Structured Named Entity Retrieval in Audio Broadcast News

Azeddine Zidouni  
LSIS Laboratory  
Univ. Aix-Marseille 2, France  
azeddine.zidouni@lsis.fr

Mohamed Quafafou  
LSIS Laboratory  
Univ. Aix-Marseille 2, France  
mohamed.quafafou@univmed.fr

Hervé Glotin  
LSIS Laboratory  
Univ. Sud Toulon Var, France  
glotin@univ-tln.fr

Abstract

This paper focuses on the role of structures in named entity retrieval inside audio transcription. We consider the transcription documents structures that guide the parsing process, and from which we deduce an optimal hierarchical structure of the space of concepts. Therefore, a concept (named entity) is represented by a node or any sub-path in this hierarchy. We show the interest of such structure in the recognition of the named entities using the Conditional Random Fields (CRFs). The comparison of our approach to the Hidden Markov Model (HMM) method shows an important improvement of recognition using Combining CRFs. We also show the impact of time axis in the prediction process.

1. Introduction

In multimedia data processing, information retrieval in its raw state is a very difficult task and requires several resources. Add to this, the enrichment data by the domain knowledge as marking Named Entities (NEs) annotations is not possible at the level of audio signal. These constraints require the use of another representation in transcription format. The audio transcriptions [3, 4] represent the same information as that on the original media, with the possibility to index and enrich it with metadata. The transcripts are enriched further for the construction of a model representing knowledge.

This paper considers dialog applications that support multiple-speakers dialogues broadcast. Learning and test data consist of French broadcast news where each broadcast is represented by a transcription in XML format (Figure 1). These transcriptions are improved by a set of secondary information as sentence segmentation and named entity (NE) annotations. The evaluation of such information is an important task that enables efficient indexing and search. To achieve this goal, we will present an approach based on an a priori knowledge of hierarchical concepts in the transcription documents for Named-Entity Recognition (NER) [5, 6] task. The NER task consists of identifying and classifying proper names in texts, including locations; people; and organizations. Given a sentence, the first step in NER task is the segmentation of words which are part of entities, and then, the classification of each entity by type. Each entity can be defined at several conceptualization levels. To define the label at one level, we use the predictions of the other levels. For example, to predict the label Localisation for Paris we use the label Geographical as an a priori information. The challenge of the NER task is that many NEs are too rare to appear even in a large training data set, and therefore it becomes necessary to identify them based only on context.

The remainder of this paper is organized as follows: In Section 2, we expose the data transcriptions and their specification. We then in Section 3, describe the NER task using two graphical models: Hidden Markov Machine (HMM), and Conditional Random Fields (CRFs). In Section 4, we define the hierarchical concepts and explain our proposed method to improve the performance of the classical approach. In Section 5, we present the experimental results considering both prediction precision and model construction time cost. We conclude in Section 6.

2 Audio broadcast news: data description

Data corpus [2] consisting on newspapers and radio broadcasts, are segmented into sections. Each section is
dedicated to a thematic set which implies speakers and guests. The provided corpus consists of 90 hours of French radio with transcription and annotation. This corpus is divided into three parts: the first one is used to NER learning models (train) whereas the second part, development set (dev), is used for the adjustment of inference parameters prediction. The third part, test sets, is used for the evaluation and the assessment of the learned models performances. In our evaluation, we choose two test sets test1 and test2 to evaluate the model in different conditions (temporal and origin). Table 1 shows the distribution of data by radio sources and Table 2 shows the distribution by dates.

<table>
<thead>
<tr>
<th>Radio</th>
<th>train</th>
<th>dev</th>
<th>test 1</th>
<th>test 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>France Inter</td>
<td>22h</td>
<td>6h</td>
<td>1h</td>
<td>5h</td>
</tr>
<tr>
<td>RFI-FM</td>
<td>18h</td>
<td>5h</td>
<td>2h</td>
<td>-</td>
</tr>
<tr>
<td>France Info</td>
<td>8h</td>
<td>1h</td>
<td>1h</td>
<td>-</td>
</tr>
<tr>
<td>RTM</td>
<td>17h</td>
<td>3h</td>
<td>1h</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>65h</td>
<td>15h</td>
<td>5h</td>
<td>5h</td>
</tr>
</tbody>
</table>

Table 1. distribution of data by sources.

<table>
<thead>
<tr>
<th>date</th>
<th>train</th>
<th>dev</th>
<th>test 1</th>
<th>test 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1998</td>
<td>16h</td>
<td>3h</td>
<td>1h</td>
<td>-</td>
</tr>
<tr>
<td>1999</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>5h</td>
</tr>
<tr>
<td>2000</td>
<td>12h</td>
<td>2h</td>
<td>1h</td>
<td>-</td>
</tr>
<tr>
<td>2003</td>
<td>37h</td>
<td>10h</td>
<td>3h</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>65h</td>
<td>15h</td>
<td>5h</td>
<td>5h</td>
</tr>
</tbody>
</table>

Table 2. distribution of data by dates.

