Coupling Knowledge-Based and Data-Driven Systems for Named Entity Recognition

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

Abstract

Within Information Extraction tasks, Named Entity Recognition has received much attention over latest decades. From symbolic / knowledge-based to data-driven / machine-learning systems, many approaches have been experimented. Our work may be viewed as an attempt to bridge the gap from the data-driven perspective back to the knowledge-based one. We use a knowledge-based system, based on manually implemented transducers, that reaches satisfactory performances. It has the undisputable advantage of being modular. However, such a hand-crafted system requires substantial efforts to cope with dedicated tasks. In this context, we recently implemented a system that extracts symbolic knowledge, using hierarchical sequential pattern mining over annotated corpora. To assess the accuracy of mined patterns, we implemented an algorithm that recognizes Named Entities in texts by determining their most probable boundaries. Instead of considering Named Entity Recognition as a labeling task, it relies on complex context-aware features provided by lower-level systems and considers the tagging task as a markovian process. Using this system, coupling knowledge-based system with extracted patterns is straightforward and leads to a competitive hybrid NE-tagger. We report experiments using this system and compare it to other hybridization strategies along with a baseline CRF model.

1 Introduction

Named Entity Recognition (NER) is an information extraction (IE) 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 (Marsh & Perzanowski, 1998) or more recently speech transcripts provided by a human expert or an automatic speech recognition (ASR) system (Galliano et al., 2009). The recognized entities may later be used by higher-level tasks for different purposes such as Information Retrieval or Open-Domain Question-Answering (Voorhees & Harman, 2000).

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 word sequences containing speech recognition errors in ASR. 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 (e.g. *"the Computer Science department of the New York University"*). These complementary needs for robust and detailed processing explains that knowledge-based and data-driven approaches remain equally competitive on NER tasks (depending on annotation guidelines, considered language, availability of train corpora) as shown by numerous evaluation campaigns. For instance, the French-speaking Ester2 evaluation campaign on radio broadcasts (Galliano et al., 2009) has shown that knowledge-based approaches (participants Xerox, Synapse) outperformed data-driven ones on manual transcriptions while a system based on Conditional Random Fields (CRFs, participant LIA) is ranked first on noisy ASR transcripts.

In this paper, we present an original strategy of hybridization benefiting from features produced by a knowledge-based system (CasEN) and a data-driven pattern extractor (mXtrK). The former has been manually implemented based on *finite-state transducers*. Such a hand-crafted system requires substantial efforts to be adapted to dedicated tasks. We developed mXtrK, a text-mining system that automatically extracts *informative rules*, based on hierarchical sequential pattern mining. Both implement processings that are context-aware and use lexicons. Finally, to recognize NEs, we propose mStrucT, a light multi-purpose automatic annotator, parametrized using logistic regression over available features. It takes into account features provided by lower-level systems and annotation scheme constraints to output a valid annotation maximizing likelihood. Our experiments show that using those modules as a hybrid system outperforms standalone systems and reaches performances comparable to a baseline hybrid CRF system. We consider this as a step forward towards a tighter integration of knowledge-based and data-driven approaches for NER.

The paper is organized as follows. Section 2 describes the context of this work and reviews related work. Section 3 describes CasEN, the knowledge-based NE-tagger. Section 4 details the process of extracting patterns from annotated data as informative rules. We then introduce the automatic annotator mStrucT in Section 5. Section 6 describes how to gather features from systems and present diverse hybridization strategies. Corpora, metrics used and evaluation results are reported in Section 7. We conclude in Section 8.

2 Context and Related Work

2.1 Ester2 Evaluation Campaign

This paper focuses on NER in the context of the Ester2 evaluation campaign (Galliano et al., 2009). This campaign assesses system's performance for IE tasks over ASR outputs and manual transcriptions of radio broadcast news (see details in Section 7). The annotation guidelines specified 7 kinds of entities to be detected and categorized: persons ('pers'), organizations ('org'), locations ('loc'), amounts ('amount'), time expressions ('time'), occupations ('occ'), products ('prod'). Technically, the annotation scheme is quite simple: only one annotation per entity, almost no nesting (to the exception of a persons collocated with its occupation: both should be embedded in an encompassing 'pers' NE).

