ROBUST TREE-STRUCTURED NAMED ENTITIES RECOGNITION FROM SPEECH

Christian Raymond

INSA de Rennes - IRISA
20 avenue des buttes de Coësmes, Rennes, France
christian.raymond@irisa.fr

ABSTRACT

Named Entity Recognition is a well-known Natural Language Processing (NLP) task, used as a preliminary processing to provide a semantic level to more complex tasks. Recently a new set of named entities has been defined; this set has a multilevel tree structure, where base entities are combined to define more complex ones. In this paper I describe an effective and original NER system robust to noisy speech inputs that ranked first at the 2012 ETAPE NER evaluation campaign with results far better than those of the other participating systems.

Index Terms— named entities recognition, Conditional Random Fields, discretization of numeric features

1. INTRODUCTION

Named Entity (NE) Recognition is a well-known Natural Language Processing (NLP) task, NE can be seen as a first level generic semantic information which can be found in many documents (text, audio, video). NEs have been proved to be helpful to improve the quality of many higher level natural language processing tasks: [1, 2] used NEs to improve machine translation and [3] for automatic summarization. NEs are also crucial for precise information retrieval systems such as Question-Answering systems [4]. Recently a new set of named entities has been defined [5]. These named entities have a multilevel tree structure where components are combined to define more complex and general entity structures. This definition increases significantly the complexity of the NER task, even more due to the type of data used for the annotation: transcriptions of French broadcast data. Given such a definition, it is not possible to tackle the task with traditional sequence labeling approaches. Since this dataset is relatively new, few published papers refer to it; as far as I know mostly all statistical NER systems deal with the structure thanks to cascade approaches like the previous best system published on this dataset [6]. Cascade methodology has the main drawback of propagating the errors made in the previous stages of the whole process; this is particularly harmful when processing noisy inputs like automatic speech transcriptions where the number of potential errors is high. I present in this paper a different way to deal with NE structure avoiding cascade processes. The proposed method has the merit to be light, fast to train and to outperform much more complex approaches. Indeed, this system was ranked first with results far better than those of the other participating systems in the ETAP NE evaluation campaign.

This paper is organized as follows: in section 2 I describe the set of named entities used for evaluation. In section 3 I detail the system I propose, and implemented for the 2012 ETAP Named Entities Recognition evaluation campaign. This system uses a conditional random field (CRF), a state-of-the-art machine learning approach to sequence labeling problems. Another interest of this paper is that I review in section 4 a discretization method useful to deal efficiently with numeric features in CRF. In section 5 I present succinctly the different challenging system in ETAPE. Section 6 presents the ETAPE evaluation campaign results[1].

2. STRUCTURED NAMED ENTITIES IN ETAPE

The set of NEs used in this work has been recently defined in [7]. It presents an important difference with respect to previous sets: NEs have a tree structure and are both hierarchical and compositional. For example, type pers (person) is split into two subtypes, pers.ind (individual person) and pers.coll (collective person), and pers entities are composed of several components, among which are name.first and name.last. Figure 1 shows the taxonomy composed of 7 main types (person, function, organization, location, product, amount, and time) and 32 subtypes. Figure 2 shows all the components. Figure 3 shows an annotation example using this definition.

The ETAPE data [8] consist of 13.5 hours of radio data and 29 hours of TV data, selected to include mostly non planned speech and a reasonable proportion of multiple speaker data. 250 hours of radio broadcast news, from French, Moroccan and African radio stations from the previous ESTER 2 campaign [9] have been re-annotated according to the new NE scheme. Table 1 presents a summary of the ETAPE data.

