Pattern Mining for Named Entity Recognition

Damien Nouvel Jean-Yves Antoine Nathalie Friburger
Université François Rabelais Tours, Laboratoire d'Informatique
3, place Jean Jaures, 41000 Blois, FRANCE
{damien.nouvel, jean-yves.antoine, nathalie.friburger} @univ-tours.fr

Abstract

Many evaluation campaigns have shown that knowledge-based and data-driven approaches remain equally competitive for Named Entity Recognition. Our research team has developed CasEN, a symbolic system based on finite state tranducers, which achieved promising results during the Ester2 French-speaking evaluation campaign. Despite these encouraging results, manually extending the coverage of such a hand-crafted system is a difficult task. In this paper, we present a novel approach based on pattern mining for NER and to supplement our system's knowledge base. The system, mXS, exhaustively searches for hierarchical sequential patterns, that aim at detecting Named Entity boundaries. We assess their efficiency by using such patterns in a standalone mode and in combination with our existing system.

1 Introduction

Named Entity Recognition (NER) is an information extraction task that aims at extracting and categorizing specific entities (proper names or dedicated linguistic units as time expressions, amounts, etc.) in texts. These texts can be produced in diverse conditions. In particular, they may correspond to either electronic written documents [10] or more recently speech transcripts provided by a human expert or an automatic speech recognition (ASR) system [7]. The recognized entities may later be used by higher-level tasks for different purposes such as Information Retrieval or Open-Domain Question-Answering [21]. While NER is often considered as quite a simple task, there is still room for improvement when it is confronted to difficult contexts. For instance, NER systems may have to cope with noisy data such as speech recognition errors or speech disfluences. In addition, NER is no more circumscribed to proper names, but may also involve common nouns (e.g., "the judge") or complex multi-word expressions with embedded NEs (e.g. "the Computer Science Department of the New York University"). These complementary needs for robust and detailed processing explain that knowledge-based and data-driven approaches remain equally competitive on NER tasks as shown by many evaluation campaigns. For instance, the French-speaking Ester2 and Etape evaluation campaigns on radio broadcasts [7] has shown that knowledge-based approaches outperformed data-driven ones on manual transcriptions.

However, despite their advantageous precision, symbolic systems need significant efforts when confronted to new NE types or when the system has to be adadpted to diverse mdalities (written vs oral transcripts). In this paper, we present an original approach, based on the adaptation of pattern mining techniques as a machine learning process (automatic training on corpora to reach a large coverage), while remaining in the framework of symbolic resources (extraction of intelligible rules of NE recognition). The performances of the resulting system (mXs) on the Etape French-speaking evaluation compaign shows that this novel approach bears comparison with standard machine learning techniques (CRFs). Besides, coupling this system with CasEN [5], our knowledge-based system, provides us with promising results.

In Sect. 2 we present and compare approaches for NER. Sections 3 and 4, we describe how lexico-syntactic patterns may be extracted from annotated corpora and used as a standalone system. Finally, Sect. 5 and 6 reports experimental results on French oral corpora.

2 Related Work

In the 90's and until now, several symbolic systems have been designed that, often, make intensive use of regular expressions formalism to describe NEs. Those systems often combine external and internal evidences [12], as patterns describing contextual clues and lists of proper names by NE categories. Those systems achieve high accuracy, but, as stated by [13], because they depend on the hand-crafted definition of lexical ressources and detection rules, their coverage remains an issue.

Machine learning introduced new approaches to address NER. The problem is then stated as categorizing words that belong to a NE, taking into account various clues (features) in a model that is automatically parametrized by leveraging statistics from a *training corpus*. Among these methods, some only focus on the current word under examination (maximum entropy, SVM) [1], while others also evaluate stochastic dependencies (HMM, CRF) [11]. Most of the time, these approaches output the most probable sequence of labels for a given sentence. This is generally known as the "labeling problem", applied to NER.

