KnowNER: Incremental Multilingual Knowledge in Named Entity Recognition

Dominic Seyler
University of Illinois at
Urbana-Champaign
dseyler2@illinois.edu

Tatiana Dembelova
Max Planck Institute for Informatics
tdembelo@mpi-inf.mpg.de

Luciano Del Corro
Max Planck Institute for Informatics
corrogg@mpi-inf.mpg.de

Johannes Hoffart
Max Planck Institute for Informatics
jhoffart@mpi-inf.mpg.de

Gerhard Weikum
Max Planck Institute for Informatics
weikum@mpi-inf.mpg.de

ABSTRACT

KnowNER is a multilingual Named Entity Recognition (NER) system that leverages different degrees of external knowledge. A novel modular framework divides the knowledge into four categories according to the depth of knowledge they convey. Each category consists of a set of features automatically generated from different information sources (such as a knowledge-base, a list of names or document-specific semantic annotations) and is used to train a conditional random field (CRF). Since those information sources are usually multilingual, KnowNER can be easily trained for a wide range of languages. In this paper, we show that the incorporation of deeper knowledge systematically boosts accuracy and compare KnowNER with state-of-the-art NER approaches across three languages (i.e., English, German and Spanish) performing amongst state-of-the art systems in all of them.

1 INTRODUCTION

Named Entity Recognition (NER) is the task of detecting named entity mentions in text and assigning them to their corresponding coarse-grained type (e.g., person, location, organization, miscellaneous). For instance, given the sentence "Jimmy Page played in New York", the goal is to recognize "Jimmy Page" and "New York" as named entities and classify them as person and location. NER is a key component in a wide range of natural language understanding tasks such as named entity disambiguation (NED), information extraction, question answering, machine translation, knowledge graph construction, etc.

Here we present KnowNER, a multilingual NER system which incorporates different degrees of external knowledge through language agnostic features, designed to exploit existing multilingual knowledge resources.

In contrast to previous approaches, KnowNER is implemented as a modular framework, drawing on different sources of external knowledge. We divide the information sources into four different categories according to the depth of knowledge they convey.

Each one carries more information than the previous. This additional knowledge boosts accuracy but also increases the processing overhead, establishing a clear accuracy-speed trade-off that can be exploited according to processing requirements and the availability of computational and knowledge resources.

This work has three main goals: (i) present a high performance knowledge intensive NER framework that can be used for a wide range of languages, (ii) understand to which extent external knowledge improves NER performance, and (iii) present a novel set of knowledge intensive features that can be used in a multilingual setting.

KnowNER implements a linear chain CRF, which was proven to work well for the NER task [5]. We divide the features according to the knowledge categories defined below:

Agnostic. These features correspond to the standard lexico-syntactic features extensively used in literature [5]. They are usually called local features since they are directly extracted from text and do not use any external knowledge. For instance, part-of-speech (POS) tags are good indicators of named entities (e.g., In "Jimmy Page plays guitar", "Jimmy" and "Page" are proper nouns).

Name-based. This set is extracted from a list with millions of named entity names. They unveil common patterns or attributes that indicate the presence of named entities. For example, the word "Jimmy" is usually associated with named entities.

KB-based. This group is generated from a knowledge base (KB) or an entity annotated corpus. The aim is to go beyond the surface forms exposing particular semantics of the named entities (e.g., their types). Following Ratinov and Roth. [19], we use gazetteers that associate named entity names with types. We generate them in an automatic way from YAGO [21], a multilingual KB. We also use it to extract richer information like the probability of a single token having a given type and appear in a specific position (e.g., the probability of "Jimmy" being a person and appearing at the beginning of the name). Additionally, we exploit an annotated corpus to estimate the likelihood of a token referring to a named entity (e.g., The number of times "Page" is linked in Wikipedia articles).

Entity-based. These features exploit information from a particular document. The idea is that if some entities in the document are identified in advance, it is easier to spot more difficult cases later. For instance, if we know that the *European Union* is mentioned in a text, we can assume that the token "EU" will most probably refer to it. Previous work [18] builds on this idea, using disambiguated entities

from ground truth data to extract document specific features. We follow this approach but, in addition, we evaluate our system in a real world scenario using AIDA [8], a state-of-the-art entity-linking system.

When KnowNER includes all knowledge categories, it performs among the best NER systems across all the evaluated languages (i.e., English, German and Spanish) on four standard datasets. In the experimental section, we also present an extensive study showing that the degree of knowledge correlates positively with task accuracy and negatively with processing time. We also show that external knowledge is particularly important for types like organizations, persons, and locations, reaching in the last two cases human-level accuracy (more than 95 F1 points) for the English language. Apart from the traditional NER metrics (class label plus text span) we additionally report the span recognition accuracy (named entity without the type tag), essential for certain tasks (e.g., NED).

Now we summarize our central contributions:

- A high performance multilingual NER system based on a modular framework for incorporating different types of external knowledge.
- A comprehensive study to verify the impact of external knowledge into NER, including ablation and timing experiments.
- A multilingual set of knowledge intensive automatically generated features derived from large list of names, or a multilingual KB.
- Real world scenario experiments to test the specific effects of NED into NER.