$\mathcal{D}$	
Sent.	Tokens and NEs
s_1	<pers> Isaac Newton </pers> was admitted in <time> June 1661 </time> to <org> Cambridge </org>.
s_2	<time> In 1696 </time>, he moved to <loc> London </loc> as <occ> warden of the Royal Mint </occ>.
s_3	He was buried in <loc> Westminster Abbey </loc>.

Table 1: Sentences from an annotated corpus

We illustrate the annotation scheme using a running example. Table 1 presents the expected annotation in the context of Ester2 from *"Isaac Newton was admitted in June 1661 to Cambridge. In 1696, he moved to London as warden of the Royal Mint. He was buried in Westminster Abbey."*. This example illustrates frequent problems for NER task. Determining the extent of a NE may be difficult, as is the case with: "Westminster" (city) and "Westminster Abbey" (church, building). Categorizign NEs is confronted to words ambiguities, for instance "Cambridge" may be considered as a city ('loc') or a university ('org'). In addition, oral transcripts may contains disfluencies, repetitions, hesitations, speech recognition errors: overall difficulty is significantly increased. For these reasons, NER over this data is a challenging task.

2.2 State of the Art

Knowledge-based approaches Most of the symbolic systems rely on shallow parsing techniques, applying regular expressions or linguistic patterns over Part-Of-Speech (POS), in addition to proper name lists checking. Some of them handle a deep syntactic analysis which has proven its ability to reach outstanding levels of performances (Brun & Hagège, 2004; Brun & Hagège, 2009; van Shooten et al., 2009).

Data-driven approaches A large diversity of data-driven approaches have been proposed during the last decade for NER. Generative models such as Hidden Markov Models or stochastic finite state transducers (Miller et al., 1998; Favre et al., 2005) benefit from their ability to take in account the sequential nature of language. On the other hand, discriminative classifiers such as

Support Vector Machines (SVMs) are very effective when a large variety of features (Isozaki & Kazawa, 2002) is used, but lack the ability to take a global decision over an entire sentence. Context Random Fields (CRFs) (Lafferty et al., 2001) have enabled NER to benefit from the advantages of both generative and discriminative approaches (McCallum & Li, 2003; Zidouni et al., 2010; Béchet & Charton, 2010). Besides, the robustness of data-driven / machine-learning approaches explains that the latter are more appropriate on noisy data such as ASR transcripts.

Hybrid systems Considering the complementary behaviors of knowledge-based and data-driven systems for NER, projects have been conducted to investigate how to conciliate both approaches. Work has been done to automatically induce symbolic knowledge (Hingston, 2002; Kushmerick et al., 1997) that may be used as NE taggers. But in most cases, hybridization for NER relies a much simpler principle: outputs of knowledge-based systems are considered as features by a machine learning algorithm. For instance, maximum entropy may be used when a high diversity of knowledge sources are to be taken into account (Borthwick et al., 1998). CRFs also have demonstrated their ability to merge symbolic and statistic processes in a machine learning framework (Zidouni et al., 2010).

We propose an original approach to combine knowledge-based and data-driven approaches in a modular way. Our first concern is to implement a module that automatically extracts knowledge, that should be interoperable with the existing system's transducers. This is done by focusing, in annotated corpora, more on '*markers*' (tags) that are to be inserted between tokens (e.g. `</pers>`, `<pers>`, `</org>`, `<org>`, etc.), than on 'labels' assigned to each token. By doing so, we also establish a better grounding for hybriding manually implemented and automatically extracted patterns. Afterwards, another module is responsible of annotating NEs by using those context-aware patterns and standard machine-learning techniques.

3 CasEN: a knowledge-based system

The knowledge-based system is based on CasSys (Friburger & Maurel, 2004), a finite-state cascade system that implements processings on texts at diverse levels (morphology, lexicon, chunking). It may be used for various IE tasks, or simply to transform or prepare a text for further processings. The principle of this finite-state processor is to first consider islands of certainty (Abney, 2011), so as to give priority to most confident rules. Each transducer describes local patterns corresponding to NEs or interesting linguistic units available to subsequent transducers within the cascade.

Casen is the set of NE recognition transducers. It was initially designed to process written texts, taking into account diverse linguistic clues, proper noun lists (covering a broad range of first names, countries, cities, etc.) and lexical evidences (expressions that may trigger recognition of a named entity).