Many approaches [14] rely on pre-processing steps that provide additional information about data, often Part-Of-speech (POS) tagging and proper names lists, to determine how to automatically tag a text, resulting in an annotated text. Some make use of data mining techniques [4, 3], but we are not aware of work that goes beyond the step of extracting patterns for NER: no model has emerged for using those patterns to recognize NEs.

In this paper, we propose a system that adapts text mining techniques to the NER problem. The benefits of text mining techniques are twofold. On the one hand, pattern mining techniques are data driven and may be combined with standard machine learning approaches. On the other hand, pattern mining allows to extract NER detection rules (e.g. transducers) which are intelligible for a human expert and can be used by a symbolic system. To the best of our knowledge, this way of combining symbolic and machine learning approaches is completely original in the framework of NER.

Besides, our pattern mining system, mXS, focuses on boundaries of NEs, as *begin-*

$\mathcal{D}$	
Sent.	Patterns from $\mathcal{L}_I$
s_1	The american <pers> president Barack Obama </pers> has arrived in <loc> Moscou </loc>.
s_2	There he has seen the former <pers> chancelor Michelle Bachelet </pers>.
s_3	The <pers> president Dimitri Medvedev </pers> was not present on the beautiful <loc> square Vladimir Lenine </loc>.

Table 1: sentences from an annotated corpus

ning or ending markers that we would like to be inserted at correct positions. To this end, we extract patterns [17, 15] that are correlated to those markers. Those patterns, casted as "annotation rules", are not constrained to necessarily recognize both boundaries of NEs. Basically, the system detects each boundary of NEs separately. This strategy is expected to present a more robust behaviour on noisy data such as ASR recognition errors or speech disfluences. They are evaluated as a standalone system or coupled with our existing knowledge-based system.

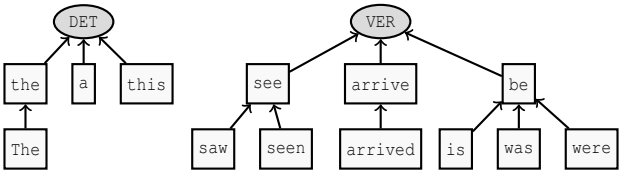
3 Mining Hierarchical and Sequential Patterns

3.1 Extracting Patterns

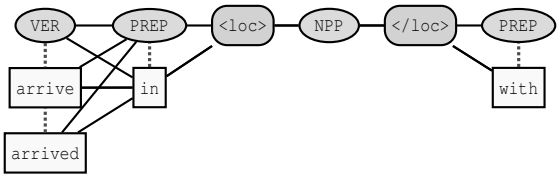
We use data mining techniques to process natural language. In this context, what is detected as a sentence will be considered as a sequence of items, precluding the extraction of patterns accross sentences. Two alphabets are defined: $\mathcal{W}$, words from natural language, and $\mathcal{M}$ as *markers*, e.g. the tags delimiting NE categories (e.g. person, location, amount). The annotated corpus $\mathcal{D}$ is a multiset of sequences based on items from $\mathcal{W} \cup \mathcal{M}$. Table 1 exemplifies this with $\mathcal{W} = \{\texttt{The}, \texttt{president}, \texttt{Obama}, \dots\}$ and $\mathcal{M} = \{\texttt{<pers>}, \texttt{</pers>}, \texttt{<loc>}, \texttt{</loc>}, \texttt{<time>}, \dots, \texttt{<org>}, \dots\}$.

Like most systems, the mining process relies as a first step on linguistic analysis of input data. Those preprocessing steps extend the language $\mathcal{W}$ to $\mathcal{W}^*$ by lemmatizing, applying a Part-Of-Speech (POS) tagger and recognizing expressions from lexical resources provided by the ProlexBase [2] database (890K entries). Those additional elements are inserted as a hierarchical representation of tokens: each may gradually be generalized to its lemma, POS or semantic type. For instance in Table 1, the pattern language contains items $\{\texttt{arrive}, \texttt{see}, \texttt{VER}, \texttt{JJ}, \texttt{DET}, \texttt{NN}, \texttt{NPP} \dots\}$. The POS tagger distinguishes common nouns (NN) from proper names (NPP). Note that we only keep semantic information for proper names, to avoid extracting patterns that would contain instances of proper names. Figure 1 illustrates how POS categories are organized as a hierarchy and what patterns may be mined through an example of a sequence.

