Combining NER Systems via a UIMA-based platform

Baptiste Gaillard, Sylvie Guillemin-Lanne, Guillaume Jacquet, Claude Martineau, Aurélie Migeotte

Combining NER Systems via a UIMA-based platform

Baptiste Gaillard
Thales Communication
1-5 Avenue Carnot, 91883 Massy
Baptiste.gaillard@fr.thalesgroup.com

Sylvie Guillemin-Lanne
Temis –Tour Gamma B
193-197 rue de Bercy, 75582 Paris
sylvie.guillemin-lanne@temis.com

Guillaume Jacquet
XRCE – 6 chemin de Maupertuis,
38240 Meylan
Guillaume.Jacquet@xrce.xerox.com

Claude Martineau
IGM-LabInfo Université Paris-Est
77454 Marne-la-Vallée Cedex 2
stavroula.voyatzi@univ-mlv.fr

Aurélie Migeotte
Arisem – 1-5 rue Carnot
91883 Massy cedex
Aurelie.Migeotte@arisem.com

Abstract

In this paper, we present a tool aiming at merging named entity annotations provided by different named entity recognition systems. This tool is based on UIMA platform and contains a merging module which uses information about the compatibility of various annotations and can point out conflicts, and thus yields annotations that are more reliable than those of any single annotator. This work has been performed as part of the Infom@gic project

1 Introduction

In this paper, we present a tool aiming at merging named entity annotations provided by different Named Entity Recognition systems (henceforth NER systems). It take as input a set of text (simple text, xml or html) and give as output a set of annotated text in a UIMA format. It contains an algorithm for merging the different annotations which uses information about the compatibility of various annotations and can point out conflicts, and thus yields annotations that are more reliable than those of any single annotator. This work has been performed as part of the Infom@gic project, whose goal is the integration and validation of knowledge engineering and information analysis applications, and which is supported by the pole of competitiveness Cap Digital "Image, MultiMédia et Vie Numérique". We first describe UIMA, which provides architecture to coordinate information from different modules. Then, we present the needed input of our software. Then, we present an algorithm for merging the different annotations. We conclude by describing the software output and a preliminary evaluation of the results we obtained.

2 Connected works

In many scientific domains, it is well known that combining different methods or systems allows a result improvement comparing to results obtained by each system. For instance, the *Adaboost* Algorithm (Schwenk, 1999) consists in training a set of similar systems as a workflow where each system is trained to correct errors from the previous system.
In the NER domain, (Borthwick *et al.* 1998) and (Kozareva *et al.*, 2007) merged some NER system using majority vote method. In each case, the merged system obtains better precision results than any system alone.
Our tool aims at helping NER community to improve their results by merging existing systems. Our specificity is to propose a common platform where we can aggregate any NER system and where the merging module is already integrated. Moreover, inside the Infom@gic project, we developed this tool as an input for search engine or information extraction systems. Consequently, our tool deals with xml and html input format. Comparing to (Borthwick *et al.* 1998) and (Kozareva *et al.*, 2007), our last algorithm specificity is to deals with annotation overlapping cases.

3 The UIMA platform

UIMA (*Unstructured Information Management Architecture*) is an architecture for managing, organizing and coordinating unstructured information (Ferrucci and Lally, 2004). It originated at IBM and is now an open source project at the Apache Foundation (http://incubator.apache.org/uima/). This architecture has been developed specifically for the management of NLP tools: Ferrucci and Lally propose an example of rapid integration of a syntactic parser and a NER system. Its goal is to increase scientific progress with a fast combination of unstructured information management technologies. In our tool, we used UIMA as a common platform for all the used NER systems and for developing our merging module.

4 Software input and output

Our software takes as input a set of text (simple text, xml or html), integrate them in the workflow of a set of existing NER systems, merge their output and finally yield a UIMA CAS object (Common Analysis System object readable as an xml file) containing a unique set of annotations without redundancy, resolving some named entity boundary or identification conflicts.
Each NER system can be implemented directly using the UIMA libraries or using another language and then encapsulate it in the UIMA platform (this is the usual case). In this case, the only added work for an existing NER system is to establish a correspondence table between its own annotations and the UIMA platform annotations. Figure 1 describes the UIMA interface to construct the NER systems workflow, to define the used merging module and to choose the input and output format.