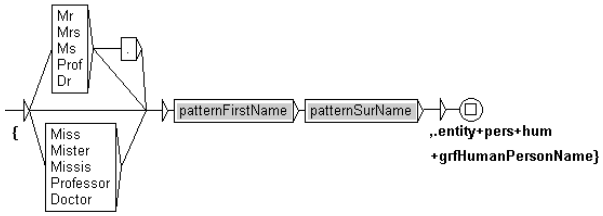

Figure 1: A transducer recognizing person names

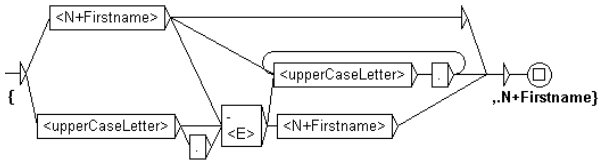

Figure 2: Transducer 'patternFirstName'

As an illustration, Figure 1 presents a very simple transducer tagging person names made of an optional title, a firstname and a surname. The boxes contain the transitions of the transducer as items to be matched for recognizing a person's name. Greyed boxes contain inclusions of other transducers (e.g. box 'patternFirstName' in Figure 1 is to be replaced by the transducer depicted in Figure 2). Other boxes can contain lists of words or diverse tags (e.g. `<N+firstname>` for a word tagged as first name by lexicon). The outputs of transducers are displayed below boxes (e.g. *'{'* and *',.entity+pers+hum}'* in Figure 1).

For instance, that transducer matches the word sequence *'Isaac Newton'* and outputs: '{{*Isaac ,.firstname*} {*Newton ,.surname*} *,.entity+pers+hum*}'. By applying multiple transduc-

ers on a text sequence, CasEN can provide several (possibly nested) annotations on a NE and its components. This has the advantage of providing detailed information about CasEN internal processings for NER.

Finally, the processing of examples in Table 1 leads to annotations such as:

- { { *June ,.month*} { *1661 ,.year*} *,entity+time+date+rel*}
- { *Westminster ,.entity+loc+city*} { *Abbey ,buildingName*} *,.entity+loc+buildingCityName* }

When run in standalone mode, post-processing steps convert these outputs into the Ester2 annotation scheme (e.g. `<pers>` *Isaac Newton* `</pers>`).

Experiments conducted on newspaper documents for recognizing persons, organisations and locations on an extract of the Le Monde corpus have shown that CasEN reaches 93.2% of recall and 91.1% of f-score (Friburger, 2002). During the Ester2 evaluation campaign, CasEN ("LI Tours" participant in (Galliano et al., 2009)) obtained 33.7% SER (Slot Error Rate, see section about metrics description) and a f-score of 75%. This may be considered as satisfying when one knows the lack of adaptation of Casen to specificities of oral transcribed texts.

4 mXtrK: Pattern Mining Method

4.1 Enriching an Annotated Corpus

We investigated the use of data mining techniques in order to supplement our knowledge-based system. For this purpose, we use an annotated corpus to mine patterns related to NEs. Sentences are considered as sequences of *items* (this precludes extraction of patterns accross sentences). An item is either a word from natural language (e.g. "admitted", "Newton") or a tag delimiting NE categories (e.g., `<pers>`, `</pers>` or `<loc>`). The annotated corpus $\mathcal{D}$ is a multiset of sequences based on those items.

Preprocessing steps enrich the corpus by (1) using lexical resources (lists of toponyms, anthroponyms and so on) and (2) lemmatizing and applying a POS tagger. This results in a *multi-dimensional corpus* where a token may be gradually generalized to its lemma, POS or lexical category. For instance, Figure 3 illustrates this process on the extract 'moved to `<loc>` London `</loc>`'.

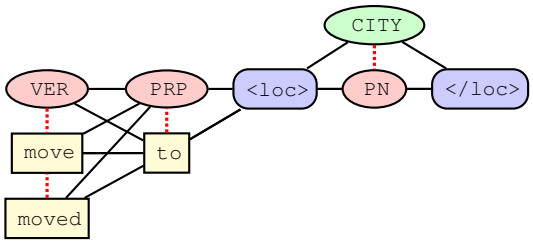

Figure 3: Multi-dimensional representation of the phrase 'moved to `<loc>` London `</loc>`'

At first, applying lexical ressources ambiguously assign, whenever applicable, tokens to lexical categories (e.g., `CITY` for "London"). Note that those resources contain multi-word expressions. Figure 4 provides a short extract limited to tokens of Table 1) of lexical ressources (totalizing 201,057 entries). For instance, processing "Westminster Abbey" would lead to categorizing 'Westminster' as `CITY` and the whole as `INST`.

