# Structured and Extended Named Entity Evaluation in Automatic Speech Transcriptions

**Olivier Galibert**[1] **Sophie Rosset**[2] **Cyril Grouin**[2]
**Pierre Zweigenbaum**[2] **Ludovic Quintard**[1]
[1]LNE, Trappes, France [2] LIMSI–CNRS, Orsay, France
first.last@lne.fr first.last@limsi.fr

## Abstract

The evaluation of named entity recognition (NER) methods is an active field of research. This includes the recognition of named entities in speech transcripts. Evaluating NER systems on automatic speech recognition (ASR) output whereas human reference annotation was prepared on clean manual transcripts raises difficult alignment issues. These issues are emphasized when named entities are structured, as is the case in the Quaero NER challenge organized in 2010. This paper describes the structured named entity definition used in this challenge and presents a method to transfer reference annotations to ASR output. This method was used in the Quaero 2010 evaluation of extended named entity annotation on speech transcripts, whose results are given in the paper.

## 1 Introduction

Named Entity Detection has been studied since the MUC conferences in 1987. The notion has been extended to deal with mono- or multi-word expressions that belong to a potentially interesting class for an application. Given a set of entity definitions and a natural language corpus, systems try to extract and categorize all the relevant occurring entities. These entities can be used to feed further systems such as Information Retrieval, Question-Answering, Distillation, Terminology studies, etc.

Traditional Named Entity Recognition (NER) is a task where proper nouns and numerical expressions are extracted from documents and classified into categories (person, location, organization, date, etc.). As shown by Voorhees and Harman (2000), it is a key technology of Information Extraction (IE) and Open-Domain Question Answering. NER is also used as a fundamental component in a variety of language processing applications such as text clustering, topic detection, and summarization.

While significant progress has been reported on the NER task, most of these approaches have generally focused on clean textual data such as Sang and Meulder (2003). In the mean time, Kubala et al. (1998), Palmer et al. (1999), Turmo et al. (2009) and many others have focused on speech data. Named Entity detection evaluation over French spoken data has been proposed within the Ester II project, as described by Galliano et al. (2009).

Within the framework of the *Quaero* project,we proposed an extended named entity definition with compositional and hierarchical structure. This extension raises new issues and challenges in NER evaluation. First, as we shall explain below in more detail, the usual evaluation methods cannot compute the Slot Error Rate (SER) metric when named entities are compositional and recursive. Second, following Burger et al. (1998) and Hirschman et al. (1999), we consider that the evaluation of named entity recognition on noisy text output by automatic speech recognition (ASR) systems should take as reference the named entities found in the human annotation of a human-transcribed text: what *should have been there* in the ASR output. This requires to project the clean reference to the noisy text, which is made all the more difficult because of the compositional and hierarchical structure of the named entities.

The remainder of the paper is structured as follows. We first present the extended named entities in Section 2, then the evaluation protocol in Section 3 with specific metrics adapted to the structure of the evaluated objects and data. Section 4 illustrates their use in a challenge and discusses system results in this challenge. Finally in Section 5 we conclude and draw perspectives for further work.



*Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 518–526,
Chiang Mai, Thailand, November 8 – 13, 2011. 

## 2 Extended Named Entities

In this section, we present our extension to named entities, starting with related work (Section 2.1) and specifying their scope (Section 2.2). Our entities are hierarchical (Section 2.3) and compositional (Section 2.4). Section 2.5 provides a discussion of the issues they raise in the evaluation of named entity recognition from speech transcripts.

### 2.1 Named Entity Types

Named Entity recognition was initially defined as recognizing proper names (Coates-Stephens, 1992). Since MUC-6 (Grishman and Sundheim, 1996), named entities are proper names categorized into three major classes: persons, locations and organizations. Proposals have been made to sub-divide these entities into finer-grained classes. For example, politicians for the person class by Fleischman and Hovy (2002) or cities for the location class by Fleischman (2001) as well as Lee and Lee (2005).

The CONLL conference added a miscellaneous type which includes proper names outside the previous classes. Some classes are sometimes added, e.g. product by Bick (2004). Some numerical types are also often described and used in the literature: date, time, and amounts (money and percents in most cases).