[1] preliminary results are presented because the adjudication process is not complete

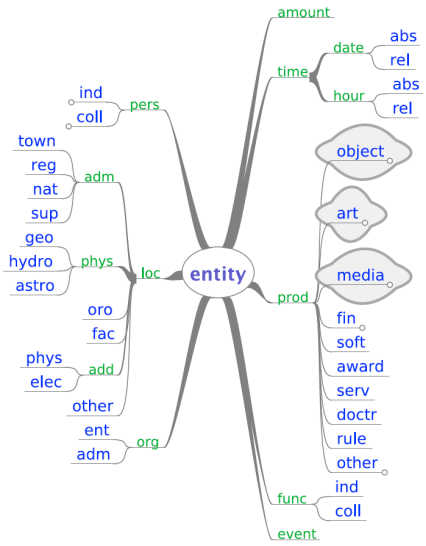

Fig. 1. Named entity hierarchy

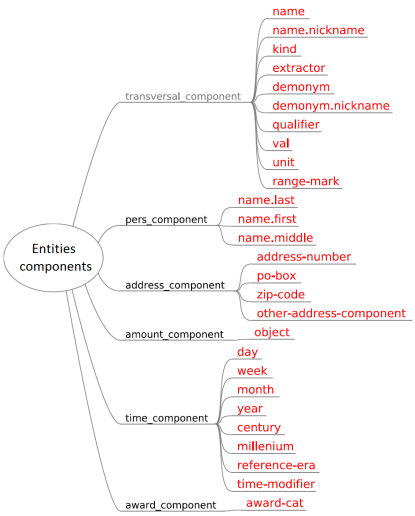

Fig. 2. Entities components

genre	train	dev	test	total	ester2
TV news	7h30	1h35	1h35	10h10	
TV debates	10h30	2h40	2h40	16h00	
TV amusements		1h05	1h05	2h10	
Radio shows	7h50	3h00	3h00	13h50	250h
Total	25h30	8h20	8h20	42h10	250h

Table 1. ETAPE 2011 data summary

3. SYSTEM DESCRIPTION

Grammar or formal based approaches fail to operate on noisy inputs like automatic transcriptions while Conditional Random Fields (CRF) have been proven to be very efficient both on clean text and noisy ones in comparison to other statistical approaches [10] to solve sequence labeling problems. Previous flat named entities detection tasks have been tackled successfully using a CRF based approach [11, 12]. Applying CRF in our case of structured named entities is not really straightforward; the previous best system proposed for this task [6] decomposed the problem in a cascade of two subproblems: first, they use a CRF to detect components, thus use a Probabilistic Context Free Grammar (PCFG) to build the named entity structure. Although the system proposed was the best system presented in the Quaero named entity detection evaluation campaign [13], it suffers for some drawbacks: first of all, the cascade approach is itself problematic, all errors made in the first stage would impact the second one. Secondly, the CRF used in the first stage is very huge, using billions of features and need a strategy (presented in their paper) to be trained fitting the computer memory.

As previous system authors, I don't know any efficient machine learning algorithm able to learn directly the structure in a reasonable learning time; thus I propose a new approach avoiding especially the two drawbacks I pointed out previously. My approach is the following:

1. because I don't know how to learn directly the structure and I do not think a cascade approach is a good solution: I choose to ignore the structure. All components and named entities are considered independently. I believe detecting components are not mandatory to detect named entities; they can be detected directly and potential errors coming from the component detection stage will be avoided. The main drawback is that all dependencies between components/entities are lost; my belief is that losing this dependency information would have less bad impact than keeping a cascade strategy.

2. I learn a CRF for each component/entity, involving as many classifiers as numbers of unique components/entities. The number of classifiers is much greater than the previous method but each of them is really fast to train since it contains only two labels, the component/entity *vs.* the rest (actually 3 labels using the BIO scheme), thus training all of them is much faster than training the CRF approach proposed in [6].

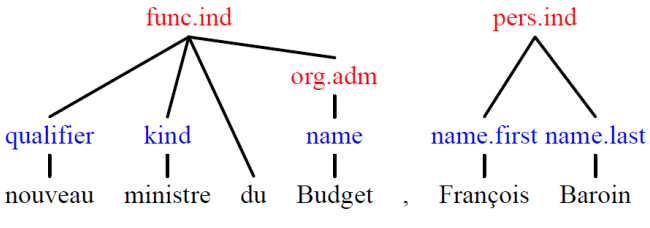

Fig. 3. Multi-level annotation of entity types (red tags) and components (blue tags): new minister of budget , François Baroin.

3. last, I have a simple procedure to re-construct the structure: I align the different CRF results and I order the CRF responses according to the relative order of component/entity in the structure observed in the corpus. To avoid expected segmentation errors that could appear if a component ends after the overhead entity in the alignment, I just force the entity to finish after all inside entities/components end.