Afterwards, a POS tagger based on TreeTagger (Schmid, 1994) distinguishes common nouns (`NN`) from proper names (`PN`). For the latter, token is deleted (only `PN` category is kept) to avoid extraction of patterns that would be specific to a given proper name (in Figure 3, "London" is removed). Figure 5 shows how POS categories are organized as a hierarchy.

Category	Tokens
ANTHRO	Newton, Royal ...
CITY	Cambridge, London, Westminster ...
INST	Cambridge, Royal Mint, Westminster Abbey ...
METRIC	Newton ...
...	...

Figure 4: Lexical Ressources

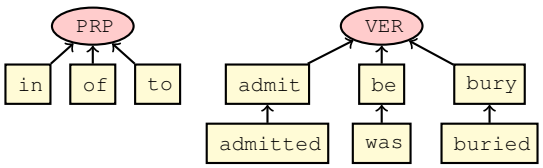

Figure 5: POS Hierarchy

4.2 Discovering Informative Rules

We mine this large enriched annotated corpus to find generalized patterns that are correlated to NE markers. It consists in exhaustively enumerating

over this corpus *all* the contiguous patterns mixing words, POS and categories. This provides a very broad spectrum of patterns, diversely accurate to recognize NEs. For instance, considering the sentence "moved to <loc> London </loc>" in Figure 3 leads to examining patterns as:

- ' VER PRP <loc> PN </loc>'
- ' VER to <loc> PN </loc>'
- ' moved PRP <loc> CITY </loc>'

As usual in data mining, we focus on the most relevant ones by setting thresholds on two interestingness measures: support and confidence (Agrawal & Srikant, 1994). 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. As we are only interested in patterns sufficiently correlated to markers, a *rule* R is defined as a pattern containing at least one marker. To estimate empirically how much R is accurate to detect markers, we calculate its *confidence*. A dedicated function $suppNoMark(R, \mathcal{D})$ returns the support of R when markers are omitted both in the rule and in the data. The confidence of R is:

$$conf(R, \mathcal{D}) = \frac{supp(R, \mathcal{D})}{suppNoMark(R, \mathcal{D})}$$

For instance, consider the rule $R =$ ' VER PRP <loc>' in Table 1. Its support is 2 (sentences s_2 and s_3). But its support without considering markers is 3, since sentence s_1 matches the rule when markers are not taken in consideration. The confidence of R is 2/3.

In practice, the whole collection of transduction rules exceeding minimal support and confidence thresholds remains too large, especially when searching for less frequent patterns. Consequently, we filter-out "redundant rules": those for which a more specific rule exists with same support (both cover same examples in corpus). For instance, the rules $R_1 =$ ' VER VER in <loc>' and $R_2 =$ ' VER in <loc>' are more general and have same support than $R_3 =$ ' was VER in <loc>': we only retain the latter.

The system mXtrK (micro-extract) implements those processing using a level-wise algorithm (Mannila & Toivonen, 1997).

5 mStrucT: Stochastic Model for NER

Our aim is to be able to get information from systems so as to recognize NEs. To this end, we established a common ground for the systems to interact with a higher level model. We assume that lower level systems examine the input (sentences) and provide valuable clues playing a key role in the recognition of NEs.

In that context, the annotator does not need to rely on the words / tokens representation. It is implemented as an abstracted view of sentences. Decisions have only to be taken whenever one of the lower-level systems provides information. Formally, beginning or ending a NE at a given position i may be viewed as the affectation of a random variable $P(Mi = m_{j_i})$ where the value of m_{j_i} is one of the markers ($\{\emptyset, <pers>, </pers>, <loc>, <org>, \dots\}$).

For a given sentence, the knowledge-based system and extracted rules will both propose indicators at a given position (see section 6.1) that are converted into as many binary features. We use these for predicting what marker is the most probable at that very position. This may be viewed as an instance of a classification problem (more precisely multilabel classification since several markers may appear at a single position, but we won't enter into that level of detail due to lack of space). Empirical experiments with diverse machine learning algorithms (using Scikit-learn (Pedregosa et al., 2011)) conducted to consider logistic regression as the most effective.