Figure 1. An example of an audio transcription in XML format.

3 Named-entity retrieval using graphical models

The words-sequence labeling problem can be formalized as follows: Let \( X = \langle x_1, x_2, \ldots, x_n \rangle \) be some observed input data sequence, such as sequence of words in speech, find the sequence of states \( Y = \langle y_1, y_2, \ldots, y_n \rangle \) (sequence of labels associated to the input sequence) which maximize the conditional probability \( P(Y|X) \):

\[
\hat{Y} = \arg \max_{Y} P(Y|X).
\]

The next subsections describe the usual model for such modelisation Hiden Markov Models and a more relevant approach Conditional Random Fields.

3.1 Hidden Markov Model

The Hidden Markov models (HMMs) [10] are used to represent the joint probability distribution \( P(Y,X) \) in sequence data, where the variable \( Y \) represents the attributes of the entities that we wish to predict, and the input variable \( X \) represents our observed knowledge about the entities (Figure 2). In the NER example, the observation \( x_t \) is the identity of word, and the state \( y_t \) is the Named-Entity label. HMM has two assumptions: (a) each state \( y_t \) depends only on its predecessor \( y_{t-1} \) (or a window of \( k \) predecessors), (b) each observation variable \( x_t \) depends only on the current state \( y_t \):
belonging of sequences [1], NEs recognition [9], or even the thematic classification.

**Definition:** Let \( X = \langle x_1, x_2, \ldots, x_n \rangle \) be an sequence of observed words of length \( n \) (input variables). Let \( \varphi \) be a set of states that we wish to predict (output variables). Every variable \( y \in \varphi \) takes outcomes from a set \( \varphi \), which is discrete. Let \( Y = \langle y_1, y_2, \ldots, y_n \rangle \) be the sequence of states in \( \varphi \) that correspond to the labels assigned to words in the input sequence \( X \). Linear-chain CRFs define the conditional probability of a state sequence \( Y \) given an input sequence \( X \) as follows:

\[
p(Y|X) = \frac{1}{Z(X)} \exp\left\{ \sum_{i=1}^{n} \sum_{k=1}^{K} \lambda_k f_k(y_{i-1}, y_i, X, i) \right\},
\]

where the normalization factor \( Z(X) \) is the sum of the scores of all possible state sequences. The number of state sequences is exponential in the input sequence length \( n \).

The CRFs parameters estimation problem is to determine the parameters \( \Lambda \) from the train data \( D = \{(x_i, y_i)\}_{i=1}^{N} \) with empirical distribution \( \bar{p}(x, y) \). In [8], the authors describe an iterative scaling algorithm that maximizes the log-likelihood objective function \( \Im(\theta) \). Where the conditional probability focuses on a sequence of elements, the Viterbi algorithm can be applied to maximize this function:

\[
\Im(\theta) = \sum_{i=1}^{N} \log P_{\theta}(x_i|y_i).
\]

Most of the works [5, 9] developed to deal with NER task use string symbols to represent NEs. An inductive learning algorithm is then applied to construct a predictive model. Unfortunately, such approaches do not consider carefully the inherent structure of the NEs space. In the next section, we show how this space structure is deduced from the training data and how the NER task is rewritten from string symbols to concepts recognition. A new learning algorithm is then proposed to recognize NEs considering their contexts.

### 4 Named-entity recognition using CRFs and concepts hierarchy

#### 4.1 Concepts hierarchy deduction

Each NE \( y \) is represented as one concept \( y^1 \) or several concepts \( y^1, y^2, \ldots, y^n \) connected by the concatenation operator, denoted \( \cdot \). Consequently, we have \( y = y^1 \cdot y^2 \cdot \ldots \cdot y^n \).
with the following semantic where each concept \( y^i \) is subsumed by the concept \( L^{i-1} \) for \( i \in \{2, 3, \ldots, n\} \) and the concept \( y^1 \) is subsumed by the most general concept in our representation called entity (The operator \( \langle \ldots \rangle \) means a subsumption relationship). Therefore, each concept is a node in the concept hierarchy and a NE is represented as a path in this hierarchy. The operator simplify is applied to reduce the representation of our concepts hierarchy and makes it more compact. In the corpus annotation, the maximum number of levels is 3 (see figure 3). Thus, a label has the form \( y = y^1, y^2, y^3 \) where \( y^1 \in \text{level}(L_1) \), \( y^2 \in \text{level}(L_2) \), and \( y^3 \in \text{level}(L_3) \). For example, we associate the label pers.hum with Michael, where pers corresponds to the most general concept Person, hum is the most specific concept Human.