We exhaustively extract contiguous patterns over this language. For instance, in Fig. 1, patterns such as '`VER in <loc> NPP`' or '`NPP </loc> with`' are extracted. The hierarchy and the properties of sequential patterns allow to partially order them. This



(a) Excerpt of hierarchy



(b) Possible paths to mine patterns for the example sentence "arrived in <loc> Moscou </loc>"

Figure 1: POS hierarchy and example of tagged sentence

use of the hierarchy as a modular description of language enables fine-grained generalizations for items inside patterns: extracting accurate patterns relies both on the data that is mined and on specifying a relevant hierarchy for NER.

3.2 Filtering Patterns as Annotation Rules

We mine a large annotated corpus to find generalized patterns that co-occur with NE markers. As usual in data mining, we set thresholds during extraction based on two interestingness measures: support and confidence. The *support* of a pattern P is its number of occurrences in $\mathcal{D}$, denoted by $supp(P,\mathcal{D})$. The greater the support of P, the more general the pattern P. To estimate empirically how much P is accurate to detect markers, we calculate its *confidence*. A function $suppNoMark(P)$ returns the support of P when markers are omitted both in the rule and in the data. Thus, the confidence of P is:

$$conf(P,\mathcal{D}) = \frac{supp(P,\mathcal{D})}{suppNoMark(P,\mathcal{D})} \quad (1)$$

As we are only interested in patterns correlated with NE markers, we extract patterns containing at least one marker as rules. For instance, consider the rule $R =$ 'the JJ <pers> NN NPP' in Table 1. Its support is 2 (sentences s_1 and s_2). But its support without considering markers is 3, since sentence s_3 matches the rule when markers (<pers> in rule and <loc> in "the beautiful <loc> square Vladimir") are omitted. Thus the confidence of R is only 2/3.

The collection of transduction rules exceeding minimal support and confidence thresholds is used as a knowledge-base. In practice, the number of discovered rules remains very large (especially when minimal support threshold is low). Thus, we decide to filter-out the redundant rules. We consider two rules to be redundant if they are related by a generalization relation and if they have same support: they cover same

sequences in data. Over a set of redundant rules, we only select the most specific ones, that we will actually use for NER. Those are named "Annotation Rules".

4 NER using Annotation Rules

We now aim at determining when transduction rules should insert markers in texts. Using rules as features, we are able estimate probabilities for marker's presence as a trained model, as we will present in Sect. 4.1. The rules and the model are provided to a simple beam search algorithm described in Sect. 4.2 to actually annotate texts.

4.1 Estimating Likelihood of Annotated Sentences

As previously mentioned, instead of assigning a category to words (or tokens), our annotation rules insert markers at diverse positions in sentences. At a given position of a sentence, a decision should choose between adding any number of beginning or ending markers for NE categories (e.g. `<pers>`, `</pers>`, `<loc>`, etc.) or not to do so, what we denote by inserting a 'void marker" ($\emptyset$).

We train as many binary classifiers as distinct markers to estimate local probabilities for each individual markers. The probability of the presence of a single marker $m \in \mathcal{M}$ at a given position i is a random variable conditioned by the set of rules that have been triggered at current position: $P(m \in M_i | R_1, R_2 \dots R_k)$ that we note $P(m \in M_i)$. Combining those separate probabilities allows a direct computation for the probability of having multiple markers, as a multilabel problem:

$$P(M_i = \{m_1, m_2\}) = \prod_{m \in \{m_1, m_2\}} P(m \in M_i) \prod_{m \in \mathcal{M} - \{m_1, m_2\}} (1 - P(m \in M_i)) \qquad (2)$$

Finally, we use those local probabilities of the sequences[1] of markers (including the void marker $\emptyset$) to compute the likelihood of any annotation as n independent decisions over a sentence:

$$P(M_1, M_2 \dots M_n) = \prod_{i=1 \dots n} P(M_i) \qquad (3)$$

Indeed, what is considered as the most likely annotation within possible ones has to maximize that measure.