Specific entities have been proposed and handled for some tasks, e.g. language and shape by Rosset et al. (2007), or email address and phone number (Maynard et al., 2001). In specific domains, entities such as gene, protein, DNA etc. are also addressed (Ohta, 2002) and campaigns are organized for gene/protein detection (Kim et al., 2004; Galibert et al., 2010)). More recently larger extensions have been proposed: Sekine (2004) defined a complete hierarchy of named entities containing about 200 types.

### 2.2 Scope

Named Entities often include four major groups: name, quantity, date and duration. The overall task in which we frame information extraction is the extraction of entities and relations to build a fact base from news sources. We thus decided to start from the traditional named entities used in information extraction from newspaper corpora. We then included named entities extensions proposed by Sekine (2004) for products and Galliano et al. (2009) for functions, and we extended the definition of named entities to some expressions which are not composed of proper names (e.g., phrases built around substantives).

In this work, we decided to extend the coverage of the named entities rather than sub-dividing the existing classes as it has been done in previous work. As we aimed to build a fact database from news data, we chose to support new kinds of entities (time, function, etc.) in order to extract a maximum of information from the corpus we processed. Compared to existing named entity structuration, our approach is more general than the extensions that have been done for specific domains, and is simpler than the complete hierarchy defined by Sekine (2004). This structure allows us to cover a large amount of named entities with a basic categorization so as to be quickly suitable for all further annotation work. The extended named entities we defined are both hierarchical and compositional (Grouin et al., 2011). This hierarchical and compositional nature of the extended named entities imply a specific method when evaluating system outputs (see Section 3).

### 2.3 Hierarchy

We used two kinds of elements: types and components. The types with their subtypes categorize a named entity. While types and subtypes were used previously, we consider that structuring the contents of an entity (its components) is important too. Components categorize the elements inside a named entity.

**Types and subtypes** refer to the general category of a named entity. They give general information about the annotated expression. The taxonomy is composed of 7 types and 32 sub-types:

- Person: *pers.ind* (invidual person), *pers.coll* (collectivity of persons);
- Location: administrative (*loc.adm.town, loc.adm.reg, loc.adm.nat, loc.adm.sup*) and physical (*loc.phys.geo, loc.phys.hydro, loc.phys.astro*);
- Organization: *org.ent* (services), *org.adm* (administration);
- Amount: quantity (with unit or general object), duration;
- Time: date *time.date.abs* (absolute date: "November 8, 2011"), *time.date.rel* (date relative to the discourse: "yesterday"), and hour

*time.hour.abs* (absolute hour), *time.hour.rel* (hour relative to the discourse);

- Production: *prod.object* (manufactury object), *prod.art* (artistic products), *prod.media* (media products), *prod.fin* (financial products), *prod.soft* (software), *prod.award, prod.serv* (transportation route), *prod.doctr* (doctrine), *prod.rule* (law);
- Functions: *func.ind* (individual function), *func.coll* (collectivity of functions).

Types and subtypes constitute the first level of annotation. They refer to a general segmentation of the world into major categories. Within these categories, we defined a second level of annotation we call *components*.

**Components** can be considered as clues that help the annotator (human or system) to produce an annotation: either to determine the named entity type (e.g. a first name is a clue for the *pers.ind* named entity subtype), or to set the named entity boundaries (e.g. a given token is a clue for the named entity, and is within its scope, while the next token is not a clue and is outside its scope). Components are second-level elements, and can never be used outside the scope of a type or subtype element.

An entity is thus composed of components that are of two kinds: transverse components that can fit each type of entity, and specific components only used for a reduce set of components:

1. Transverse components
   - *name* (the entity name),
   - *kind* (hypernym of the entity),
   - *qualifier* (a qualifying adjective),
   - *demonym* (inhabitant or ethnic group name),
   - *val* (a number),
   - *unit* (a unit),
   - *object* (an object),
   - *range-mark* (a range between two values).
2. Specific components
   - *name.last, name.first, name.middle, title* for “pers.ind" (Figure 1),
   - *address.number, po-box, zip-code, other-address-component* for “loc.add.phys",
   - and *week, day, month, year, century, millenium, reference-era, time-modifier* for “time.date" (Figure 3).