2 THE NAMED ENTITY RECOGNITION TASK

2.1 Task Definition

The goal of NER is to find named entity mentions in text and map them to pre-defined types (e.g., person, location, etc.). For instance, in the sentence "Jimmy Page plays guitar." the goal is to recognize that the text span "Jimmy Page" refers to a named entity that can be categorized as person.

The task implies two challenges: (i) Find the text span of a named entity name and (ii) Annotate each named entity with a type. The first challenge requires identifying tokens that refer to named entities. A named entity may be composed by more than one token ("United States"), and a named entity may be embedded in another named entity ("Supreme Court of the United States"). The second challenge requires deeper semantic understanding (e.g. understand that "Jimmy Page" is not only a named entity but specifically a person).

Although NER commonly refers to both tasks, some applications may rely only on the first one (e.g. NED). In Sec. 4 we present results for the named entity mention span detection separately.

2.2 A linear chain CRF model

Previous work [5, 9, 14, 16, 18, 19] proved the effectiveness of CRFs [11] for the NER task. We implemented KnowNER as a linear chain CRF similar to [5]. The underlying idea is to cast NER as a sequence model with a bidirectional flow. The CRF represents the probability of a hidden state sequence (i.e., token labels) given a set of observations. In a linear chain CRF, the probability of a token

being a named entity depends on a set of observations including the label of its adjacent neighbors. For a more in-depth description of the model refer to Finkel et al., 2005.

3 KNOWLEDGE AUGMENTED NER

Here, we describe the knowledge categories, which function as modules in our system. We define four: *agnostic (A)*, *name-based*, *KB-based* and *entity-based*, each containing an increasing amount of external knowledge. A category consists of the set of features, sumarized in Tab. 1.

3.1 Knowledge Agnostic

This category contains the so-called "local" features. Their distinctive characteristic is that they can be extracted directly from text without any external knowledge. These features are mostly of a lexical, syntactic or linguistic nature and have been well-studied in literature. We implement most of the features described in Finkel et al. [5] and Zhang and Johnson. [24], namely:

(1) The current word and words in a window of size 2 ; (2) Word shapes of the current word and words in a window of size 2; (3) POS tags in a window of size 2; (4) Prefixes (length three and four) and Suffixes (length one to four); (5) Presence of the current word in a window of size 4; (6) Beginning of sentence.

3.2 Name-Based Knowledge

In this category, the knowledge is extracted from a list of named entity names. This list does not carry any additional information apart from the names themselves. The intuition is that names tend to follow patterns and even the set of possible names is limited. To the best of our knowledge, these features have not been previously used. We extracted a list of all names from YAGO [21] (30.85M for the languages we trained on) and created the following features:

Frequent mention tokens. Reflects the frequency of a given token in a list of entity names. We tokenized the list to compute the frequencies. The feature assigns a weight to each token in the text corresponding to their normalized frequency. The intuition is that some words like "John" or "Organization" may be indicative of a named entity and thus carry a high weight. For instance, the top-5 tokens we found in English were "county", "john", "school", "station" and "district". All tokens without occurences are assigned 0 weight.

Frequent POS Tag Sequences. This feature intends to identify POS sequences common to named entities. For example, person names tend to be described as a series of proper nouns, while organizations may have richer patterns. For instance, both "Organization of American States" and "Union for Ethical Biotrade" share the pattern NNP-IN-NNP-NNP, where NNP is a proper noun and IN a preposition. To generate these patterns, we construct a simple artificial sentence for each name in our list and run a POS-tagger. We then compute and rank the entity POS tag sequences and keep the top 100. The feature is implemented by finding the longest matching POS sequences in the input text and marking whether the current token belongs to a frequent sequence or not. We search the sequences from left to right and, in case of overlap, annotate only the leftmost sequence. This might need to be done differently for languages that read right to left.

Cat.	Feature	Description	Example
A	Word	specific words tend to indicate the presence of a NE	John says that … → [SOMEONE] says that …
	Word shapes	NE have specific shapes	John, Paul → Xxxx
	POS tags	NE tend to have specific POS tags	John, Paul → NNP
	Prefixes/Suffixes	NE tend to share prefixes and suffixes	Freiburg; Marburg
	Presence Window	NEs usually don't appear twice in a small window	To be or not to be Obama was born in Hawaii
	Sentence Begin	NE at the beginning of sentences difficult to spot	John says …; Computers are …
Name	**Mention tokens**	Some tokens are strongly associated to NEs	county,john,school,station,…
	POS-tag sequence	Multi-word NEs tend to share POS patterns	Organization of American States Union for Ethical Biotrade → NNP IN NNP NNP
KB	Type gazetteers	Some names are strongly associated to types	Barack Obama → person Florida → location
	Wiki. link prob.	Certain tokens are usually associated to NEs	Obama is usually linked to *Barack Obama* in Wikipedia
	Type prob.	Certain tokens are associated to types with high probability	Barack → person;
Entity	Doc. gazetteers	Presence of specific NEs may indicate other NE names	European Union → EU

