A Survey on Recent Advances in Named Entity Recognition from Deep Learning models

Vikas Yadav
University of Arizona
vikasy@email.arizona.edu

Steven Bethard
University of Arizona
bethard@email.arizona.edu

Abstract

Named Entity Recognition (NER) is a key component in NLP systems for question answering, information retrieval, relation extraction, etc. NER systems have been studied and developed widely for decades, but accurate systems using deep neural networks (NN) have only been introduced in the last few years. We present a comprehensive survey of deep neural network architectures for NER, and contrast them with previous approaches to NER based on feature engineering and other supervised or semi-supervised learning algorithms. Our results highlight the improvements achieved by neural networks, and show how incorporating some of the lessons learned from past work on feature-based NER systems can yield further improvements.

1 Introduction

Named entity recognition is the task of identifying named entities like person, location, organization, drug, time, clinical procedure, biological protein, etc. in text. NER systems are often used as the first step in question answering, information retrieval, co-reference resolution, topic modeling, etc. Thus it is important to highlight recent advances in named entity recognition, especially recent neural NER architectures which have achieved state of the art performance with minimal feature engineering.

The first NER task was organized by Grishman and Sundheim (1996) in the Sixth Message Understanding Conference. Since then, there have been numerous NER tasks (Tjong Kim Sang and De Meulder, 2003; Tjong Kim Sang, 2002; Piskorski et al., 2017; Segura Bedmar et al., 2013; Bossy et al., 2013; Uzuner et al., 2011). Early NER systems were based on handcrafted rules, lexicons, orthographic features and ontologies. These systems were followed by NER systems based on feature-engineering and machine learning (Nadeau and Sekine, 2007). Starting with Collobert et al. (2011), neural network NER systems with minimal feature engineering have become popular. Such models are appealing because they typically do not require domain specific resources like lexicons or ontologies, and are thus poised to be more domain independent. Various neural architectures have been proposed, mostly based on some form of recurrent neural networks (RNN) over characters, sub-words and/or word embeddings.

We present a comprehensive survey of recent advances in named entity recognition. We describe knowledge-based and feature-engineered NER systems that combine in-domain knowledge, gazetteers, orthographic and other features with supervised or semi-supervised learning. We contrast these systems with neural network architectures for NER based on minimal feature engineering, and compare amongst the neural models with different representations of words and sub-word units. We show in Table 1 and Table 2 and discuss in Section 7 how neural NER systems have improved performance over past works including supervised, semi-supervised, and knowledge based NER systems. For example, NN models on news corpora improved the previous state-of-the-art by 1.59% in Spanish, 2.34% in German, 0.36% in English, and 0.14%, in Dutch, without any external resources or feature engineering. We provide resources, including links to shared tasks on NER, and links to the code for each category of NER system. To the best of our knowledge, this is the first survey focusing on neural architectures for NER, and comparing to previous feature-based systems.

Proceedings of the 27th International Conference on Computational Linguistics, pages 2145–2158
Santa Fe, New Mexico, USA, August 20-26, 2018.

We first discuss previous summary research on NER in section 2. Then we explain our selection criterion and methodology for selecting which systems to review in section 3. We highlight standard, past and recent NER datasets (from shared tasks and other research) in section 4 and evaluation metrics in section 5. We then describe NER systems in section 6 categorized into knowledge-based (section 6.1), bootstrapped (section 6.2), feature-engineered (section 6.3) and neural networks (section 6.4).

2 Previous surveys

The first comprehensive NER survey was Nadeau and Sekine (2007), which covered a variety of supervised, semi-supervised and unsupervised NER systems, highlighted common features used by NER systems during that time, and explained NER evaluation metrics that are still in use today. Sharnagat (2014) presented a more recent NER survey that also included supervised, semi-supervised, and unsupervised NER systems, and included a few introductory neural network NER systems. There have also been surveys focused on NER systems for specific domains and languages, including biomedical NER, (Leaman and Gonzalez, 2008), Chinese clinical NER (Lei et al., 2013), Arabic NER (Shaalan, 2014; Etaiwi et al., 2017), and NER for Indian languages (Patil et al., 2016).

The existing surveys primarily cover feature-engineered machine learning models (including supervised, semi-supervised, and unsupervised systems), and mostly focus on a single language or a single domain. There is not yet, to our knowledge, a comprehensive survey of modern neural network NER systems, nor is there a survey that compares feature engineered and neural network systems in both multi-lingual (CoNLL 2002 and CoNLL 2003) and multi-domain (e.g., news and medical) settings.

3 Methodology

To identify articles for this survey, we searched Google, Google Scholar, and Semantic Scholar. Our query terms included *named entity recognition*, *neural architectures for named entity recognition*, *neural network based named entity recognition models*, *deep learning models for named entity recognition*, etc. We sorted the papers returned from each query by citation count and read at least the top three, considering a paper for our survey if it either introduced a neural architecture for named entity recognition, or represented a top-performing model on an NER dataset. We included an article presenting a neural architecture only if it was the first article to introduce the architecture; otherwise, we traced citations back until we found the original source of the architecture. We followed the same approach for feature-engineering NER systems. We also included articles that implemented these systems for different languages or domain. In total, 154 articles were reviewed and 83 articles were selected for the survey.