4.2 Decoding Step

The probability model may lead to an invalid sequence of markers according to the considered annotation scheme. The decoding algorithm must therefore consider only the valid proposals of the model. Depending on the dataset, we'll expect to be able to generate flat (no embeddings) or strutured xml-like (Etape) annotations. Those constraints

[1] Each set of markers is mapped to a predetermined number of corresponding markers sequences, e.g. $P(\{m_1, m_2\}) = P(< m_1, m_2 >) = P(< m_2, m_1 >) = P(< m_1, m_2, m_1 >)$

are implemented into the sequential algorithm, as a simple grammar to be checked while decoding. The latter uses a beam search approach[2], where adding a marker is considered as making a transition: for instance, inserting a <loc> marker at the beginning of a sentence moves from a "not inside any NE" state to a "in loc" state.

The algorithm starts from the beginning of a sequence and, for any position, generates new annotation hypothesis at this point by taking into account probabilities, possible marker sequences and annotation scheme constraints. Annotation hypothesis are then ordered and selected depending on their likelihood. Note that only the best hypothesis is kept for any given state, and that the number of states may be finite (flat annotation schemes) or infinite (when embeddings are allowed without depth restrictions). At the end of the sentence, the resulting annotation is the hypothesis that ends up with a "not inside any NE" state.

5 Ester2 and Etape French Evaluation Campaigns

Our system, mXs, has initially been developed for the Ester2 campaign. Sub-sections 5.2 and 5.3 present detailed experimental results conducted on this corpus. They give a better insight of the system behaviour and assess the influence of the support and confidence thresholds. Then, Sect. 5.4 presents the official performances during the Etape evaluation campaign, which was a follow up of Ester2.

5.1 Data: French Radio Transcripts

Our system participated to the Ester2 and Etape evaluation campaigns, which involved the French-speaking research community on the problematic of NER on radio transcripts. This task is much more challenging on this kind of noisy data, due to speech disfluences, speech recognition repairs and absence of sentence boundaries. This accordingly lowers performance of POS tagging and, at a higher level, requires a much more robust approach to find entities.

Corpus	**Tokens**	**Sentences**	**NEs**
Ester2-dev	73 386	2 491	5 326
Ester2-held	48 143	1 683	3 074
Ester2-corr	40 167	1 300	2 798
Total	128 477	4 283	8 670
Etape-train	355 975	14 989	46 259
Etape-dev	115 530	5 724	14 112
Etape-test	123 221	6 770	13 055
Total	594 726	27 483	73 426

Table 2: characteristics of Ester2 and Etape corpora

The French Ester2 and Etape evaluation campaigns included NER on transcribed texts [7]. The competing systems had to recognize persons, locations, organizations,

[2]It limits the search space by considering at any position N most probable solutions

products, amounts, time and positions. Entitites were manually annotated for evaluation purposes. As reported by [16], the Ester2 reference corpus contains many annotation inconsistencies. This is why we have decided to re-annotate consistantly one half of the corpus. This gold corpus will be named Ester2-corr while the second part was held out (Ester2-held). Such inconsistencies were avoided in the Etape corpus. Its annotation scheme is an extended version of Ester2's: evaluation is fine-grained and substructures of NEs are annotated as "components" [6]. Quantitative characteristics of those corpora are presented in Table 2.

5.2 Annotation Rules Extraction for Ester2

Corpora Ester2-dev and Ester2-held are merged to extract patterns. We used TreeTagger [20] for robustly tokenizing, POS-tagging and lemmatizing words (on French written texts, this tool provides high accuracy, more than 90% but, as far as we know, no evaluation has been made over oral transcriptions). The mining task requires many optimizations[15] and we used a level-wise algorithm [9] which leverages the generalization over patterns to mine frequent ones. Table 3 reports the number of rules, the number of non-redundant rules and the gain (i.e., the ratio between the number of rules and that of non-redundant ones). This elimination of redundant rules leads to a very significant reduction without loss of information from train corpus, what is very important for using this collection as a knowledge-base.