I learn 68 binary CRF, thanks to Wapiti [14] a CRF open source implementation, to learn each entity/component using the BIO scheme to deal with the possible case where the same entity/component follows another one. Each CRF uses features extracted in two-levels of information [12]; see figure 4:

1. the first level: the words
2. the second level: a composition of three different information aspect:
   (a) classes corresponding to *a priori* knowledge (list of cities, countries, *etc.*)
   (b) or the word itself, if the word has a high mutual information with a component/entity
   (c) or the corresponding Part-Of-Speech for words that do not belong to the previous cases

Each CRF extracts *Ngram* features with $N = 3$ within a window $[-2, +2]$ around the decision time state. The idea to fuse the 3 information aspects together is crucial: let's consider the 3 information aspects separately, if I extract *Ngrams*[2] from POS only, extracted features will have low precision power because POS is too general. Extracting word *Ngrams* is the opposite; the features will be too precise with very low generalization power (recall). Combining the 3 information parts allows extracting *Ngrams* having both properties that can be called robust patterns. These robust patterns are more predictive than the addition of patterns built on POS and words.

I wanted to add the word size (number of letters) to the list of features. This information might be interesting for some special entities components like zip code or year with specific length. Current CRF implementations do not manage numeric features, and numeric values are considered as symbols. I review in the next section a solution to deal with numeric feature within a CRF. Marginal improvements could be obtained adding several common features (*i.e.*, word prefix or suffix) but this is not the point of this paper and did not use them.

Adaptation to automatic transcription has been done in a simple way: I just removed all capitalization and punctuations from the data before training my system.

[2] with $N > 1$

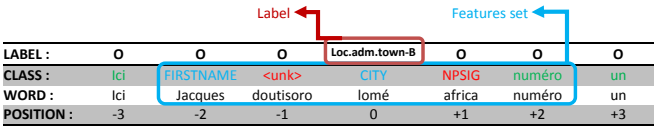

Fig. 4. Set of features used within the CRF model. Two levels of information are used, the first one is composed of words, the second one by a corresponding class of the word among three possibles ones defined in 2a,2b and 2c

$$\begin{aligned} & \frac{\log_2(N-1)}{N} \\ + & \log_2(3^k - 2) \\ - & [k.Ent(F) - k_1.Ent(S_1) - k_2.Ent(S_2)] \end{aligned}$$

Fig. 5. Threshold used to stop the decision tree induction if the entropy gain of a split is below. N is the number of examples in F (Father Node), S_1 and S_2 are the left and right children. k is the number of unique labels present in F while k_1 and k_2 are the number of unique labels in S_1 and S_2. Ent stands for Entropy.

4. INTEGRATION OF NUMERIC FEATURES IN CRF

Integration of numeric features in CRF is not straightforward because CRF implementations process numeric values as symbols. For most of numeric feature, doing so is nonsense while for others too much symbols make the feature inefficient. A solution is to discretize numeric features; thus grouping together similar values and reducing the number of symbols. Most of the time, the discretization process is done by selecting empirically the number of classes and the intervals of values. The result is far from optimal, especially when the number of numeric features is high and features difficult to interpret. An efficient method to transform numeric features to discrete ones is the method described in [15]. This supervised method allows finding automatically both the number of classes and the intervals of values. The algorithm is the following:

1. consider each numeric attribute independently
2. induce a binary decision tree based on the information gain (like ID3 or C4.5) to predict labels from this single attribute
3. stop when the information gain (entropy) of a split is below a threshold computed as shown in figure 5:
4. repeat the operation for all numeric attributes

discretize4crf [16] is an open source implementation of this principle that works with the Wapiti file format.

The numeric feature used in this work is not very good to illustrate the relevance of the method. In order to pro-

pose a good illustration I discretized a word confidence measure. This confidence measure is a probability that a word of an automatic transcription is correctly recognized. Figure 6 shows the decision tree and the discrete classes computed. We can observe that discrete classes group samples with different probability to be correct or not.