4.2 Hierarchical composition

The simplest application on CRFs is to consider each label \( y_i \) as the String concatenation of the three levels concepts \( y^1_i, y^2_i \) and \( y^3_i \) (see Figure 3). The problem is that this approach assumes that given input, all of the NE labels are independent. In fact, the number of labels is more important (it contains all the possible combinaisons of the three levels) and requires a large amount of data learning to construct model that characterizes all the labels. In our approach, we construct one model for each conceptualization level. Each model \( M_k \) with \( k \in \{L_1, L_2, L_3\} \) is defined in a non ambiguous domain \( D_k = \{(x_i, y^k_i)\}_{i=1}^N \). These levels are intertwined (organized into categories and subcategories) and relations can be defined between them. Indeed, the learning phase of the CRFs makes possible the integration of several input-entities features. This property can improve classification by integrating relationship between levels (see the next subsection for an illustrative example).

4.3 Example

Figure 4 depicts an application of labeling by levels for a sentence composed by 8 words \( \langle w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8 \rangle \). The \( L_1 \) level labeling generates three labels \( y^1_1, y^1_3 \) and \( y^1_4 \) (Figure 4.a). With the same manner, the labeling at level \( L_2 \) and \( L_3 \) are given by Figure 4.b and Figure 4.c respectively. The final labeling is the concatenation of the three levels labeling. In the later, we notice that the word \( w_4 \) is labeled at \( L_2 \) and \( L_3 \) but not at \( L_1 \) level. In this case we can improve the labeling approach by affecting to \( w_4 \) the label \( y^1_4 \) which is associated to the branch \( y^1_2, y^3_3, y^3_2 \) in the labeling tree.

4.4 Combined models

The conditional learning of the CRFs method can be enriched by the information of other levels. Using this property, we can apply a combined learning by levels on the data corpus. The \( M_k \) models built separately give us a prediction for each level on the test data. These predictions can be used as input entities features. Thus, we can built combined models \( M_k^{\text{comb}} \) with domain \( D_k^{\text{comb}} = \{(x(i), y(1)_{(i)}, \ldots, y(k-1)_{(i)}, y(k)_{(i)})\}_{i=1}^N \), each \( M_k^{\text{comb}} \) is a model for the level \( k \) knowing predictions of other levels \( (1, \ldots, k-1) \). For example, to construct the combined model of the third level \( M_{L_3}^{\text{comb}} \), we use a priori information, the predictions of models \( M_{L_1} \) and \( M_{L_2} \). This approach allows
us to refine the models learned for each level, because we integrate a priori knowledge in our general process.

CRFs construct a model which characterizes the entities of input data. Each entity has a rich set of features that can help modeling (for example, a syntactical position is one feature of words in textual context). The entities (input data) are segmented in clusters, like word-sentences in text. In our case, we assume that the observation sequence boundaries are fixed and we segment the text transcription into sets of words. Each set represents what a given speaker have said. All sets represent dialogues between the different actors. In the learning process, we assign the label Other to the non-named entities. In our CRFs implementation, the data input are structured by line. Each line represents one word with its features. For example, the word Paris may have loc (localisation) and geo (geographical) as features. For this, we will rewrite the transcription data to construct the training set where each instance is represented by a set of words and each word has its specific features (step 1 in Figure 5). The labels used in the transcriptions corpus are spread over three levels: \( L_1, L_2, L_3 \). First, we built the simple models \( L_1, L_2, L_3 \) (step 2 in Figure 5). We used in the phase of testing the predictions of these models (step 3 in Figure 5) as an a priori data for testing the combined models \( L_1/L_2, L_1/(L_2, L_3) \), etc. (step 4 in Figure 5). Figure 6 shows the results of these experiments with different models. To measure the performance of each model, we use three valuation measures: the recall \( R = \frac{\text{nbr(correct)}}{\text{nbr(reference)}} \), the precision \( P = \frac{\text{nbr(correct)}}{\text{nbr(hypo.)}} \) and the \( F(\beta) \) measure \( = \frac{(1+\beta^2)PR}{\beta^2P+R} \).