Sup.	**Conf.**	**Rules**	**Rules**	**Gain**
10	.5	207 544	7 172	**29**
5	.5	3 279 248	17 739	**185**
3	.3	85 187 894	46 019	**1851**

Table 3: extraction over Ester2 corpus at support and confidence thresholds

5.3 mXS Performance for Ester2

To assess efficiency of patterns for NER, we use Ester2-dev and Ester2-held merged to extract patterns and learn model, Ester2-corr to evaluate accuracy of the predicted markers. We train as many binary classifiers as necessary, using extracted rules as features to feed the logistic regression algorithm of SciKit toolkit [19][3]. In order to retrieve a set of rules that covers as much as possible actual markers in texts, we hereby extract rules at low support (3) and confidence (0.3) thresholds. With this exhaustive set of rules, only 52 markers out of 5196 (1%) are undetectable by the model because no rules are triggered at the considered position.

Table 4 presents global score Slot Error Rate (SER) [8] for diverse support and confidence thresholds. Those are computed by counting typed errors: insertions, deletions, types, extents. Results show that, for any support threshold, the model obtains better results at low confidence: even very generic (and thus less confident) rules are to be

[3]With regularization parameter C=4.

Support	Confidence	Insert	Delete	Type	Extent	SER
3	.3	18	**632**	102	287	**38,34**
3	.5	12	751	106	255	40.86
3	.7	13	944	**53**	257	48.60
5	.3	20	641	112	285	39.10
5	.5	10	752	108	271	41.57
5	.7	10	967	60	256	49.06
10	.3	18	693	114	292	40.77
10	.5	12	822	100	263	44.78
10	.7	**9**	1050	55	**238**	52.15

Table 4: detailed results

included as potentially relevant features. Globaly, using those rules as a standalone system remains unsufficient compared to state-of-the-art, but opens up great possibilities for coupling.

5.4 mXS performance for Etape

The Etape evaluation campaign extends Ester2 by considering TV broadcasts (including debates) and adding both fine grained and recursivity: NER is more difficult and requires robust approaches. Besides, the systems are now expected to recognize "componants" inside NEs [6], e.g. first names and last names for persons, day, month and year for dates, etc. As reported in Table 5, mXS is ranked 4th after one knowledge-based and two data-driven systems. Results show that the former have better performance for entities, while the latter are more accurate for componants. Despite being a data-driven system, mXS exhibits a behaviour similar to knowledge-based ones.

Participant	SER				Prec.	Rec.	F-score
	All	Entities	Comp.	Prim.			
Rules	85.5	80.4	88.2	74.6	36.8	16.5	22.8
Rules	156.0	178.1	143.9	172.3	17.3	28.0	21.4
CRF-bin	36.4	40.4	32.3	39.5	**85.3**	63.2	72.6
Rules	49.9	58.0	43.1	55.0	63.0	64.6	63.8
CRF+PCFG	44.6	39.0	49.3	36.2	66.1	53.6	59.2
CRF+PCFG	37.2	42.3	**32.0**	40.6	78.9	65.5	71.6
CRF	62.4	38.4	78.6	36.4	54.1	34.7	42.3
Rules	39.0	41.8	36.0	37.7	72.0	67.9	69.9
CasEN	**35.2**	**37.6**	34.1	**35.1**	74.2	**73.7**	**73.9**
mXS	37.9	39.2	34.1	36.8	77.6	65.1	70.8
Hybrid	51.3	54.7	45.9	52.7	76.8	49.6	60.3

Table 5: global performances of participants for Etape

The detailed results by NE types of Table 6 indicates that our system obtains good results for product category, which is the most difficult one. Conversely, it seems less

efficient for recognizing organizations, which are known to be very ambiguous [17]. This points out that pattern mining is accurate for detecting new NE types, but lacks fine-tuning for more traditional ones.