## 2.4 Composition

During the Ester II evaluation campaign, there was an attempt to use compositionality in named entities for two categories (persons and functions) where a person entity could contain a function entity.[1] Nevertheless, the evaluation did not take into account this inclusion and only focused on the encompassing annotation.[2]

In the present work, we also considered the compositional nature of those extended named entities. Entities can be compositional for three reasons:

1. a type contains a component: the *pers.ind* type is composed of several components such as *name.first* and *name.last* (Figure 1);

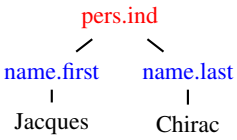


Figure 1: Basic type and component inclusion.

2. a type includes another type, used as a component. Cases of inclusion can be found in the *function* type (Figure 2), where type *func.ind*, which spans the whole expression, includes type *org.adm*, which spans the single word *Budget*:

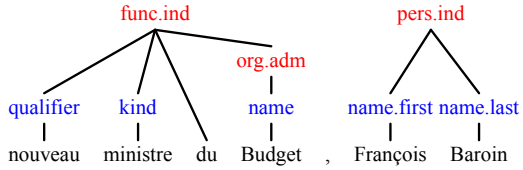


Figure 2: Multi-level annotation of entity types (red tags) and components (blue tags): *new minister of budget , François Baroin*.

[1] Example of compositionality in Ester II campaign: *<pers.hum> <func.pol> président </func.pol> <pers.hum> Chirac </pers.hum> </pers.hum>*

[2] Final annotation: *<pers.hum> président Chirac </pers.hum>*

3. in cases of metonymy (a term is substituted for another one in a relation of contiguity) and antonomasia (a proper name is used as a substantive and vice versa), where a type of entity is used to refer to another entity type (Figure 3). The type to which the entity intrinsically belongs is annotated. This entity is over-annotated with the type to which the expression belongs in the considered context:

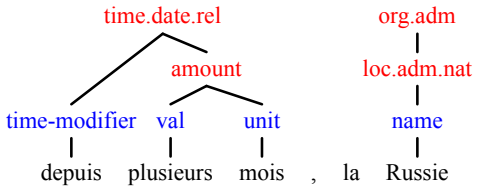


Figure 3: Annotation with types (red tags) and components (blue tags) including metonymy: *since several months , Russia. . .*

### 2.5 Discussion

Due to its non-flat structure, the representation of compositionality in extended named entities is richer than that used so far in spoken language understanding, compared with Bonneau-Maynard et al. (2006) and Mori et al. (2008); due to its extended definition, it is also richer than that used in named entity detection, in contrast with Galliano et al. (2009) or Nadeau and Sekine (2007). This calls for novel ways to evaluate named entity detection systems.

A consequence of the representation's richer structure is an increased complexity in the evaluation methodology. The 1:1 comparisons applied to traditional, flat named entities must give way to the mapping-based approaches we will present in the next section.

Moreover, when working on speech, evaluating the results of systems applied to automatic speech transcriptions is central to real-world use cases. This leads us to the issue of evaluating named entity detection systems applied to noisy inputs (produced by automatic speech recognition systems) using references built on clean data (manual transcriptions). The reference projection approach we propose will be described in the second half of the next section.

## 3 Evaluation methodology

We now come to the issues raised by the evaluation of automatically annotated extended named entities in speech transcripts. We first lay out the basic evaluation metrics (Section 3.1), then address the issues raised by compositionality (Section 3.2) and by ASR output errors (Section 3.3).

### 3.1 Metrics

The metrics used in Named Entity extraction evaluation are precision ($P$), recall ($R$), and their weighted mean F-measure ($F$) (van Rijsbergen, 1979). They are easy to use, since they only require to determine whether a hypothesized entity element is correct or not.