Considering those probabilities, it is now possible to estimate the likelihood of a given annotation over a sentence. To do so, we assume markers are independent (what remains an approximation). Computing the likelihood of an annotation becomes a fairly simple product:

$$P(M_1 = m_{j_1}, M_2 = m_{j_2}, \dots, M_n = m_{j_n})$$
$$\approx \prod_{i=1\dots n} P(M_i = m_{j_i})$$

Estimating markers probabilities independently allows the model to combine evidences from separate knwoledge sources to recognize starting or ending boundaries. For instance, CasEN may recongize intermediary structures but not the whole entity (e.g. when unexpected words appear inside it) while extracted rules may propose markers that

are not necessarily paired. Based on the lower-level systems, the hybridization will still be able to recognize the entity, without having to match all NE's tokens.

Figure 6 gives an example for finding an annotation from probabilities of markers, using the Ester2 annotation scheme. For clarity purposes, only sufficiently probable markers (including $\emptyset$) are displayed at each position. A possible <occ> is discarded (crossed out), being less probable than a previous one. An annotation solution <org>...</org> is evaluated, but is less likely ($0.3 * 0.4 * 0.9 * 0.4 * 0.4 * 0.1 = 0,0017$) than *warden of the Royal Mint* as an occupation ($0.6 * 0.4 * 0.9 * 0.3 * 0.5 * 0.4 = 0,0129$) which will be retained (and is the expected annotation).

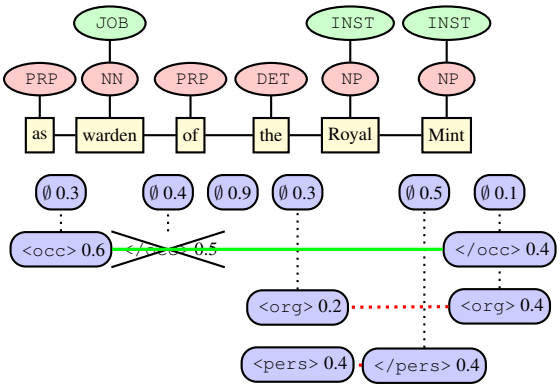

Figure 6: Stochastic Model running example

Indeed, it is not necessary to compute likelihoods over all possible combination of markers, since the annotation scheme is much constrained. As the sentence is processed, some annotation solution are to be discarded. It is straightforward to see that this problem may be resolved using dynamic programming, as did Borthwick et al. (1998). Depending on the annotation scheme, constraints are provided to the annotator which outputs an annotation for a given sentence that is valid and that maximizes likelihood. Our system mStrucT (micro-structurate) implements this (potentially multi-purpose) automatic annotation process as a separate module.

6 Hybriding systems

6.1 Gathering Clues from Systems

Figure 7 describes the diverse resources and algorithms that are plugged together. The knowledge-based system uses lists that recognize lexical patterns useful for NER (e.g. proper names, but also automata to detect time expressions, occupations, etc.). Those resources are exported and available to the data mining software as lexical resources (see section 4) and (as binary features) to the baseline CRF model.

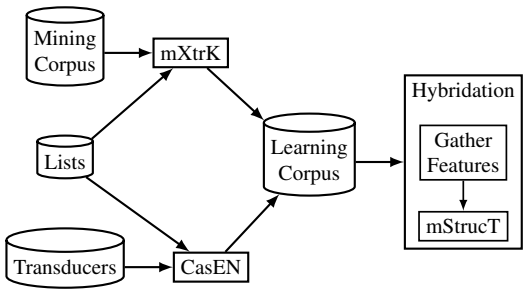

Figure 7: Systems Modules (Hybrid data flow)

Each system processes input text and provides features used by the Stochastic Model mStrucT. It is quite simple to take in consideration mined informative rules: each time a rule i proposes its j^{th} marker, a boolean feature M_{ij} is activated. What is provided by CasEN is more sophisticated, since each transducer is able to indicate more detailed information (see section 3), as multiple features separated by '+' (e.g. 'entity+pers+hum'). We want to benefit as much as possible from this richness: whenever a CasEN tag begins or ends, we activate a boolean feature for each mentioned feature plus one for each prefixes of features (e.g. 'entity', 'pers', 'hum' but also 'entity.pers' and 'entity.pers.hum').