Participant	loc	org	pers	amo.	time	prod	func
Rules	59.6	29.2	64.4	18.2	17.0	24.3	10.0
Rules	21.4	38.9	51.2	8.8	52.0	39.2	36.9
CRF-bin	73.3	60.1	85.9	**68.7**	63.0	51.5	59.4
Rules	66.7	47.1	69.3	46.7	65.4	50.6	48.3
CRF+PCFG	74.1	61.0	83.3	66.8	64.2	54.3	61.5
CRF+PCFG	78.6	58.5	81.1	49.9	62.7	58.0	62.5
CRF	79.7	54.8	83.4	67.0	70.1	46.3	55.5
Rules	75.7	58.1	82.1	61.5	68.7	**62.5**	63.5
CasEN	**82.0**	**65.6**	**86.5**	44.0	**79.7**	57.0	**70.4**
mXS	81.4	58.4	79.9	60.3	65.1	**62.5**	67.2
Hybrid	71.2	44.1	77.3	51.7	11.6	44.3	52.8

Table 6: f-score of participants per primary NE type

6 Coupling mXS with the CasEN Symbolic System

We aim at improving performances of the existing system, CasEN, with the extracted patterns. Our symbolic system is precise, but lacks coverage because it would have to describe all regular expressions that may constitute a NE. Our idea is that automatically extracted patterns may supplement the symbolic system. We test this coupling by making CasEN's output a feature provided to mXS's input.

Table 7 reports the initial symbolic system's results, the differences of errors by NE categories and the resulting coupled system's performance. The symbolic system alone outperforms our standalone system using rules (28 vs 38 SER). By coupling systems, we observe a significant improvement of the symbolic system's output. The insertion of a small amount (2) of false-positive (Ins. total) is the counterpart for the correction of 26 type errors made by the symbolic system. This mainly concern amounts and the balancing between location and organizations (which are known to be very amibiguous).

We also isolated and manually examined rules that were responsible for the decrease of errors (coverage). Most of these rules are short and generalized rules, and quite frequently inserting only one marker (for instance 'from `<pers> NPP NPP`' or 'to `<loc> NPP`'). Interestingly, two time expressions have been found thanks to the separate detection of the beginning and the ending markers using local clues: 'for `<time>`' and 'years `</time>`' (recognizing "for a few years" for instance). How those shallow rules may be taken into account by the knowledge base of the symbolic system remains to be investigated.

Due to lack of space, we do not report other configurations and coupling strategies that has been successfully experimented, those are reported in [18]. They achieve

	Ins.	Del.	Typ.	Ext.	SER
Symbolic	45	343	165	257	28.7
amount	-1	+15	-25	-19	37.6
fonc	+2	+19	-1	-2	41.4
loc	-9	+8	+73	+22	26.7
org	+5	-27	-78	+49	41.5
pers	0	-4	+8	+26	19.4
prod	0	+2	-2	-2	85.2
time	+5	-11	-1	-74	18.3
total	+2	+2	-26	0	-1.3
Coupled	47	345	139	257	27.8

Table 7: error differences on CasEN with extracted rules

performances close to state-of-the-art systems when correctly set up. Our latest experiences that hybridates mXS and CasEN for Etape obtains 32.9 SER (compared with the performance of the best system, CasEN: 35.2). Generally, our experiments suggest that our system is efficient for combining other systems outputs, we plan to conduct more experimentation on this topic in future.

7 Conclusion

In this paper, we reported experimentations on the use of pattern mining techniques to automatically enrich a knowledge-based NER system. We implemented a prototype which extracts patterns correlated to NE markers. The system exhaustively looks for annotation rules from a training corpus and filters out those of interest. During the mining process, the text is represented as a sequence of items, which may be generalized using a hierarchy through POS categories, and where the beginning or ending markers of NEs may be separately mined.