Let $Ref$ = total number of elements in the reference, $Hyp$ = total number of elements in the hypothesis, and $C$ = number of correct elements in the hypothesis. Precision is defined as the observed probability for a hypothesized element to be correct:

$$P = \frac{C}{Hyp}$$

In the same way, recall is the observed probability for a reference element to have been found:

$$R = \frac{C}{Ref}$$

F-measure is the weighted harmonic mean of $P$ and $R$, generally balanced with $\beta = 1$:

$$F = \frac{(1+\beta^2)RP}{\beta^2 P + R}$$

The main issue in these metrics lies in their binary decision process: either an entity element is correct, or it is not, whereas we generally want finer control.

Errors in named entities are in fact bidimensional: their span or their type can be incorrect. It is interesting to count only "half an error" if one of the two is correct. Within each category, some errors can be considered as less severe than others (e.g., presence of a determiner in span errors, entity types with fuzzy boundaries in the annotation guide).

A popular alternative is to proceed with an *error enumeration* approach, such as the Slot Error Rate (SER) defined by Makhoul et al. (1999): collect the individual errors, sum a cost for each one and divide the total by the number of elements in the reference (the slots). In our case, we went

for a simple weighting scheme where insertions ($I$), deletions ($D$) and elements with errors both in span and in type ($S_{ST}$) cost 1, while elements with errors only in either span ($S_S$) or type ($S_T$) cost 0.5. Span or type errors are substitutions ($S_S$, $S_T$, and $S_{ST}$).

We chose our final score as:

$$\text{SER} = \frac{D + I + S_{ST} + 0.5 \times (S_S + S_T)}{Ref}$$

Dividing by $Ref$ normalizes the result, allowing us to compare results more easily across different files. This value is traditionally expressed as a percentage.

### 3.2 Evaluation on manual transcriptions

For simple annotation guides with no compositionality, enumerating all errors is easy: a word can only be associated with at most one entity in the reference, and likewise in the hypothesis, so entities can be directly compared when they have common words without any ambiguity.

In our case, the annotation compositionality makes things harder. Entity elements (entities or components) can be nested, and words are usually associated to at least two elements: one entity and one component, and sometimes more. The enumeration phase needs to establish explicitly which hypothesis element should be compared with each reference element.

Building on methodologies used in speech diarization evaluation (Diarization Error Rate), we defined a *mapping* as a set of 1–1 associations between reference and hypothesis elements. Each element from one side can be associated to at most one from the other side, and a number of elements can remain unassociated on both sides. From a given mapping, an error list can be built, where associated elements result in either correct matches or substitutions, and unassociated elements result in insertions and deletions. Hence given a mapping, a score can be computed. The final score of a system is then defined as the minimal error rate attainable over all possible mappings.

Enumerating all possible mappings is unthinkable. Since the score is additive, and restricting the acceptable associations to elements with at least one common word, it becomes possible to apply a dynamic programming approach. We thus use a variant of the Viterbi algorithm where "time" is the word stream, "probability" the score and "hidden state" the associations.

The text is split into a series of segments cut where reference or hypothesis entities start or end. An empty association hypothesis is first created, then segments are handled in the text order.

Two phases are applied for each segment: the first one, *opening*, expands the association hypothesis set by combining each one with every possible association choice for each of the entities beginning at the segment start. Two constraints are applied at this level: an entity can only have zero or one association, and associations must not cross one another (i.e. parent-descendant links between entities must not be inverted when projected through the association set).

Once all hypotheses are built, the *closure* phase follows where ending entities are taken into account. The post-segment state of each association hypothesis is computed, including its score, and for every set of equivalent hypotheses in a dynamic programming sense, only the best score is kept.

Two association hypotheses are considered equivalent if, for every entity present in both closing and following segments, the specific hypothesized associations are identical. We underline that where no entity is present in the text (reference or hypothesis), only one association hypothesis is left. The same happens at the end of the text where the remaining association is the optimal one.

### 3.3 Generalization to automatic transcriptions

The main issue when evaluating a Named Entity extraction system over Automatic Speech Recognition (ASR) systems output is: what must be evaluated first? We can either evaluate *what is there* (the system annotation) or *what should have been there*.