6.2 Coupling Strategies

Up to now, we have described the diverse modules (the knowledge-based system CasEN, the informative rules extractor mXtrK and the stochastic model mStrucT) involved in our NER system. Several combination have been experimented to make those modules interact together. Furthermore, we also implemented a baseline CRF model using Wapiti (Lavergne et al., 2010) so as to have an idea of how the hybrid system compares to a baseline state of the art model. We report results for the following systems:

- CasEN: knowledge-based system standalone
- mXS: mXtrK extracts rules, mStrucT annotates
- Hybrid: gather features from CasEN and mXtrK, mStrucT annotates
- Hybrid-sel: as Hybrid, but features are selected

Corpus	Tokens	Sentences	NEs
Ester2-Train	1 269 138	44 211	80 227
Ester2-Dev	73 375	2 491	5 326
Ester2-Test-corr	39 704	1 300	2 798
Ester2-Test-held	47 446	1 683	3 067

Table 2: Characteristics of Corpora

- CasEN-mXS-mine: as mXS, but text is pre-processed by CasEN (adding a higher generalization level above lexical lists)
- mXS-CasEN-vote: as mXS, plus a post-processing step as a vote based on mXS and CasEN outputs
- CRF: baseline CRF, using BIO and common features (unigrams: lemma and lexical lists, bigrams: previous, current and next POS)
- CasEN-CRF: same as CRF, but the output of CasEN is added as a single feature (concatenation of CasEN features)

7 Experimentations

7.1 Corpora and Metrics

For experimentations, we use the corpus that has been made available after the Ester2 evaluation campaign. Table 2 gives statistics on diverse subparts of this corpus. Unfortunately, much inconsistencies where noted for manual annotation, especially for 'Ester2-Train' part that won't be used for training.

There were fewer irregularities in other parts of the corpus. Although, manual corrections were done on half of the Test corpus (Nouvel et al., 2010) (Ester2-Test-corr in Table 2), to obtain a gold standard that we will use to evaluate our approach. The remaining part of the Test corpus (Ester2-Test-held in Table 2) merged with the Dev part constitute our training set (Ester2-Dev in Table 2), used as well to extract rules with mXtrK, to estimate stochastic model probabilities of mStrucT and to learn CRF models.

We evaluate systems using following metrics:

- *detect*: rate of detection of the presence of any marker (binary decision) at any position
- *desamb*: f-score of markers when comparing the N actual markers to N most probable markers, computed over positions where k markers are expected (N=k) or at least one non-$\emptyset$ marker is probable (N=1)

System	support	confidence	detect	disamb	f-score	SER
CasEN	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	78	30.8
mXS	5	0.1	97	73	76	28.4
	5	0.5	96	71	74	31.2
	15	0.1	96	72	73	30.1
Hybrid	5	0.1	97	78	79	26.3
	5	0.5	97	77	77	28.3
	15	0.1	97	78	76	28.2
	inf	inf	96	71	70	42.0

Table 3: Performance of Systems

- *precision, recall, f-score*: evaluation of NER by categories by examining the label assigned to *each* token (similarly to Ester2 results)
- *SER* (Slot Error Rate): weighted error rate of NER (official Ester2 performance metric, to be lowered), where errors are discounted per entity, this metric is used as Galliano et al. (2009) does: deletion and insertion errors are weighted 1 whereas type and boundary errors, 0.5

7.2 Comparing Hybridation with Systems

First, we separately evaluate systems. While CasEN is not to be parameterized, mXtrK has to be given minimum frequency and support thresholds. Table 3 shows results for each system separately and for the combination of systems. Results obtained by mXS show that even less confident rules are improving performances. Generally speaking, the *detect* score is very high, but this mainly due to the fact that the $\emptyset$ case is very frequent. The *disamb* score is much correlated to the SER. This reflects the fact that the challenge is for mStrucT to determine the correct markers to insert.

Comparing systems shows that the hybridization strategy is effective. The knowledge-based system yields to satisfying results, mXS obtains slightly better SER and the hybrid system outperforms both in most configurations. Considering SER, the only exception to this is the 'inf' line (mStrucT uses only CasEN features) where performances are degraded. In general, we remark that mStrucT obtains better results as more rules are extracted.