5 Experiments

CRFs construct a model which characterizes the entities of input data. Each entity has a rich set of features that can help modeling (for example, a syntactical position is one feature of words in textual context). The entities (input data) are segmented in clusters, like word-sentences in text. In our case, we assume that the observation sequence boundaries are fixed and we segment the text transcription into sets of words. Each set represents what a given speaker have said. All sets represent dialogues between the different actors. In the learning process, we assign the label Other to the non-named entities. In our CRFs implementation, the data input are structured by line. Each line represents one word with its features. For example, the word Paris may have loc (localisation) and geo (geographical) as features. For this, we will rewrite the transcription data to construct the training set where each instance is represented by a set of words and each word has its specific features (step 1 in Figure 5). The labels used in the transcriptions corpus are spread over three levels: \( L_1, L_2, L_3 \). First, we built the simple models \( L_1, L_2, L_3 \) (step 2 in Figure 5). We used in the phase of testing the predictions of these models (step 3 in Figure 5) as an a priori data for testing the combined models \( L_1/L_2, L_1/(L_2, L_3) \), etc. (step 4 in Figure 5). Figure 6 shows the results of these experiments with different models. To measure the performance of each model, we use three valuation measures: the recall \( R = \frac{\text{nbr(correct)}}{\text{nbr(reference)}} \), the precision \( P = \frac{\text{nbr(correct)}}{\text{nbr(hypo.)}} \) and the \( F(\beta) \) measure \( = \frac{(1+\beta^2)PR}{\beta^2P+R} \).

![Figure 5. Process.](image)

**Figure 5. Process.**

**Figure 6. Simple and combined results for each conceptualisation level.**

It is to note that the results of experimental combined models represent an improvement from 1 to 3 points on the
$F(1)$ – measure compared to simple models (Figure 6). This leads us to use the concatenation of combined CRFs predictions (C-CRF) to rebuild the label with its three levels. The use of learning levels divides by 5 the learning CPU cost. This due to the diminution of annotation complexity. The CRF complexity go throw $O(L_1 + L_2 + L_3)$ in CRF to $O(L_1) + O(L_2) + O(L_3)$ in C-CRF with $L_k$ as the number of categories at level $k$.

<table>
<thead>
<tr>
<th>part</th>
<th>Rec</th>
<th>Prec</th>
<th>$F(1)$</th>
<th>Rec</th>
<th>Prec</th>
<th>$F(1)$</th>
<th>CPU cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMM</td>
<td>59.9</td>
<td>69.8</td>
<td>63.0</td>
<td>62.5</td>
<td>75.7</td>
<td>68.5</td>
<td>238 h 41</td>
</tr>
<tr>
<td>CRF</td>
<td>61.0</td>
<td>85.8</td>
<td>71.3</td>
<td>71.3</td>
<td>85.3</td>
<td>77.7</td>
<td>273 h 39</td>
</tr>
<tr>
<td>C-CRF</td>
<td>61.4</td>
<td>86.5</td>
<td>72.2</td>
<td>72.8</td>
<td>84.8</td>
<td>78.8</td>
<td>57 h 56</td>
</tr>
</tbody>
</table>

Table 3. Results.

In the evaluation step, we have used two samples test1 and test2. Samples are chosen in different times and different sources: In the first experimentation denoted test1, both the training and the testing data are taken from the same source (media) and concern the same period, whereas, for the test2, both the sources and the periods are different for training and testing data. The purpose of this dual testing is to show the influence of sources and periods on the prediction process. By comparing the results of test1 and test2, we can note that time axis affect negatively on the predictions. The $F(1) – measure$ of test1 is better than 6 points when the test data are provided from the same training data (Table 3).

6 Conclusion

We showed in this paper that linear CRFs overcomes HMM for NE retrieval in audio broadcast news, with a relative recall (respectively precision) gain against HMM of 14.2 % (resp. 12.7 %). Moreover, we showed that NEs recall can be enhanced taking advantage of the hierarchical structure of NEs, by combining best CRFs sublevel models. Thus we get an improvement of recall of 16.6 % against HMM, while the precision is nearly the same than CRFs with 12 % relative gain against HMM. This can be explained by the fact that hierarchical CRFs combine different levels of NE with equal precision, but generating NE labels at various levels, yielding then to a bigger recall (Figure 6 and table 2). Another advantage of hierarchical CRFs approach is the significant decrease of the CPU cost (nearly divided by a factor 5) to train the models, compared to the classic global CRFs approach. Finally, we note that combined CRFs, for a higher $F(1)$, is 5 times faster than CRFs.

Therefore, we demonstrated that in the field of semantic indexing, the conceptual representation surpasses its language representation. The NER task is very important in the chain of semantic processing of audio transcriptions. In this work, we showed the performance of graphical methods in solving the classification problem and more specifically the labeling problem. The use of structures in the learning process provides a gain in quality labeling, because it provides an a priori knowledge of the data description. We have shown that the use of hierarchical-concepts aspect improves classification and allows the application of heuristics approximation concepts.

References