Table 1: Features by category (novel features are highlighted)

3.3 Knowledge-Base-Based Knowledge

This category groups features that are extracted from a KB or an entity annotated corpus. They encode knowledge about named entities themselves or their usages. Conceptually, we aim to incorporate the likelihood of a particular token being linked to an entity of a specific type. We implemented three features:

Type-infused Gazetteer Match. It finds the longest occurring token sequence in a type specific gazetteer. It adds a binary indicator to each token, depending on whether the token is part of a sequence. We use 30 dictionaries distributed by Ratinov and Roth, 2009 containing type-name information for English. For instance, "New York" is a place and "McDonald's" a corporation. These dictionaries have been successfully used in the past [14, 16, 18]. For the rest of the languages we generated the dictionaries automatically by mapping each dictionary to a set of YAGO types and extracting the corresponding names. For the dictionary containing corporations, for example, we incorporated all the names in the specific language corresponding to types *company* and *enterprise*.

Wikipedia Link Probability. This feature measures the likelihood of a token being linked to a named entity Wikipedia page. The intuition is that tokens linked to named entity pages tend to be indicative of named entities. For instance, the token "Obama" is usually linked while the term "box" is not. The list of pages referring to named entities is extracted from YAGO. Given a token in the text, it is assigned the probability of being linked according to Eq. 1, where $link_d(t)$ equals 1, if token t in document d is linked to another Wikipedia document. $present_d$ equals 1 if t occurs in d.

$$P_{Wiki}(t) = \frac{\sum_{d \in D} link_d(t)}{\sum_{d \in D} present_d(t)} \quad (1)$$

Since usually in Wikipedia only the first occurrence of a named entity is linked, we count a word on a page as linked if it links to a named entity page at least once.

Type Probability. Intended to discriminate between types, it encodes the likelihood of a token belonging to a given type. The idea is to capture the fact that, for instance, the token "Obama" is more

likely a person than a location. Since YAGO contains types and names for each entity, we can calculate the conditional probability.

Given a set of entities E with mentions M_e and tokens T_{em} we calculate the probability of a class $c \in C$ given a token t as

$$P(c|t) = \frac{\sum_e^E \sum_{m_e}^{M_e} \sum_{t_{em}}^{T_{em}} c(e)}{\sum_e^E \sum_{m_e}^{M_e} \sum_{t_{em}}^{T_{em}} \sum_{c_i}^{C} c_i(e)} \quad (2)$$

where $c(e) = 1$ if entity e belongs to class c and $c(e) = 0$ otherwise. For each token in the text, we create one feature per type with the respective probability as its value.

Token Type Position. Attempts to reflect that tokens may appear in different positions according to the entity type. For instance, "Supreme Court of the United States", is an organization and "United" occurs at the end. In "United States", a location, occurs at the beginning. This helps with named entities inside other named entities.

This idea is implemented using the BILOU (**B**egin, **I**nside, **L**ast, **O**utside, **U**nit) encoding [19], which tags each token with respect to the position in which it occurs (e.g., "**O**-The **B**-Supreme **I**-Court **I**-of **I**-the **I**-United **L**-States"). The number of features depends on the number of types in the dataset (4 BILU positions times n classes + O position). For each token, each feature receives the probability of a class given the token and position. The class probabilities are calculated as in Equation 2, incorporating also the token position. This strategy gives us the possibility to combine the class type probabilities with the token positions.

To the best of our knowledge, the last three features (*Token Type Position*, *Type Probability* and *Wikipedia Link Probability*) have not been used in previous work.

3.4 Entity-Based Knowledge

This category encodes document specific knowledge about the entities found in text. The idea is to exploit the inherent association between NER and NED. Previous work showed that the flow of information between the two tasks generates significant improvements in NER performance [14, 18].

Comparatively, this module requires more (computational and knowledge) resources than the previous ones. It requires a first run of NED to generate document specific features, based on the disambiguated named entities. The generated features are used in a second run of NER.

Following Radford et al. [18], after the first run of NED, we create a set of document-specific gazetteers derived from the named entities found. The idea is that this information will help in the second round to find new named entities missed in the first one. Take the sentence "Three-quarters of citizens of the European Union working in the United Kingdom would not meet current visa requirements for non-EU overseas workers if the uk left the bloc". We can imagine that in the first round of NED *European Union* and *United Kingdom* can be easily identified. However, "EU" or the wrongly capitalized "uk" might be missed. After the disambiguation, we know that both disambiguated entities are organizations and have the aliases *EU* and *UK* respectively. The idea is that if we introduce this information in a second NER run, they are easier to spot.

For each document we gather all entities that were disambiguated in the first NED run. Then we extract all surface forms of the identified entities from YAGO. The surface forms are tokenized and assigned the type of the corresponding entity plus its BILOU position. For example, the surface form "Barack Obama" will result in the two tokens "Barack" and "Obama", which will be assigned to "B-Person" and "L-Person" respectively. In KnowNER this feature is incorporated as 17 binary features (BILU tags multiplied by 4 coarse grained types + O tag), which fire when a token is encountered that is part of a list that contains the mappings from tokens to type–BILOU pairs.