5 Algorithm for merging annotations

This merging process combines annotations from different NER systems in order to obtain a system that benefits from the unique characteristics of each annotator. This merging process is not trivial: First of all, all annotators must agree on a common type hierarchy of annotations. Secondly, the merging process must deals with redundant annotations. It corresponds to six different cases which are covered by our algorithm. Thirdly, it must deal with conflicting annotations.
The algorithm has been developed by the four partners of Experiment 1:

1. NEs with same offsets (A = B) :
    a. Same annotation → Merge
    b. NEs with different annotations where one annotation is descendant of the other
    → keep the most specific annotation
    c. NEs with different annotations that have a common parent type
    → use the parent annotation
    d. NEs with different annotations (other cases)
    → majority vote.
2. One NE included in another (A= Barack, B= Barack Obama or A= Obama, B= Barack Obama or A= Hussein, B=Barack Hussein Obama)
    a. Same annotation → keep the longer one.
    b. NEs with different annotations which have a common parent type or where one annotation type is a descendant of the other
    → keep the longer one with its type
    c. NEs with different annotations (other cases)
    → keep both NEs.
3. One NE overlaps another (AB= Barack Hussein, BC= Hussein Obama)
    a. Same annotation → merge ABC.
    b. NEs with different annotations which have a common parent type or where one annotation type is a descendant of the other
    → merge ABC with the coarsest annotation.
    c. NEs with different annotations (other cases)
    → keep both NEs.

6 Experiments and results

Our tool has been tested on two experiments.
Experiment 1 (Brun *et al*, 2009): software input is a set of web sites and the merged NER systems are four NER systems based on symbolic approaches implemented by four different partners (Arisem, IGM, Temis and Xerox). After an agreement on a common hierarchy of annotation types, each partner encapsulated its own NER system in a common platform. This experiment is done on French texts. Table below present a summary of the obtained results:

	P	R	F2
Annot. 1	67.53	60.85	64.02
Annot. 2	61.85	44.36	51.67
Annot. 3	73.24	49.38	58.99
Annot. 4	75.23	66.35	70.51
Fusion	**80.10**	**65.11**	**71.83**

Experiment 2 (Ah-Pine and Jacquet, 2008): The input is a hybrid case combining a new statistical approach implemented in the UIMA formalism with different existing systems: Stanford, Gate and Xerox NER systems. This experiment is done on English texts. Table below present a summary of the obtained results:

	P	R	F2
Xerox system	77.77	56.55	65.48
Stanford system	67.94	68.01	67.97
CBC system	70.66	32.86	44.86
Fusion	**73.55**	**78.93**	**76.15**

We refer to the corresponding papers for more details on these experiments. These results must be seen as an evidence that our software has been implemented, that it works with different set of NER systems and with different input types: html texts for experiment 1 and classic texts for experiment 2 (cf. figures 2 and 3 for some illustrations of these texts). Finally, these experiments show that the merged system improve the results of each initial NER system. This improvement is a last evidence of the usefulness of such tool for the NLP community.

Demo technical requirements

This demo doesn't need any specific technical requirement except an internet access.

References

Ah-Pine J. and Jacquet G.. 2009. *Clique-Based Clustering for improving NER Systems*, in proceedings of EACL2009, Athens.