7.3 Assessing Hybridation Strategies

In a second step, we look in detail what NE types are the most accurately recognized. Those results are reported in Figure 8, where is depicted the error rates (to be lowered) for main types ('prod',

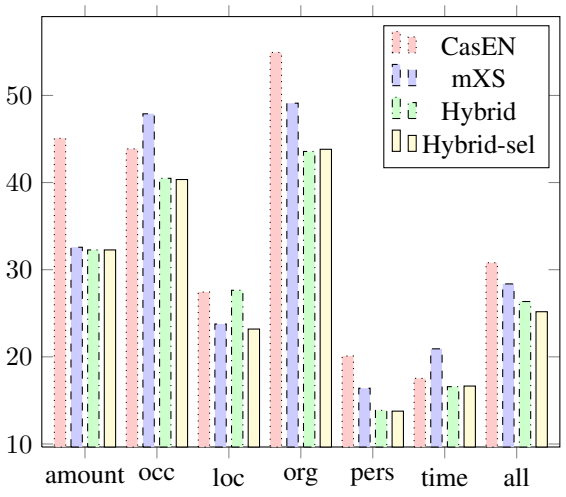

Figure 8: SER of Systems by NE types

System	precision	recall	f-score	SER
Hybrid-sel	83.1	74.8	79	25.2
CasEN-mXS-mine	76.8	75.5	76	29.4
mXS-CasEN-vote	78.7	79.0	79	26.9
CRF	83.8	77.3	80	26.1
CasEN-CRF	84.1	77.5	81	26.0

Table 4: Comparing performances of systems

being rare, is not reported). This revealed that features provided by CasEN for 'loc' type appeared to be unreliable for mStrucT. Therefore, we filtered-out related features, so as to couple systems in a more efficient fashion. This leads to a 1.1 SER gain (from 26.3 to 25.2) when running the so-called 'Hybrid-sel' system, and demonstrates that the hybridation is very sensitive to what is provided by CasEN.

With this constrained hybridization, we compare previous results to other hybridization strategies and a baseline CRF system as described in section 6. Those experiments are reported in Table 4. We see that, when considering SER, the hybridization strategy using CasEN features within mStrucT stochastic model slightly outperforms 'simpler' hybridizations schemes (pre-processing or post-processing with CasEN) and the CRF model (even when it uses CasEN preprocessing as a single unigram feature).

However the f-score metric gives advantage to CasEN-CRF, especially when considering recall. By looking indepth into errors and when reminded that SER is a weighted metric based on slots (entities) while f-score is based on tokens (see section 7.1), we noted that on longest

System	NE type	insert	delet	type	SER	f-score
Hybrid-sel	occ	8	21	7	40.3	65
	all	103	205	210	25.2	79
CasEN-CRF	occ	9	37	0	53.5	64
	all	77	251	196	26.0	81

Table 5: Impact of 'occ' over SER and f-score

NEs (mainly 'occ'), Hybrid-sel does type errors (discounted as 0.5 in SER) while CasEN-CRF does deletion errors (1 in SER). This is pointed out by Table 5. The influence of error's type is clear when considering the SER for 'occ' type for which Hybrid-sel is better while f-score doesn't measure such a difference.

7.4 Discussion and Perspectives

Assessment of performances using a baseline CRF pre-processed by CasEN and the hybrided strategy system shows that our approach is competitive, but do not allow to draw definitive conclusions. We keep in mind that the evaluated CRF could be further improved. Other methods have been successfully experimented to couple more efficiently that kind of data-driven approach with a knowledge-based one (for instance Zidouni et al. (2010) reports 20.3% SER on Ester2 test corpus, but they leverage training corpus).

Nevertheless, the CRFs models do not allow to directly extract symbolic knowledge from data. We aim at organizing our NER system in a modular way, so as to be able to adapt it to dedicated tasks, even if no training data is available. Results show that this proposed hybridization reaches a satisfactory level of performances.

This kind of hybridization, focusing on "markers", is especially relevant for annotation tasks. As a next step, experiments are to be conducted on other tasks, especially those involving nested annotations that our current system is able to process. We will also consider how to better organize and integrate automatically extracted informative rules into our existing knowledge-based system.

8 Conclusion

In this paper, we consider Named Entity Recognition task as the ability to detect boundaries of Named Entities. We use CasEN, a knowledge-based system based on transducers and mXtrK, a text-mining approach to extract informative rules from annotated texts. To test these rules, we

propose mStrucT, a light multi-purpose annotator that has the originality to focus on boundaries of Named Entities ("markers"), without considering the labels associated to tokens. The extraction module and the stochastic model are plugged together, resulting in mXS, a NE-tagger that gives satisfactory results. Those systems altogether may be hybridized in an efficient fashion. We assess performances of our approach by reporting results of our system compared to other baseline hybridization strategies and CRF systems.