4 NER datasets

Since the first shared task on NER (Grishman and Sundheim, 1996)[1], many shared tasks and datasets for NER have been created. CoNLL 2002 (Tjong Kim Sang, 2002)[2] and CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003)[3] were created from newswire articles in four different languages (Spanish, Dutch, English, and German) and focused on 4 entities - PER (person), LOC (location), ORG (organization) and MISC (miscellaneous including all other types of entities).

NER shared tasks have also been organized for a variety of other languages, including Indian languages (Rajeev Sangal and Singh, 2008), Arabic (Shaalan, 2014), German (Benikova et al., 2014), and slavic languages (Piskorski et al., 2017). The named entity types vary widely by source of dataset and language. For example, Rajeev Sangal and Singh (2008)'s southeast Asian language data has named entity types person, designation, temporal expressions, abbreviations, object number, brand, etc. Benikova et al. (2014)'s data, which is based on German wikipedia and online news, has named entity types similar to that of CoNLL 2002 and 2003: PERson, ORGanization, LOCation and OTHer. The shared task[4] or-

[1]Shared task: https://www-nlpir.nist.gov/related_projects/muc/
[2]Shared task: https://www.clips.uantwerpen.be/conll2002/ner/
[3]Shared task: https://www.clips.uantwerpen.be/conll2003/ner/
[4]Shared task: http://bsnlp.cs.helsinki.fi/shared_task.html

ganized by Piskorski et al. (2017) covering 7 slavic languages (Croatian, Czech, Polish, Russian, Slovak, Slovene, Ukrainian) also has person, location, organization and miscellaneous as named entity types.

In the biomedical domain, Kim et al. (2004) organized a BioNER task on MedLine abstracts, focusing on protien, DNA, RNA and cell attribute entity types. Uzuner et al. (2007) presented a clinical note de-identification task that required NER to locate personal patient data phrases to be anonymized. The 2010 I2B2 NER task[5] (Uzuner et al., 2011), which considered clinical data, focused on clinical problem, test and treatment entity types. Segura Bedmar et al. (2013) organized a Drug NER shared task[6] as part of SemEval 2013 Task 9, which focused on drug, brand, group and drug_n (unapproved or new drugs) entity types. (Krallinger et al., 2015) introduced the similar CHEMDNER task [7] focusing more on chemical and drug entities like trivial, systematic, abbreviation, formula, family, identifier, etc. Biology and microbiology NER datasets[8] (Hirschman et al., 2005; Bossy et al., 2013; Delėger et al., 2016) have been collected from PubMed and biology websites, and focus mostly on bacteria, habitat and geolocation entities. In biomedical NER systems, segmentation of clinical and drug entities is considered to be a difficult task because of complex orthographic structures of named entities (Liu et al., 2015).

NER tasks have also been organized on social media data, e.g., Twitter, where the performance of classic NER systems degrades due to issues like variability in orthography and presence of grammatically incomplete sentences (Baldwin et al., 2015). Entity types on Twitter are also more variable (person, company, facility, band, sportsteam, movie, TV show, etc.) as they are based on user behavior on Twitter.

Though most named entity annotations are flat, some datasets include more complex structures. Ohta et al. (2002) constructed a dataset of nested named entities, where one named entity can contain another. Strassel et al. (2003) highlighted both entity and entity head phrases. And discontinuous entities are common in chemical and clinical NER datasets (Krallinger et al., 2015). Eltyeb and Salim (2014) presented an survey of various NER systems developed for such NER datasets with a focus on chemical NER.

5 NER evaluation metrics

Grishman and Sundheim (1996) scored NER performance based on *type*, whether the predicted label was correct regardless of entity boundaries, and *text*, whether the predicted entity boundaries were correct regardless of the label. For each score category, *precision* was defined as the number of entities a system predicted correctly divided by the number that the system predicted, recall was defined as the number of entities a system predicted correctly divided by the number that were identified by the human annotators, and (micro) F-score was defined as the harmonic mean of precision and recall from both type and text.

The *exact match* metrics introduced by CoNLL (Tjong Kim Sang and De Meulder, 2003; Tjong Kim Sang, 2002) considers a prediction to be correct only when the predicted label for the complete entity is matched to exactly the same words as the gold label of that entity. CoNLL also used (micro) F-score, taking the harmonic mean of the exact match precision and recall.

The *relaxed F1* and *strict F1* metrics have been used in many NER shared tasks (Segura Bedmar et al., 2013; Krallinger et al., 2015; Bossy et al., 2013; Delėger et al., 2016). Relaxed F1 considers a prediction to be correct as long as part of the named entity is identified correctly. Strict F1 requires the character offsets of a prediction and the human annotation to match exactly. In these data, unlike CoNLL, word offsets are not given, so relaxed F1 is intended to allow comparison despite different systems having different word boundaries due to different segmentation techniques (Liu et al., 2015).