A system should not be penalized for missing things that have been lost earlier in the pipeline, or extracting entities that were not actually said but are present in ASR output. This leads us to an evaluation methodology equivalent to that of manual transcriptions.

The ASR output is just considered as a distinct, independent document, to be annotated by humans and by systems, and the results are compared. The human annotation part becomes way more difficult. It is quite hard to annotate documents in which parts make no sense, where adjudication discussions can become endless. More sig-

nificantly for an application, ASR is a step in the document handling pipeline where the end user is only interested in the final result.

We thus decided to evaluate system performance compared to *what should have been there*, expecting the systems to find the entities present in the manual transcriptions whatever the quality of the ASR output. There is room thus for developers to try methodologies to cope with ASR errors using a higher-level understanding of annotations.

**Reference projection.** To evaluate system output from noisy text with a reference built from a clean text, we followed Burger et al. (1998) and Hirschman et al. (1999) who proposed to *project* the clean reference on the noisy text in order to build a new reference. That new reference then allows us to apply the clean text methodology.

This projection method consists in finding new positions for the frontiers through either a dynamic programming alignment (standard sclite-type ASR evaluation alignment) or a phone-level dynamic programming alignment using canonical phonetizations. They noticed the result was too strict frontier-wise and required reducing the weight of frontier errors to obtain significant results.

In Question Answering from speech transcripts evaluation, Moreau et al. (2010) required that QA systems extract answers to natural language questions from ASR outputs of broadcast news shows. The inherent application was to replay the sound segment containing the answer, with a time interval as an answer; it tolerated a time interval around the boundaries. The results were satisfactory.

We thus decided to project the clean reference on the noisy text following five steps:

1. Build a forced alignment of the reference text to the speech signal;

2. Extract the start and end times from the reference annotations using the alignment;

3. Select a tolerance time interval;

4. Find the ASR word frontiers within the tolerance intervals placed around the frontiers of reference entity elements;

5. Build a *fuzzy* reference when multiple frontiers are possible.

A fuzzy reference means that each reference element can have multiple frontiers, which is equivalent to having multiple references, and choosing the one that gives the best score for the system. The number of possible references is way too large and the enumeration has to be done locally and coupled with the Viterbi alignment to reduce the search space to tolerable limits.

Apart from the alignment algorithm, the main difficulty is that a structurally correct reference post-projection does not always exist. Indeed, the ASR system may not output words where an entity element is supposed to be, or may merge small words into a larger one, preventing from fitting all reference elements to the available words. Such colliding elements have to be handled and we decided when encountered to remove one arbitrarily. They are rare enough for that decision to have a minor impact on the scores.

A more satisfactory method would be to merge the colliding elements into one when possible, creating reference elements with multiple acceptable types. Figure 4 illustrates the results of building a fuzzy reference.

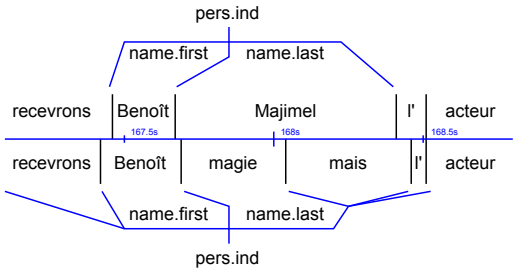


Figure 4: Example of a fuzzy reference built by temporal alignment of clean reference and noisy transcript

The top part of the figure shows the manual reference (clean reference), with a *pers.ind* entity "Benoît Magimel" which is decomposed into two components *name.first* "Benoît" and *name.last* "Magimel". In the middle, the temporal line shows the results of the forced alignment of these words on the audio signal.

In the lower part of the figure is the ASR result, "recevrons Benoît magie mais l' acteur", with its own temporal positions as given by the ASR system itself. Accepting any frontier within an interval then gives us the final fuzzy reference at the bottom, where the *name.first* component and the associated *pers.ind* can either start before "re-

cevrons" or just after, the *name.first—name.last* transition still happens after "Benoït", and the *name.last* and complete entity can stop either after "magie", "mais" or "l"'.