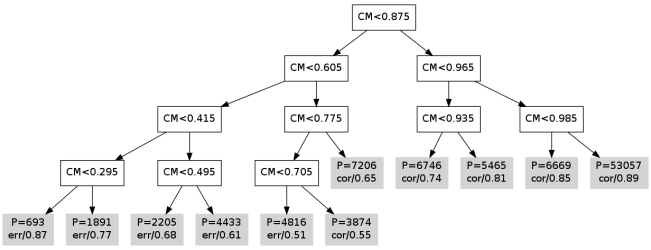

Fig. 6. Decision tree induced according to the discretization method. Leaves represent discrete classes. P is the number of learning samples in the leaf. For each leaf is mentioned the probability for a word to be correct(cor) or incorrect(err).

5. BRIEF DESCRIPTION OF 4 BEST CHALLENGING SYSTEMS

I'm providing here a very basic view of the 4 best challenging systems in ETAPE. Systems 1 and 2 are particularly interesting to compare with my system (sys0) because they involve CRF but differ in the way they deal with the NE structure.

- sys1 this system uses a two-stage approach; it uses CRF for labeling the components then a PCFG for semantic tree reconstruction; this approach is close to the one described in [6]. Adaptation to automatic transcriptions is done by case reconstruction and punctuation addition.
- sys2 this system decomposes the NE structure at a hierarchic level (specific depth in the tree structure) and learns a CRF by level for the entities. Components are retrieved by a local classifier knowing the underlying structure.
- sys3 rule-based system with knowledge sources
- sys4 use a data mining approach to extract NE annotation rules

6. SYSTEM RESULTS

6.1. Evaluation Metrics

All results are expressed in terms of Slot Error Rate (SER) [17], which has a similar definition of word error rate for ASR systems, with the difference that substitution errors are split in three types:

1. correct entity type with wrong segmentation;
2. wrong entity type with correct segmentation;
3. wrong entity type with wrong segmentation;

	Man.	Rov.	s23	s24	s25	s30
sys0	**33.81**	**55.51**	**58.35**	**63.40**	**62.53**	**52.71**
sys1	36.44	67.16	68.57	67.73	75.02	60.44
sys2	43.58	69.54	74.55	71.93	85.60	69.24
sys3	42.89	68.65	74.93	70.77	86.10	66.23
sys4	41.01	65.97	71.01	66.89	90.32	65.37
sys5	55.63	94.24	107.71	82.67	142.96	97.19
sys6	62.76	76.45	80.84	77.97	82.71	76.63
sys7	84.78	98.82	101.45	95.03	100.72	97.28

Table 2. Performances of systems from the ETAPE campaign in terms of SER, on the manual transcription and the automatic transcriptions produced by 4 different automatic speech recognition systems where the corresponding word error rate is indicated after the s, the column Rov. indicates the SER on the rover combination of the 4 ASR system outputs

here, 1 and 2 are given half points, while 3, as well as insertion and deletion errors, are given full points.

6.2. Systems comparison

Table 2 presents the results in terms of SER of the 8 participants at the ETAPE NER evaluation campaign. SER is given for manual transcription, 4 automatic transcriptions produced by 4 different automatic speech recognition systems with different word error rates (23,24,25 and 30%) plus a rover combination of them. Results clearly show that the proposed approach is the most efficient. In comparison to the second best system sys1, the improvement is not large on the clean transcription $\approx$3% absolute, but on the automatic transcriptions, improvement goes up to 13% absolute. That clearly demonstrates that the strategy proposed is better than the cascade strategy that is no more efficient: mistakes made on the first stage have an impact on the second stage.

7. CONCLUSION

I propose in this work a named entities recognition system for tree-structured named entities robust to noisy speech inputs.This system got the best results in the ETAPE evaluation campaign with results far better than the other participants. As opposite to many challenger systems that have tackled the problem of tree-structure with cascade approaches, I decomposed the problem passing over the structure to avoid cascade processes. Although all explicit dependencies between the NE structure elements are lost, results clearly demonstrate that this choice was better to tackle the problem of tree-structured named entities while it leaves room for a lot of improvements. In a future work, I will investigate a way to modelize dependency information among NE elements keeping the same decomposition method I proposed.