4 EVALUATION

In this section, we analyze the effect of external knowledge (Sec. 4.2) and compare KnowNER with state-of-the-art approaches (Sec. 4.1) for three languages: English, German and Spanish.

4.1 Experimental Setup

KnowNER. The CRF was trained using CRF-suite [15] with the Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm and L1 regularization (coeff. = 1), which performed best on the English CoNLL2003 dev. set. We provide two settings for the system: (i) KnowNER$_{gold}$, which as Radford et al. [18] uses the gold standard named entity annotations for the Entity-based features and was used to analyze the impact of knowledge into the NER task, and (ii) KnowNER$_{aida}$ which runs AIDA to produce the Entity-based features and was used across all languages to compare with other available NER systems.

Datasets. We evaluated KnowNER on four well established datasets that provide annotated named entity mentions and types.

CoNLL2003e. By Sang and Meulder [20], it is a collection of English Reuter's newswires with named entity mentions annotated with types (i.e., persons, locations, organizations and miscellaneous). CoNLL2003e-dev does not include the developing set in training while CoNLL2003e-test does.

MUC-7. A set of New York Times articles (in English) [1] designed for NED and NER. It annotates named entities and their types

(i.e., organizations, persons, locations), dates, times, and quantities (monetary values, percentages). We only focused on the named entity types. MUC-7-dev does not include the developing set in training while MUC-7-test does.

CoNLL2003g. A German dataset also by Sang and Meulder [20], similar to CoNLL2003e the named entities are classified according to four types (i.e., persons, locations, organizations and miscellaneous). It consists of a collection of news articles from the Frankfurter Rundschau.

CoNLL2002. By Tjong Kim Sang [22], it is a collection of news wire articles in Spanish made available by the Spanish EFE News Agency. The named entities are classified into persons, organizations, locations, times and quantities. We only focus on the first three.

Metrics. We report F_1-score for all systems and two evaluation methodologies for our system and the other methods, when available.

Mention-based. It considers a named entity prediction as correct, if and only if the mention boundaries and predicted types are exact matches with the gold standard.

Span-based. Measures the correctness of mention boundaries ignoring type labels. This measure is important for applications which do not necessarily require type annotations such as NED.

Knowledge depth. To demonstrate the impact of increasing knowledge on NER performance we tested four variations of KnowNER, equivalent to the categories introduced in Sec. 3. Each variation contains the features corresponding to a category name plus all those from the lighter categories.

Agnostic: KnowNER$_A$ uses only local lexico-syntactic features without any external knowledge resources (Sec. 3.1).

Name-based: KnowNER$_{Name}$ uses features based on a list of names (Sec. 3.2) plus those in KnowNER$_A$.

KB-based: KnowNER$_{KB}$ utilizes features derived from a knowledge-base (Sec. 3.3) plus KnowNER$_{Name}$ features.

Entity-based: KnowNER$_{Entity}$ requires the execution of NED to generate document based features (Sec. 3.4) in addition to all the features in KnowNER$_{KB}$.

4.2 Incremental Knowledge

Here we analyze the impact of external knowledge on the system. Our detailed analysis is specific for the English language on the CoNLL2003e and MUC-7 datasets. Results show a clear improvement when deeper knowledge is used across datasets and entity types.

Fig. 1a shows the effect of the span detection in each category. Although it drops slightly for the name-based category, it quickly recovers as deeper knowledge is added. The effect is similar for both datasets.

Regarding the mention-based metric, Fig. 1b shows the effect of different knowledge categories for the Mention-based metric. In all cases adding knowledge generates a boost in performance. The effect is particularly strong for MUC-7-test which registered an overall increment of almost 10 F1 points. In both cases, the biggest boost is registered when the KB-based features are added.

However, the ablation study in Tab. 2 suggests that some KB-based features may be subsumed by the Entity-based ones which
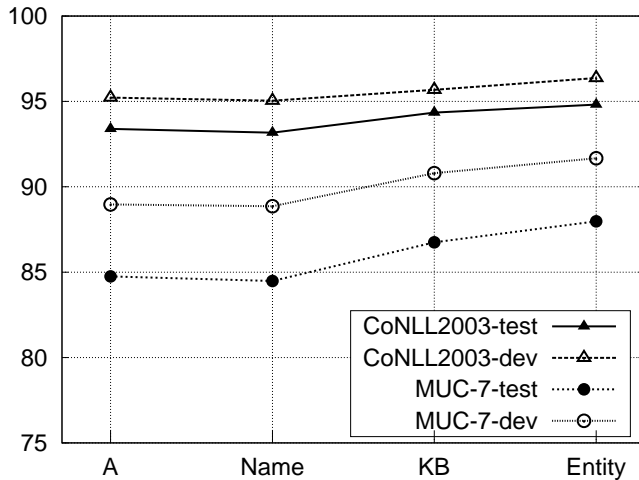
(a) Span-based F_1 score
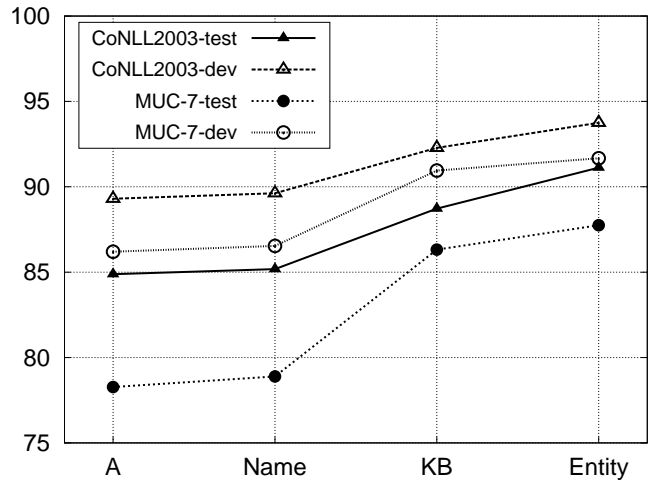
(b) Mention-based F_1 score