In our case, all systems gave the same hypothesis with "Benoït" as both *pers.ind* and *name.first*, which gave them one correct (the *name.first*), one bad frontier (the *pers.ind*) and one miss (the *name.last*). A more advanced system could have used "Benoït" as a trigger to search for "Magimel", and other last names associated with that specific first name, in a phonetic representation of the following words, or in the signal itself. Then, it may have output "magie mais" or "magie mais l"' as the last name. It is interesting to note that either hypothesis would have ended with a perfect score for the system. That example shows how the fuzzy projection methodology opens the door to the evaluation of more advanced systems that try to use higher-level knowledge to see through the ASR system errors.

An interesting and useful side effect of the mapping methodology we used is that the chosen mapping is human-readable. One can check what associations were chosen and in the ASR case what frontiers were selected among the possible ones. This is useful for both error analysis and convincing oneself of the quality of the evaluation measure. It also makes it possible to merge all of the systems outputs in an evaluation and collate the errors in order to help correct the references more efficiently where needed.

## 4 Quaero evaluation results

As an illustration of the use of the extended named entities and of the evaluation methods introduced above, we present here an evaluation of extended named entity recognition from speech transcripts, which we organized in the context of the project Quaero. The task consisted in extracting and categorizing a large number of named entities in transcriptions of broadcast spoken data in French. Three teams participated, each with one NER system.

The training data were those of the Ester II evaluation campaign: 188 shows from various sources have been manually transcribed and annotated given this new definition of extended named entities (Table 1, Training). The test data were composed of test and development data from the Quaero 2010 ASR evaluation (Table 1, Test) (Lamel, 2010). Several test data versions were provided:

- a manual transcription prepared by a human expert,
- three different ASR outputs (ASR1, ASR2, ASR3) with a word error rate (WER) ranging from 20.96% to 27.44% (Table 1, last three rows), and
- an improved version of the ASR1 output, where punctuation and capitalization have been automatically added (ASR1+).

The training data consisted of Broadcast News data (BN) while the test data included Broadcast News data and more varied data including talk shows and debates (Broadcast Conversations, BC; see Table 1, last two columns). One of the objectives of this work was to measure the robustness of the NER systems against different types of data and unknown types of data.

| Inf. \ Data | Training | Test | Test BN | Test BC |
|---|---|---|---|---|
| # shows | 188 | 18 | 8 | 10 |
| # lines | 43289 | 5637 | 1704 | 3933 |
| # distinct words | 39639 | 10139 | 5591 | 6836 |
| # words | 1251586 | 97871 | 32951 | 64920 |
| # types | 113885 | 5523 | 2762 | 2761 |
| # distinct types | 41 | 32 | 28 | 29 |
| # compon. | 146405 | 8902 | 4541 | 4361 |
| # distinct compon. | 29 | 22 | 22 | 21 |
| WER ASR1 | – | 20.96% | 16.32% | 23.34% |
| WER ASR2 | – | 21.56% | 18.77% | 22.99% |
| WER ASR3 | – | 27.44% | 24.06% | 29.18% |

Table 1: Data description.

Table 2 shows the results of the three participating NER systems, with a breakdown into broadcast news and broadcast conversations.

On the manual transcriptions, values of slot error rate (SER) ranged from 33.3% to 48.9% for the NER systems on the whole data. Similarly to the ASR systems, NER systems obtained better SER (from 29.7% to 42.7%) on broadcast news than on broadcast conversations (37% to 55.3%).

| | Whole data | | | | |
|---|---|---|---|---|---|
| | Man. | ASR1 | ASR1+ | ASR2 | ASR3 |
| P1 | 48.9% | 71.4% | 71.1% | 68.3% | 75.2% |
| P2 | 33.3% | 61.1% | 66.3% | 59.3% | 63.2% |
| P3 | 41.0% | 72.2% | 68.7% | 70.7% | 72.9% |
| | Broadcast News data | | | | |
| P1 | 42.7% | 55.3% | 52.7% | 58.5% | 61.4% |
| P2 | 29.7% | 48.5% | 53.8% | 52.2% | 53.5% |
| P3 | 39.1% | 55.6% | 54.5% | 60.3% | 61.8% |
| | Broadcast Conversations data | | | | |
| P1 | 55.3% | 87.9% | 89.9% | 78.3% | 89.2% |
| P2 | 37.0% | 73.9% | 79.0% | 66.6% | 73.0% |
| P3 | 43.0% | 89.3% | 83.3% | 81.2% | 84.1% |

Table 2: SER results for the overall data, broadcast news data and broadcast conversations data. The ASR1+ column is a version of the ASR1 with automatically added punctuation and capitalization.