References

Steven P. Abney. 1991. Parsing by Chunks. *Principle-Based Parsing*, 257–278.

Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast algorithms for mining association rules in large databases. *Very Large Data Bases*, 487–499.

Frédéric Bechet and Eric Charton. 2010. Unsupervised knowledge acquisition for Extracting Named Entities from speech. *Acoustics, Speech, and Signal Processing (ICASSP'10)*, Dallas, USA.

Andrew Borthwick, John Sterling, Eugene Agichtein and Ralph Grishman. 1998. Exploiting Diverse Knowledge Sources via Maximum Entropy in Named Entity Recognition. *Very Large Corpora (VLC'98)*, Montreal, Canada.

Caroline Brun and Caroline Hagège. 2004. Intertwining Deep Syntactic Processing and Named Entity Detection. *Advances in Natural Language Processing*, 3230:195-206.

Caroline Brun and Maud Ehrmann. 2009. Adaptation of a named entity recognition system for the ester 2 evaluation campaign. *Natural Language Processing and Knowledge Engineering (NLPK'09)*, Dalian, China.

Benoît Favre, Frédéric Béchet, and Pascal Nocera. 2005. Robust Named Entity Extraction from Large Spoken Archives. *Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP'05)*, Vancouver, Canada.

Nathalie Friburger. 2002. Reconnaissance automatique des noms propres: Application à la classification automatique de textes journalistiques. *PhD*.

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

Sylvain Galliano, Guillaume Gravier and Laura Chaubard. 2009. The ESTER 2 evaluation campaign for the rich transcription of French radio broadcasts. *International Speech Communication Association (INTERSPEECH'09)*, Brighton, UK.

Philip Hingston. 2002. Using Finite State Automata for Sequence Mining. *Australasian Computer Science Conference (ACSC'02)*, Melbourne, Australia.

Hideki Isozaki and Hideto Kazawa. 2002. Efficient support vector classifiers for named entity recognition. *Conference on Computational linguistics (COLING'02)*, Taipei, Taiwan.

Nicholas Kushmerick and Daniel S. Weld and Robert Doorenbos. 1997. Wrapper Induction for Information Extraction. *International Joint Conference on Artificial Intelligence (IJCAI'97)*, Nagoya, Japan.

John D. Lafferty, Andrew McCallum and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *International Conference on Machine Learning (ICML'01)*, Massachusetts, USA.

Thomas Lavergne and Olivier Cappé and François Yvon 2010. Practical Very Large Scale CRFs. *Association for Computational Linguistics (ACL'10)*, Uppsala, Sweden.

Heikki Mannila and Hannu Toivonen. 1997. Level-wise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery*, 1(3):241–258.

Elaine Marsh and Dennis Perzanowski. 1998. MUC-7 Evaluation of IE Technology: Overview of Results. *Message Understanding Conference (MUC-7)*.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. *Computational Natural Language Learning (CONLL'03)*, Edmonton, Canada.

Scott Miller, Michael Crystal, Heidi Fox, Lance Ramshaw, Richard Schwartz, Rebecca Stone and Ralph Weischedel. 1998. Algorithms That Learn To Extract Information BBN: Description Of The Sift System As Used For MUC-7. *Message Understanding Conference (MUC-7)*.

Damien Nouvel, Jean-Yves Antoine, Nathalie Friburger and Denis Maurel. 2010. An Analysis of the Performances of the CasEN Named Entities Recognition System in the Ester2 Evaluation Campaign. *Language Resources and Evaluation (LREC'10)*, Valetta, Malta.

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. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Helmut Schmid. 1994. Probabilistic POS Tagging Using Decision Trees. *New Methods in Language Processing (NEMLP'94*, Manchester, UK.

Boris W. van Schooten, Sophie Rosset, Olivier Galibert, Aurélien Max, Rieks op den Akker, and Gabriel Illouz. 2009. Handling speech in the ritel QA dialogue system. *International Speech Communication Association (INTERSPEECH'09)*, Brighton, UK.

Ellen M. Voorhees and Donna Harman. 2000. Overview of the Ninth Text REtrieval Conference (TREC-9). *International Speech Communication Association (INTERSPEECH'09)*, Brighton, UK.

Azeddine Zidouni and Sophie Rosset and Hervé Glotin 2010. Efficient combined approach for named entity recognition in spoken language. *International Speech Communication Association (INTERSPEECH'10)*, Makuhari, Japan.