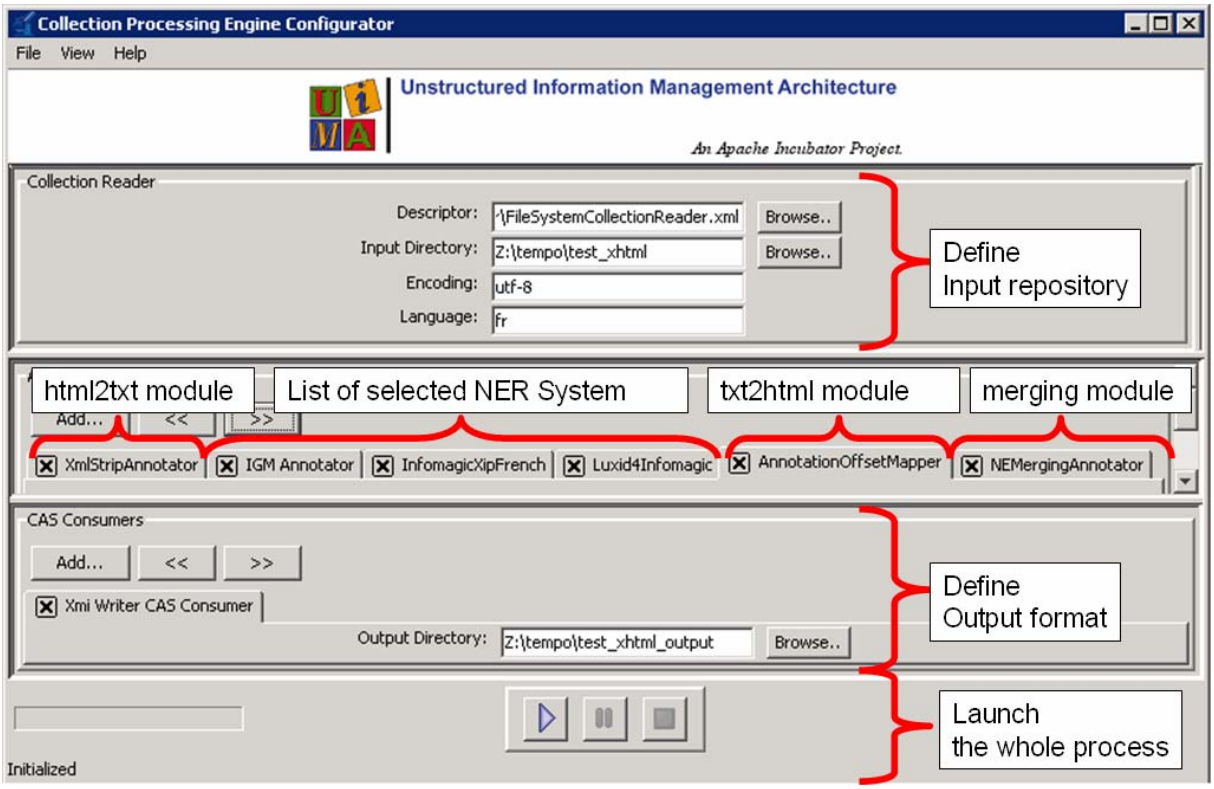
Borthwick. A., Sterling J., Agichtein E. et Grishman R. (1998). Exploiting diverse knowledge sources via maximum entropy in named entity recognition, In *Proceedings of the Sixth Workshop on Very Large Corpora, Montreal*, pp. 152-160.

Brun C., Dessaigne N., Ehrmann M., Gaillard B., Guillemin-Lanne S., Jacquet G., Kaplan A., Kucharski M., Martineau C., Migeotte A., Nakamura T., Voyatzi S. submitted. *Une expérience de Fusion pour l'annotation d'entités nommées*, in proceedings of TALN2009, Senlis.

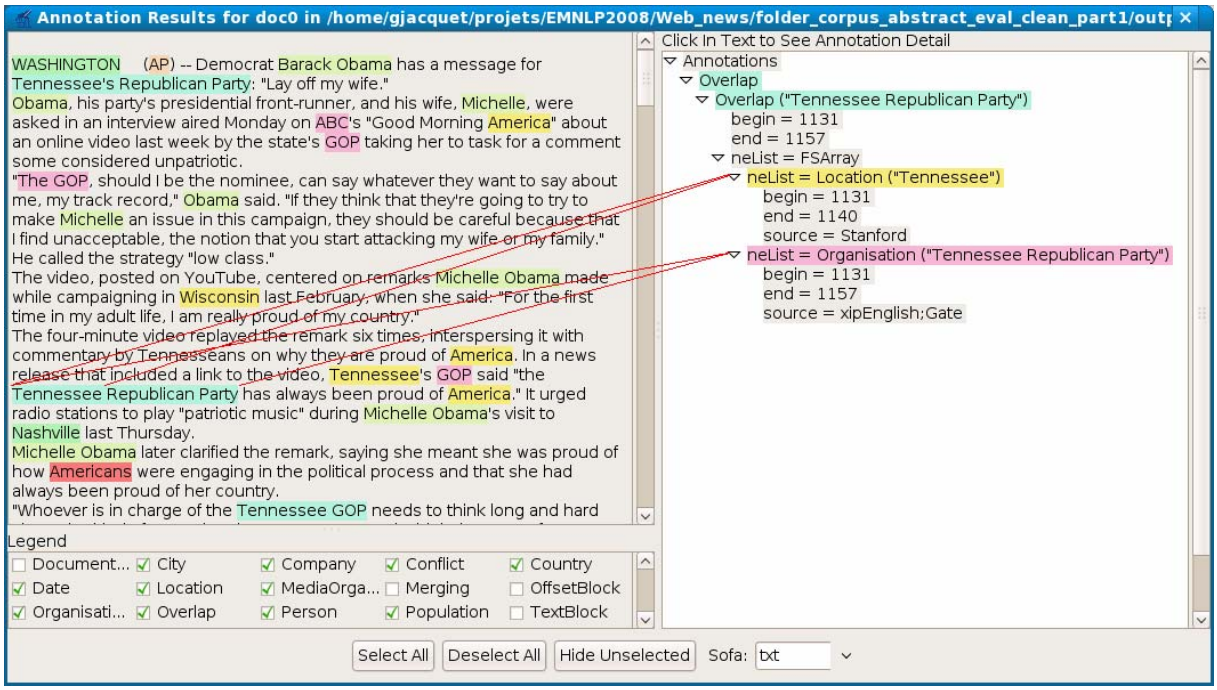
Ferrucci D. and Lally A. 2004. UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 327-348.