Figure 1: KnowNER$_{gold}$: Mention and span-based results for CoNLL2003e and MUC-7

generates the most significant boost. This is somehow expected as the entity specific information is extracted from the same KB and strongly relies on the entity types. The Entity-based component is also the most expensive concerning timing performance. Fig. 2 shows the time required by each setting, establishing a trade-off between accuracy and runtime. The Stanford agnostic system was faster than our implementation as it took 158.55 ms per document on average.

Feature Categories	F_1
A, Name, KB	88.73
A, Name, Entity	89.32
A, KB, Entity	91.09
All	91.12

Table 2: KnowNER$_{gold}$: Ablation study by categories on CoNLL2003e-test

Fig. 3a and Fig. 3b show the performance for each specific entity type for both CoNLL2003e-test and CoNLL2003e-dev. KnowNER achieves human-level performance for labelling persons (F_1 96.03 and F_1 95.86) and locations (F_1 92.13 and F_1 96.39). The positive effect of external knowledge is quite significant for organizations (F_1 80.86 to F_1 89.32 on test; F_1 83.94 to F_1 89.75 on dev) while it is relatively moderate for miscellaneous. In the case of MUC-7 (Fig. 3c and Fig. 3d), the effect is similar except for locations in MUC-7-test which tend to slightly drop when the entity-based category is used. The positive impact on persons for MUC-7-test is especially significative as it generates a change in ranking performance with respect to the other types. It jumps from the second and third position on MUC-7 test and MUC-7 dev with agnostic features to the very first position in MUC-7-test and the second on MUC-7-dev.

Finally, Tab. 3 and Tab. 4 display mention-based results on CoNLL2003e and MUC-7 for all knowledge categories and entity types. They also display the span-based performance for each
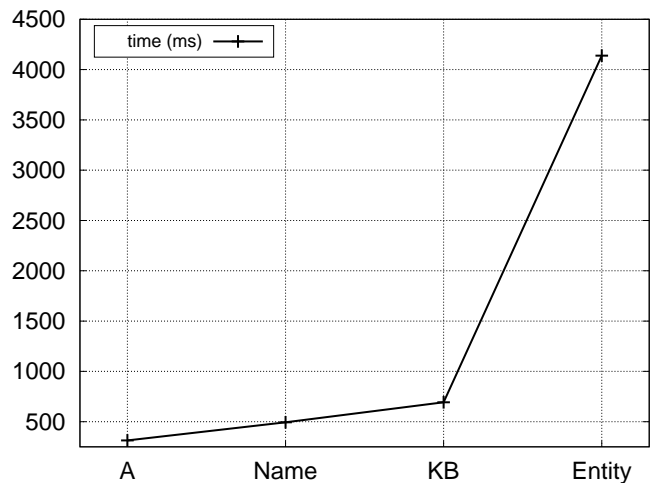

Figure 2: KnowNER$_{aida}$: Timing experiments for CoNLL2003e-test in average milliseconds per document

knowledge category. The numbers suggest that adding knowledge improves task performance.

4.3 Comparative Performance

English. KnowNER performs amongst state-of-the-art NER English systems on both datasets. Tab. 5 reports the results for mention-based performance on CoNLL2003e-test compared to the best-known systems. The results for KnowNER correspond to a setting using all the knowledge categories when using the gold standard for the entity-based step (as in Radford at al. [18]) or using the AIDA system for the entity-based knowledge category.