The quality of patterns and their potential to recognize entites has been assessed and allowed us to state which are the most efficient and what markers categories remain to be improved. These experiments also investigated the idea of separately evaluating the probability to begin or end an entity, a beam search being afterwards responsible for finding the most likely and valid annotation. The resulting system was used coupled with a symbolic system, showing significant improvement of the performance. This work provides us with some interesting directions to improve a symbolic NER system, including in its foundations.

8 Acknowledgement

We'd like to thank Springer for allowing authors to share this authored version of the article published with reference LTC 2011, LNAI 8387 (Chapter 19, 978-3-319-08957-7, 328129_1_En).

References

[1] Andrew Borthwick, John Sterling, Eugene Agichtein, and Ralph Grishman. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *6th Workshop on Very Large Corpora (WVLC'1998)*, 1998.

[2] Béatrice Bouchou and Denis Maurel. Prolexbase et lmf : vers un standard pour les ressources lexicales sur les noms propres. *Traitement Automatique des Langues (TAL)*, 49:61–88, 2008.

[3] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165:91–134, 2005.

[4] Dianne Freitag and Nicholas Kushmerick. Boosted wrapper induction. In *European Conference on Artificial Intelligence (ECAI'00) - Workshop on Machine Learning for Information Extraction*, Berlin, Germany, 2000.

[5] Nathalie Friburger and Denis Maurel. Finite-state transducer cascades to extract named entities in texts. *Theoretical Computer Sciences (TCS)*, 313:93–104, 2004.

[6] Olivier Galibert, Sophie Rosset, Cyril Grouin, Pierre Zweigenbaum, and Ludovic Quintard. Structured and extended named entity evaluation in automatic speech transcriptions. In *International Joint Conference on Natural Language Processing (IJCNLP'11)*, 2011.

[7] Sylvain Galliano, Guillaume Gravier, and Laura Chaubard. The ester 2 evaluation campaign for the rich transcription of french radio broadcasts. In *10th Conference of the International Speech Communication Association (INTERSPEECH'2009)*, 2009.

[8] John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel. Performance measures for information extraction. *DARPA Broadcast News Workshop*, pages 249–252, 1994.

[9] Heikki Mannila, Hannu Toivonen, and A. Inkeri Verkamo. Discovery of frequent episodes in event sequences. In *Data Mining and Knowledge Discovery (DMKD)*, volume 1, pages 259–289, 1997.

[10] Elaine Marsh and Dennis Perzanowski. Muc-7 evaluation of ie technology: Overview of results. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998.

[11] Andrew McCallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *13th Conference on Computational Natural Language Learning (CONLL'2003)*, 2003.

[12] David D. McDonald. Internal and external evidence in the identification and semantic categorization of proper names. *Corpus Processing for Lexical Acquisition*, pages 21–39, 1996.

[13] Andrei Mikheev, Marc Moens, and Claire Grover. Named entity recognition without gazetteers. In *9th Conference of the European Chapter of the Association for Computational Linguistics (EACL'1999)*, 1999.

[14] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30:3–26, 2007.

[15] Damien Nouvel. *Reconnaissance des entités nommées par exploration de règles d'annotation*. PhD thesis, 2012.

[16] Damien Nouvel, Jean-Yves Antoine, Nathalie Friburger, and Denis Maurel. An analysis of the performances of the casen named entities recognition system in the ester2 evaluation campaign. In *7th International Language Resources and Evaluation (LREC'2010)*, 2010.

[17] Damien Nouvel, Jean-Yves Antoine, Nathalie Friburger, and Arnaud Soulet. Recognizing named entities using automatically extracted transduction rules. In *Language & Technology Conference (LTC'11)*, 2011.

[18] Damien Nouvel, Jean-Yves Antoine, Nathalie Friburger, and Arnaud Soulet. Coupling knowledge-based and data-driven systems for named entity recognition. In *Innovative hybrid approaches to the processing of textual data (HYBRID'12, EACL Workshop)*, 2012.

[19] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[20] Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *2nd International Conference on New Methods in Language Processing (NEMLP'1994)*, 1994.

[21] Ellen M. Voorhees and Donna Harman. In *International Speech Communication Association (INTERSPEECH'09)*.