Obviously, the SER worsened when dealing with ASR outputs, which are all true case (*i.e.*, upper and lower case are those expected in normal text). The loss ranged from 19.4% (P1 on ASR2 with a 21.56% WER) to 33% (P2 on ASR1+ with 20.96% WER). It is interesting to note that the ASR1+ system, which is ASR1 with automatically added punctuation and capitalization at the beginning of sentences, hindered system P2.

Another interesting point is that the ASR2 output with a higher WER than the ASR1 system on the whole data allowed better performance for entity detection than the ASR1 output.

## 5 Conclusion and perspectives

In this paper, we presented a representation of structured named entities, and methods to evaluate the recognition of such structured named entities. We contributed a mapping between reference and hypothesis elements which allows us to enumerate errors and compute the value of the slot error rate. We also provided a projection of extended named entities from a clean reference to noisy texts produced by automatic speech transcription systems, which allows us to compute the slot error rate against what was actually said rather than against what was recognized by the ASR systems.

These extended named entities and evaluation algorithms have been used in the Quaero Named Entities on Spoken Data evaluation. Evaluation results are consistent with expectations, which is a first test of the validity of the method. Indeed, further work is planned to study more closely and more systematically the obtained alignments.

Compared to the recognition of standard named entities, this new task is harder for systems, but this new structuring will be useful to make information extraction more precise. Moreover, the evaluation methodology we proposed is very flexible and should be usable for other tasks such as syntactic analysis on spoken data. An interesting and useful side-effect of this mapping methodology is its human readability. This makes it easier to check the chosen associations, as well as the selected frontiers in the ASR case.

This work is useful for both error analysis and convincing oneself of the evaluation measure quality. It also makes it possible to merge all systems outputs in an evaluation and collate the errors to help correct the reference more efficiently where needed.

Due to the scarcity of annotated corpora in named entities, we plan to provide both guidelines and annotated corpora for free to the scientific community.

## Acknowledgments

This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation and by the French ANR ETAPE project.

## References

Eckhard Bick. 2004. A named entity recognizer for Danish. In *Proc. of LREC*, Lisbon, Portugal.

Hélène Bonneau-Maynard, Christelle Ayache, Frédéric Bechet, Alexandre Denis, Anne Kuhn, Fabrice Lefèvre, Djamel Mostefa, Mathieu Quignard, Sophie Rosset, Christophe Servan, and Joanne Villaneau. 2006. Results of the French Evalda-Media evaluation campaign for literal understanding. In *Proc. of LREC*, pages 2054–2059, Genoa, May.

John D. Burger, David Palmer, and Lynette Hirschman. 1998. Named entity scoring for speech input. In *Proc. of COLING*, pages 201–205.

Sam Coates-Stephens. 1992. The analysis and acquisition of proper names for the understanding of free text. *Computers and the Humanities*, 26:441–456.

Michael Fleischman and Eduard Hovy. 2002. Fine grained classification of named entities. In *Proc. of COLING*, pages 1–7. Association for Computational Linguistics.

Michael Fleischman. 2001. Automated subcategorization of named entities. In *Proc. of the ACL 2001 Student Research Workshop*, pages 25–30.

Olivier Galibert, Ludovic Quintard, Sophie Rosset, Pierre Zweigenbaum, Claire Nédellec, Sophie Aubin, Laurent Gillard, Jean-Pierre Raysz, Delphine Pois, Xavier Tannier, Louise Deléger, and Dominique Laurent. 2010. Named and specific entity detection in varied data: The Quaero named entity baseline evaluation. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proc. of LREC*, Valletta, Malta, may. European Language Resources Association (ELRA).