Tab. 6 displays detailed results for one of the latest versions of Finkel et al.[5] (Stanford NER 3.6.0), probably the most widely used NER system to date, which KnowNER outperforms.
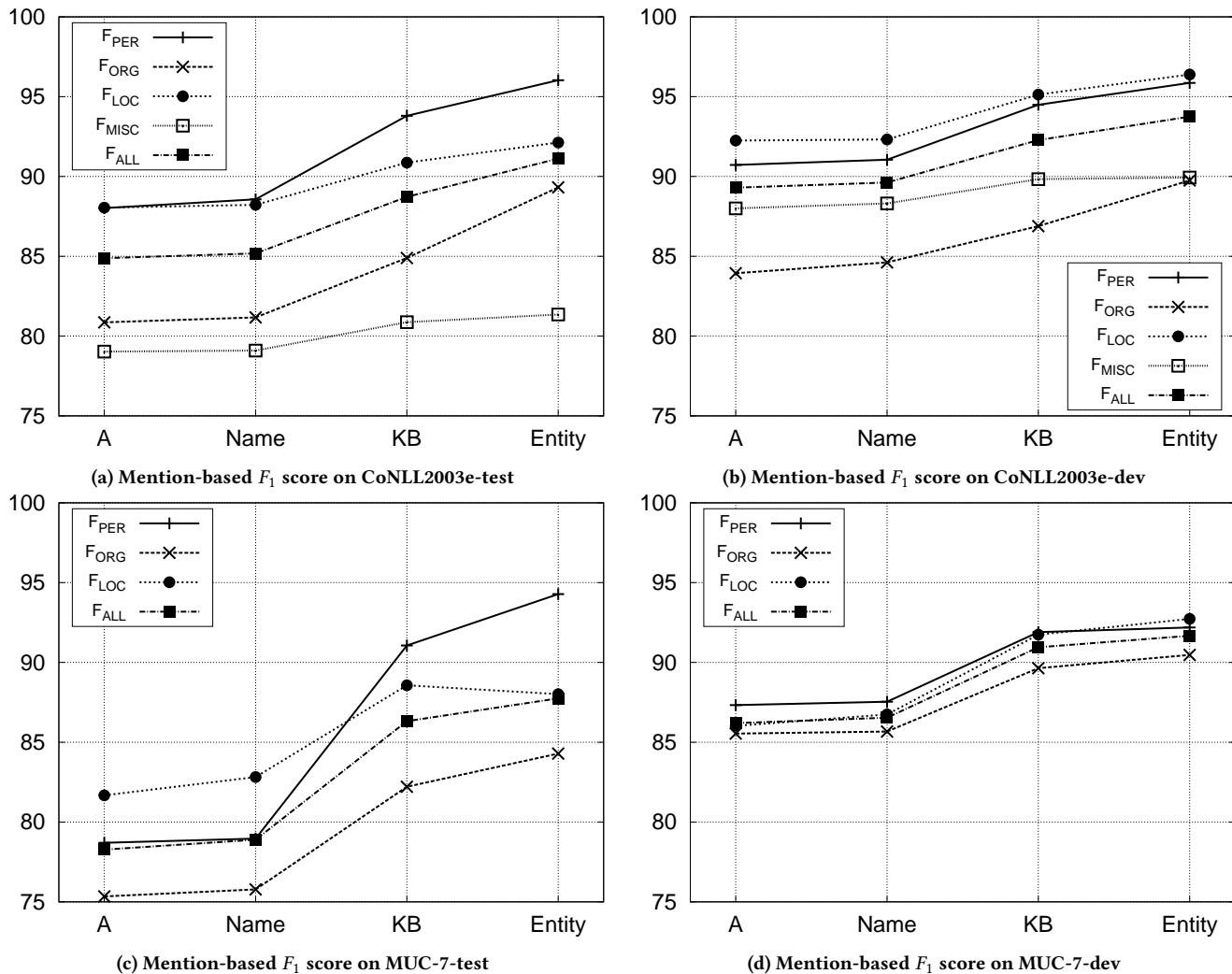
(a) Mention-based F_1 score on CoNLL2003e-test

(b) Mention-based F_1 score on CoNLL2003e-dev

(c) Mention-based F_1 score on MUC-7-test

(d) Mention-based F_1 score on MUC-7-dev

Figure 3: KnowNER$_{gold}$: Incremental knowledge results

Cat.	MUC-7 test					MUC-7 dev				
	F_{PER}	F_{ORG}	F_{LOC}	F_{ALL}	F_{SPAN}	F_{PER}	F_{ORG}	F_{LOC}	F_{ALL}	F_{SPAN}
A	78.70	75.34	81.67	78.27	84.75	87.33	85.54	86.03	86.20	88.96
Name	78.96	75.78	82.82	78.89	84.48	87.54	85.67	86.74	86.54	88.86
KB	91.06	82.21	88.57	86.32	86.75	91.90	89.64	91.74	90.95	90.80
Entity	94.28	84.29	88.01	87.75	87.97	92.19	90.47	92.73	91.67	91.67

Table 3: KnowNER$_{gold}$: Mention F_1 (knowledge category and type) and Span F_1 (knowledge category) on MUC-7.

German. To the best of our knowledge, KnowNER is one of the best performing systems to date for the German language on CoNLL2003g. Tab. 7 presents the results for KnowNER compared with state-of-the-art systems. Tab. 8 presents detailed results for each named entity type. The biggest boost in Germany is generated by the entity-based features, which generate an increment of more than 7 points in recall with respect to the previous knowledge category (i.e., kb-based).