8. REFERENCES

[1] Bogdan Babych and Anthony Hartley, "Improving machine translation quality with automatic named entity recognition," in *Proceedings of the 7th International EAMT workshop on MT and other Language Technology Tool*, Morristown, NJ, USA, 2003, pp. 1–8.

[2] Rohini K. Srihari and Erik Peterson, "Named entity recognition for improving retrieval and translation of chinese documents," in *Proceedings of the 11th International Conference on Asian Digital Libraries*, Berlin, Heidelberg, 2008, pp. 404–405.

[3] Martin Hassel, "Exploitation of Named Entities in Automatic Text Summarization for Swedish," in *Proceedings of NODALIDA'03 - 14th Nordic Conference on Computational Linguistics*, Reykjavik, Iceland, 2003.

[4] Mahboob Alam Khalid, Valentin Jijkoun, and Maarten De Rijke, "The impact of named entity normalization on information retrieval for question answering," in *Proceedings of the 30th European conference on Advances in information retrieval*, Berlin, Heidelberg, 2008, pp. 705–710.

[5] Cyril Grouin, Sophie Rosset, Pierre Zweigenbaum, Karën Fort, Olivier Galibert, and Ludovic Quintard, "Proposal for an extension of traditional named entities : From guidelines to evaluation, an overview," in *Fifth Linguistic Annotation Workshop (LAW-V)*, Association for Computational Linguistics., Ed., Portland, Oregon, 2011, pp. 92–100.

[6] Marco Dinarelli and Sophie Rosset, "Models cascade for tree-structured named entity detection," in *Proceedings of 5th International Joint Conference on Natural Language Processing*, Chiang Mai, Thailand, November 2011, pp. 1269–1278, Asian Federation of Natural Language Processing.

[7] Sophie Rosset, Cyril Grouin, and Pierre Zweigenbaum, *Entités nommées structurées : guide d'annotation Quaero*, 2011.

[8] Guillaume Gravier, Gilles Adda, Niklas Paulson, Matthieu Carré, Aude Giraudel, and Olivier Galibert, "The ETAPE corpus for the evaluation of speech-based TV content processing in the French language," in *International Conference on Language Resources, Evaluation and Corpora*, Turquie, 2012, p. na.

[9] Sylvain Galliano, Guillaume Gravier, and Laura Chaubard, "The ester 2 evaluation campaign for the rich transcription of french radio broadcasts.," in *Interspeech*. 2009, pp. 2583–2586, ISCA.

[10] Stefan Hahn, Marco Dinarelli, Christian Raymond, Fabrice Lefèvre, Patrick Lehnen, Renato De Mori, Alessandro Moschitti, Hermann Ney, and Giuseppe Riccardi, "Comparing stochastic approaches to spoken language understanding in multiple languages," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 6, pp. 1569–1583, Aug. 2011.

[11] Andrew McCallum and Wei Li, "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons," in *CoNLL-2003*, 2003, pp. 188–191.

[12] Christian Raymond and Julien Fayolle, "Reconnaissance robuste d'entités nommées sur de la parole transcrite automatiquement," in *Traitement Automatique des Langues Naturelles*, Montréal, Canada, July 2010.

[13] Olivier Galibert, Ludovic Quintard, Sophie Rosset, Pierre Zweigenbaum, Claire Nedellec, Sophie Aubin, Laurent Gillard, Jean-Pierre Raysz, Delphine Pois, Xavier Tannier, Louise Deléger, and Dominique Laurent, "Named and specific entity detection in varied data: The quæro named entity baseline evaluation," 2010.

[14] Thomas Lavergne, Olivier Cappé, and François Yvon, "Practical very large scale CRFs," in *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*. July 2010, pp. 504–513, Association for Computational Linguistics.

[15] Usama M. Fayyad and Keki B. Irani, "Multi-interval discretization of continuousvalued attributes for classification learning," in *Thirteenth International Joint Conference on Articial Intelligence*. 1993, vol. 2, pp. 1022–1027, Morgan Kaufmann Publishers.

[16] Christian Raymond, "discretize4crf," `https://gforge.inria.fr/projects/discretize4crf/`, 2011.

[17] John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel, "Performance measures for information extraction," in *In Proceedings of DARPA Broadcast News Workshop*, 1999, pp. 249–252.