Sylvain Galliano, Guillaume Gravier, and Laura Chaubard. 2009. The ESTER 2 evaluation campaign for the rich transcription of French radio broadcasts. In *Proc. of InterSpeech*.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference - 6: A brief history. In *Proc. of COLING*, pages 466–471, Copenhagen, Denmark, August.

Cyril Grouin, Sophie Rosset, Pierre Zweigenbaum, Karën Fort, Olivier Galibert, and Ludovic Quintard. 2011. Proposal for an extension of traditional named entities: From guidelines to evaluation, an overview. In *Proc. of the Fifth Linguistic Annotation Workshop (LAW-V)*, Portland, OR, june. Association for Computational Linguistics.

Lynette Hirschman, John Burger, David Palmer, and Patricia Robinson. 1999. Evaluating content extraction from audio sources. In *ECSA, ETRW Workshop: Accessing Infomation in Spoken Audio*. University Press.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, and Yuka Tateisi and. Nigel Collier. 2004. Introduction to the Bio-Entity task at JNLPBA. In *BioCreative Challenge Evaluation Workshop*, Granada, Spain.

Francis Kubala, Richard Schwartz, Rebecca Stone, and Ralph Weischede. 1998. Named entity extraction from speech. In *Proc. of the DARPA Broadcast News Transcription and Understanding Workshop*.

Lori Lamel. 2010. Quaero Program - CTC Project - Progress Report on Task 5.1: Speech to Text. Technical Report CD.CTC.5.6, Quaero Program.

Seungwoo Lee and Gary Geunbae Lee. 2005. Heuristic methods for reducing errors of geographic named entities learned by bootstrapping. In *Proc. of IJCNLP*, pages 658–669.

John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel. 1999. Performance measures for information extraction. In *Proc. of DARPA Broadcast News Workshop*, pages 249–252.

Diana Maynard, Valentin Tablan, Cristian Ursu, Hamish Cunningham, and Yorick Wilks. 2001. Named entity recognition from diverse text types. In *Proc. RANLP*, Tzigov Chark.

Nicolas Moreau, Olivier Hamon, Djamel Mostefa, Sophie Rosset, Olivier Galibert, Lori Lamel, Jordi Turmo, Pere R. Comas, Paolo Rosso, Davide Buscaldi, and Khalid Choukri. 2010. Evaluation protocol and tools for question-answering on speech transcripts. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proc. of LREC*, Valletta, Malta, may. European Language Resources Association (ELRA).

Renato De Mori, Frédéric Bechet, Dilek Z. Hakkani-Tür, Michael McTear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding. *IEEE Signal Processing Magazine*, 25(3):50–58, May.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Linguisticæ Investigationes*, 30(1):3–26.

Tomoko Ohta. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proc. of the Human Language Technology Conference*, pages 73–77.

David D. Palmer, John D. Burger, and Mari Ostendorf. 1999. Information extraction from broadcast news speech data. In *Proc. of the DARPA Broadcast News Workshop*.

Sophie Rosset, Olivier Galibert, Gilles Adda, and Eric Bilinski. 2007. The LIMSI participation to the QAst track. In *Working Notes for the CLEF 2007 Workshop*, Budapest, Hungary, September.

Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proc. of CoNLL*.

Satoshi Sekine. 2004. Definition, dictionaries and tagger of extended named entity hierarchy. In *Proc. of LREC*, Lisbon, Portugal.

Jordi Turmo, Pere R. Comas, Sophie Rosset, Olivier Galibert, Nicolas Moreau, Djamel Mostefa, Paolo Rosso, and Davide Buscaldi. 2009. Overview of QAST 2009 - question answering on speech transcriptions. In *CLEF 2009 Working Notes*, Corfu, Greece.

Cornelis Joost van Rijsbergen. 1979. *Information Retrieval*. Butterworths, London. `http://www.dcs.gla.ac.uk/Keith/Preface.html`.

Ellen M. Voorhees and Donna Harman. 2000. Overview of the ninth text retrieval conference. In *Proc. of TREC-9*.