Spanish. Tab. 9 presents the results for KnowNER compared with state-of-the-art systems for Spanish. Tab. 10 presents detailed results for each named entity type.

5 RELATED WORK

NER is a widely studied problem in the natural language understanding community. Well developed work has established a clear direction towards the use of CRFs [11] with systems achieving high

Cat.	CoNLL2003e test						CoNLL2003e dev					
	F_{PER}	F_{ORG}	F_{LOC}	F_{MISC}	F_{ALL}	F_{SPAN}	F_{PER}	F_{ORG}	F_{LOC}	F_{MISC}	F_{ALL}	F_{SPAN}
A	88.02	80.86	88.05	79.03	84.88	93.39	90.72	83.94	92.25	88.00	89.30	95.22
Name	88.57	81.17	88.22	79.09	85.18	93.17	91.05	84.62	92.32	88.31	89.62	95.04
KB	93.80	84.89	90.87	80.87	88.72	94.35	94.49	86.89	95.13	89.83	92.28	95.68
Entity	96.03	89.32	92.13	81.35	91.12	94.82	95.86	89.75	96.39	89.93	93.75	96.38

Table 4: KnowNER$_{gold}$: Mention F_1 (knowledge category and type) and Span F_1 (knowledge category) on CoNLL2003e.

System	F_1
Chiu and Nichols [2]	91.62
Luo et al. [14]	91.20
Yang et al. [23]	91.20
KnowNER$_{gold}$	**91.12**
Lample et al. [12]	90.94
Passos et al. [16]	90.90
Lin and Wu [13]	90.90
Ratinov and Roth [19]	90.80
KnowNER$_{aida}$	**90.16**
Radford et al. [18]	89.35
Finkel et al. [5]	86.86

Table 5: English: Mention-based performance on CoNLL2003e-test as reported in literature.

System	Type	CoNLL2003e test	CoNLL2003e dev
Stanford CoreNLP (3.6.0)	LOC	89.04	94.38
	MISC	81.51	87.44
	ORG	85.62	88.33
	PER	92.35	94.28
	All	88.05	91.95
KnowNER$_{aida}$	LOC	91.28	95.33
	MISC	81.82	88.59
	ORG	87.43	88.82
	PER	95.99	95.75
	All	90.16	93.11
KnowNER$_{gold}$	All	91.12	93.75

Table 6: English: F_1 Performance for the English language on CoNLL2003e and MUC-7 datasets.

System	F_1
Lample et al. [12]	78.76
KnowNER$_{aida}$	**77.20**
Gillick et al. [7]	76.22
Qi et al. [17]	75.72

Table 7: German: Mention-based performance on CoNLL2003g (German) as reported in literature.

Type	CoNLL2003g test	CoNLL2003g dev
LOC	77.24	78.03
MISC	68.79	72.59
ORG	65.58	75.65
PER	88.59	90.21
All	77.20	79.93

Table 8: German: KnowNER$_{aida}$ F_1 Performance for the German language on CoNLL2003g dataset.

System	F_1
Yang et al. [23]	85.77
Lample et al. [12]	85.75
KnowNER$_{aida}$	**83.79**
Gillick et al. [7]	82.95
dos Santos and Guimarães [3]	82.21

Table 9: Spanish: Mention-based performance on CoNLL2002 as reported in literature.

Type	CoNLL2002 test	CoNLL2002 dev
LOC	83.92	81.18
MISC	59.19	55.15
ORG	83.03	80.79
PER	94.34	93.48
All	83.79	82.14

Table 10: Spanish: KnowNER$_{aida}$ F_1 Performance for the Spanish language on CoNLL2002 dataset.

relative performance [5, 9, 14, 16, 18, 19]. A new line, focused on neural networks methods [2, 3, 7, 12, 23, 23]. Chiu and Nichols [2], for instance, the best NER system for English to date implemented a hybrid bidirectional LSTM-CNN whose inputs are tokens, word and character embeddings, and a set of gazetteers with type encodings.

Among the CRF methods, early work has focused on purely agnostic systems [5, 10]. Klein et al. [10] presents a system addressing the importance of substring features, an idea that we also capture in our agnostic model via prefixes and suffixes. Finkel et al. [5] is one of the most popular agnostic systems. Following this work, our agnostic category implements most of the features described in the

paper plus prefixes and suffixes used in Zhang and Johnson [24]. We do not make use of feature type statistics from the dataset which may explain a small drop in performance in our agnostic setting compared to Finkel et al. [5] in our English experiments. In contrast to agnostic approaches, our system strongly relies on background knowledge to improve performance.

Previous work has already regarded NER as a knowledge intensive task [6, 9, 13, 14, 16, 18, 19, 24]. Most of these works incorporate background knowledge in the form of entity-type gazetteers [6, 9, 16, 19, 24]. In fact, dictionaries were already provided for the early CoNLL2003 shared-task encouraging the use of external knowledge. Ratinov and Roth [19] used 30 gazetteers mostly extracted from Wikipedia, thereby generating big boosts in performance. These gazetteers have been successfully reused by other systems [14, 16, 18]. In particular, Luo et al. [14] used a total of 655 gazetteers including those from Ratinov and Roth [19]. We also incorporate gazetteers in our knowledge-based features. Finally, Kazama and Torisawa [9] was one of the first works to extract type information from Wikipedia. Their approach extracts category labels from the first sentence of the Wikipedia entity pages. In our method we also explicitly incorporate type information in the KB-based and the entity-based categories but in a cleaner way as they are derived from a high precision knowledge base like YAGO.