Kozareva Z., Ferrández O., Montoyo A., Muñoz R. et Suárez A. (2007). Combining Data-Driven Systems for Improving Named Entity Systems, Data & Knowledge Engineering, 61:3, Elsevier Science Publishers B.V., Amsterdam, The Netherlands,.

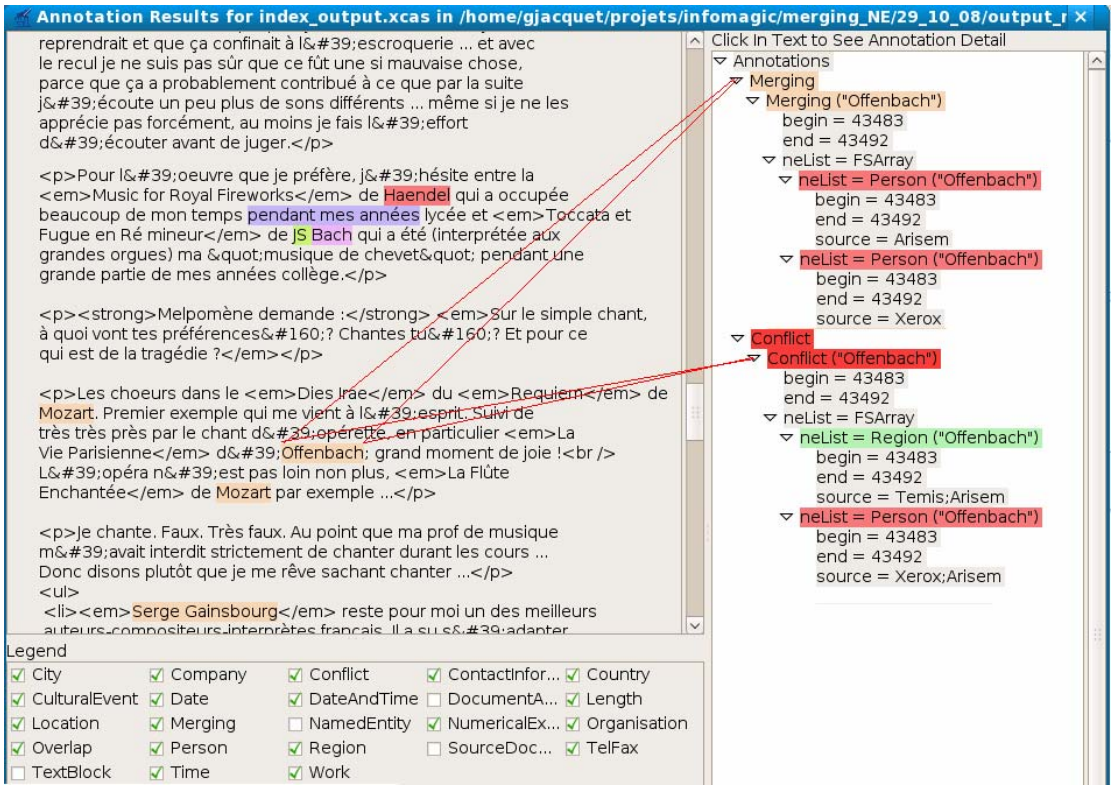
Schwenk H. 1999. Using boosting to improve a hybrid Hmm/Neural Network speech recognizer, In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Phoenix, AZ, USA, pp. 1009-1012.



UIMA Interface for workflow construction and launching



Output visualization from Experiment 2 on an English text. We focus on how our system describes an overlap case (« Tennessee Republican Party » as *Organisation* and « Tennessee » as *Location*)



Output visualization from Experiment 1 on a French html file. We focus on how our system describes a conflict case (« Offenbach » as a *Region* and « Offenbach » as a *Person*)