annotation <person>.

Second, in the sentence "Got 3 percent chance of survival, what ya gonna do?" The back read, "A) Fight Through, b) Stay Strong, c) Overcome Because I Am a *Warrior*.", the baseline hybrid system Stanford + XIP annotated "*Warrior*" as <organization> whereas Stanford + XIP + CBC A corrected this annotation with <none>.

Finally, in the sentence "*Matthew*, also a favorite to win in his fifth and final appearance, was stunningly eliminated during the semifinal round Friday when he misspelled "secernent".", the baseline hybrid system Stanford + XIP didn't give any annotation to "*Matthew*" whereas Stanford + XIP + CBC A allowed to give the annotation <person>.

4 Related works

Many previous works exist in NEs recognition and classification. However, most of them do not build a NEs resource but exploit external gazetteers (Bunescu and Pasca, 2006), (Cucerzan, 2007).

A recent overview of the field is given in (Nadeau and Sekine, 2007). According to this paper, we can classify our method in the category of semi-supervised approaches. Our proposal is close to (Cucchiarelli and Velardi, 2001) as it uses syntactic relations (§2.2) and as it relies on existing NER systems (§2.6.2). However, the particularity of our method concerns the clustering of cliques of NEs that allows both to represent the different annotations of the NEs and to group the latter with respect to one precise annotation according to a local context.

Regarding this aspect, (Lin and Pantel, 2001) and (Ngomo, 2008) also use a clique computation step and a clique merging method. However, they do not deal with ambiguity of lexical units nor with NEs. This means that, in their system, a lexical unit can be in only one merged clique.

From a methodological point of view, our proposal is also close to (Ehrmann and Jacquet, 2007) as the latter proposes a system for NEs fine-grained annotation, which is also corpus dependent. However, in the present paper we use all syntactic relations for measuring the similarity between NEs whereas in the previous mentioned work, only specific syntactic relations were exploited. Moreover, we use clustering techniques for dealing with the issue related to over production of cliques.

In this paper, we construct a NE resource from the corpus that we want to analyze. In that context, (Pasca, 2004) presents a lightly supervised method for acquiring NEs in arbitrary categories from unstructured text of Web documents. However, Pasca wants to improve web search whereas we aim at annotating specific NEs of an analyzed corpus. Besides, as we want to focus on corpus-specific NEs, our work is also related to (Shinyama and Sekine, 2004). In this work, the authors found a significant correlation between the similarity of the time series distribution of a word and the likelihood of being a NE. This result motivated our choice to test our approach on recent news articles rather than on well-known annotated corpora.

5 Conclusion

We propose a system that allows to improve NE recognition. The core of this system is a clique-based clustering method based upon a distributional approach. It allows to extract, analyze and discover highly relevant information for corpus-specific NEs annotation. As we have shown in our experiments, this system combined with another one can lead to strong improvements. Other applications are currently addressed in our team using this approach. For example, we intend to use the concept of clique-based clustering as a soft clustering method for other issues.

[9] Except for XIP+CBC A in line 2 where the precision is slightly lower than XIP's one.

References

S. Aït, J.P. Chanod, and C. Roux. 2002. Robustness beyond shallowness: incremental dependency parsing. *NLE Journal.*

C. Bédécarrax and I. Warnesson. 1989. Relational analysis and dictionnaries. In *Proceedings of ASMDA 1988*, pages 131–151. Wiley, London, New-York.

C. Brun and C. Hagège. 2004. Intertwining deep syntactic processing and named entity detection. In *Proceedings of ESTAL 2004*, Alicante, Spain.

R. Bunescu and M. Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of EACL 2006.*

A. Cucchiarelli and P. Velardi. 2001. Unsupervised Named Entity Recognition using syntactic and semantic contextual evidence. *Computational Linguistics*, 27(1).

S. Cucerzan. 2007. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of EMNLP/CoNLL 2007*, Prague, Czech Republic.

H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. 2002. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of ACL 2002*, Philadelphia.

M. Ehrmann and G. Jacquet. 2007. Vers une double annotation des entités nommées. *Traitement Automatique des Langues*, 47(3).

J.R. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of ACL 2005.*

E.A. Fox and J.A. Shaw. 1994. Combination of multiple searches. In *Proceedings of the 3rd NIST TREC Conference*, pages 105–109.

Z. Harris. 1951. *Structural Linguistics*. University of Chicago Press.

J.A. Hartigan. 1975. *Clustering Algorithms*. John Wiley and Sons.

A. Kilgarriff, P. Rychly, P. Smr, and D. Tugwell. 2004. The sketch engine. In *In Proceedings of EURALEX 2004*.

V. Lavrenko and W.B. Croft. 2003. Relevance models in information retrieval. In W.B. Croft and J. Lafferty (Eds), editors, *Language modeling in information retrieval*. Springer.

D. Lin and P. Pantel. 2001. Induction of semantic classes from natural language text. In *Proceedings of ACM SIGKDD.*

D. Lin. 1998. Using collocation statistics in information extraction. In *Proceedings of MUC-7.*

J.F. Marcotorchino and P. Michaud. 1981. Heuristic approach of the similarity aggregation problem. *Methods of operation research*, 43:395–404.

P. Michaud and J.F. Marcotorchino. 1980. Optimisation en analyse de données relationnelles. In *Data Analysis and informatics*. North Holland Amsterdam.

D. Nadeau and S. Sekine. 2007. A survey of Named Entity Recognition and Classification. *Lingvisticae Investigationes*, 30(1).

A. C. Ngonga Ngomo. 2008. Signum a graph algorithm for terminology extraction. In *Proceedings of CICLING 2008*, Haifa, Israel.

M. Pasca. 2004. Acquisition of categorized named entities for web search. In *Proceedings of CIKM 2004*, New York, NY, USA.

S. Ploux and B. Victorri. 1998. Construction d'espaces sémantiques à l'aide de dictionnaires de synonymes. *TAL*, 39(1).

Y. Shinyama and S. Sekine. 2004. Named Entity Discovery using comparable news articles. In *Proceedings of COLING 2004*, Geneva.