Compared to previous approaches using external knowledge, our method is more modular in the way the knowledge is incorporated. Our framework allows us to classify and easily derive more features. We have both more light-weight and knowledge-intensive features from different sources: entity names, knowledge bases and NED. Our distinctive features include frequent mention tokens, frequent mention shapes, frequent POS mention patterns, Wikipedia token probability and the class type probability, among others (Sec. 3).

The association between NER and NED has been successfully exploited by recent work [4, 14, 18] as a means to boost NER performance. Radford et al. [18] uses a two-step approach. They showed that local features derived from an initial NED run improve the performance on a second NER step. Specifically, alternative entity names and types tend to be important. We follow a similar two-step approach but, in contrast, we also run NED using a real world setting. Luo et al. [14] present a joint model for named entity recognition and disambiguation, as a CRF with a topology for joint optimization. NER and NED tasks are inherently associated so performing these tasks jointly poses natural advantages. However, NER has multiple applications apart from NED, which tends to be a computationally expensive task. Our modular approach, on top of a simpler and more tractable model, avoids expensive joint optimisation and permits to easily decouple the NED module for settings with small computational requirements. It also benefits from the mutual dependency between NER and NED when heavy computation is not an issue.

Regarding multilinguality, recent work has focused on methods to handle NER across a wide set of languages [7, 12, 23]. Yang et al. [23], one of the best systems across languages, implements a hierarchical recurrent neural network for joint POS tagging, chunking and NER, implemented on top of a CRF layer to do the labelling.

6 CONCLUSION

We presented KnowNER, a multilingual system that explicitly encodes different degrees of external knowledge for NER. KnowNER's framework defines four knowledge categories, each containing deeper external knowledge. Our experimental study shows that KnowNER performs among state-of-the-art NER systems across languages. It also shows that increasing the degree of external knowledge encoded in the system significantly boosts NER performance.

REFERENCES

[1] Nancy Chinchor and Patricia Robinson. 1997. MUC-7 named entity task definition. In *Proceedings of MUC-7*.
[2] Jason Chiu and Eric Nichols. 2016. Named Entity Recognition with Bidirectional LSTM-CNNs. *TACL*.
[3] Cícero Nogueira dos Santos and Victor Guimarães. 2015. Boosting Named Entity Recognition with Neural Character Embeddings. *Proceedings of the Fifth Named Entity Workshop* (2015).
[4] Greg Durrett and Dan Klein. 2014. A Joint Model for Entity Analysis: Coreference, Typing, and Linking. In *TACL*.
[5] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of ACL*.
[6] Radu Florian, Abraham Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named Entity Recognition through Classifier Combination. In *Proceedings of CoNLL*.
[7] Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2016. Multilingual Language Processing From Bytes. In *Proceedings of NAACL*.
[8] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust Disambiguation of Named Entities in Text. In *Proceedings of EMNLP*.
[9] Kazama Jun'ichi and Kentaro Torisawa. 2007. Exploiting Wikipedia as External Knowledge for Named Entity Recognition. In *Proceedings of EMNLP-CoNLL*.
[10] Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. 2003. Named Entity Recognition with Character-Level Models. In *Proceedings of CoNLL*.
[11] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of ICML*.
[12] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of NAACL*.
[13] Dekang Lin and Xiaoyun Wu. 2009. Phrase Clustering for Discriminative Learning. In *Proceedings of ACL*.
[14] Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. 2015. Joint Entity Recognition and Disambiguation. In *Proceedings of EMNLP*.
[15] Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs). (2007). http://www.chokkan.org/software/crfsuite/
[16] Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon Infused Phrase Embeddings for Named Entity Resolution. In *Proceedings of CoNLL*.
[17] Yanjun Qi, Ronan Collobert, Pavel P. Kuksa, Koray Kavukcuoglu, and Jason Weston. 2009. Combining labeled and unlabeled data with word-class distribution learning. In *Proceedings of CIKM*.
[18] Will Radford, Xavier Carreras, and James Henderson. 2015. Named entity recognition with document-specific KB tag gazetteers. In *Proceedings of EMNLP*.
[19] Lev-Arie Ratinov and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *Proceedings of CoNLL*.
[20] Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL*.
[21] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of WWW*.
[22] Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 Shared Task: Language-independent Named Entity Recognition. In *Proceedings of CoNLL*.
[23] Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2016. Multi-Task Cross-Lingual Sequence Tagging from Scratch. *CoRR* (2016).
[24] Tong Zhang and David Johnson. 2003. A Robust Risk Minimization based Named Entity Recognition System. In *Proceedings of CoNLL*.