6 NER systems

6.1 Knowledge-based systems

Knowledge-based NER systems do not require annotated training data as they rely on lexicon resources and domain specific knowledge. These work well when the lexicon is exhaustive, but fail, for example, on every example of the drug_n class in the DrugNER dataset (Segura Bedmar et al., 2013), since drug_n

[5]Shared task: `https://www.i2b2.org/NLP/Relations/`
[6]Shared task: `https://www.cs.york.ac.uk/semeval-2013/task9/index.html`
[7]Similar datasets can be found here: `http://www.biocreative.org`
[8]Shared task: `http://2016.bionlp-st.org/tasks/bb2`

is defined as unapproved or new drugs, which are by definition not in the DrugBank dictionaries (Knox et al., 2010). Precision is generally high for knowledge-based NER systems because of the lexicons, but recall is often low due to domain and language-specific rules and incomplete dictionaries. Another drawback of knowledge based NER systems is the need of domain experts for constructing and maintaining the knowledge resources.

6.2 Unsupervised and bootstrapped systems

Some of the earliest systems required very minimal training data. Collins and Singer (1999) used only labeled seeds, and 7 features including orthography (e.g., capitalization), context of the entity, words contained within named entities, etc. for classifying and extracting named entities. Etzioni et al. (2005) proposed an unsupervised system to improve the recall of NER systems applying 8 generic pattern extractors to open web text, e.g., *NP is a <class1>*, *NP1 such as NPList2*. Nadeau et al. (2006) presented an unsupervised system for gazetteer building and named entity ambiguity resolution based on Etzioni et al. (2005) and Collins and Singer (1999) that combined an extracted gazetteer with commonly available gazetteers to achieve F-scores of 88%, 61%, and 59% on MUC-7 (Chinchor and Robinson, 1997) location, person, and organization entities, respectively.

Zhang and Elhadad (2013) used shallow syntactic knowledge and inverse document frequency (IDF) for an unsupervised NER system on biology (Kim et al., 2004) and medical (Uzuner et al., 2011) data, achieving 53.8% and 69.5% accuracy, respectively. Their model uses seeds to discover text having potential named entities, detects noun phrases and filters any with low IDF values, and feeds the filtered list to a classifier (Alfonseca and Manandhar, 2002) to predict named entity tags.

6.3 Feature-engineered supervised systems

Supervised machine learning models learn to make predictions by training on example inputs and their expected outputs, and can be used to replace human curated rules. Hidden Markov Models (HMM), Support Vector Machines (SVM), Conditional Random Fields (CRF), and decision trees were common machine learning systems for NER.

Zhou and Su (2002) used HMM (Rabiner and Juang, 1986; Bikel et al., 1997) an NER system on MUC-6 and MUC-7 data, achieving 96.6% and 94.1% F score, respectively. They included 11 orthographic features (1 numeral, 2 numeral, 4 numeral, all caps, numerals and alphabets, contains underscore or not, etc.) a list of trigger words for the named entities (e.g., 36 trigger words and affixes, like *river*, for the location entity class), and a list of words (10000 for the person entity class) from various gazetteers.

Malouf (2002) compared the HMM with Maximum Entropy (ME) by adding multiple features. Their best model included capitalization, whether a word was the first in a sentence, whether a word had appeared before with a known last name, and 13281 first names collected from various dictionaries. The model achieved 73.66%, 68.08% Fscore on Spanish and Dutch CoNLL 2002 dataset respectively.

The winner of CoNLL 2002 (Carreras et al., 2002) used binary AdaBoost classifiers, a boosting algorithm that combines small fixed-depth decision trees (Schapire, 2013). They used features like capitalization, trigger words, previous tag prediction, bag of words, gazetteers, etc. to represent simple binary relations and these relations were used in conjunction with previously predicted labels. They achieved 81.39% and 77.05% F scores on the Spanish and Dutch CoNLL 2002 datasets, respectively.

Li et al. (2005) implemented a SVM model on the CoNLL 2003 dataset and CMU seminar documents. They experimented with multiple window sizes, features (orthographic, prefixes suffixes, labels, etc.) from neighboring words, weighting neighboring word features according to their position, and class weights to balance positive and negative class. They used two SVM classifiers, one for detecting named entity starts and one for detecting ends. They achieved 88.3% F score on the English CoNLL 2003 data.

On the MUC6 data, Takeuchi and Collier (2002) used part-of-speech (POS) tags, orthographic features, a window of 3 words to the left and to the right of the central word, and tags of the last 3 words as features to the SVM. The final tag was decided by the voting of multiple one-vs-one SVM outputs.

Ando and Zhang (2005a) implemented structural learning (Ando and Zhang, 2005b) to divide the main task into many auxiliary tasks, for example, predicting labels by looking just at the context and masking

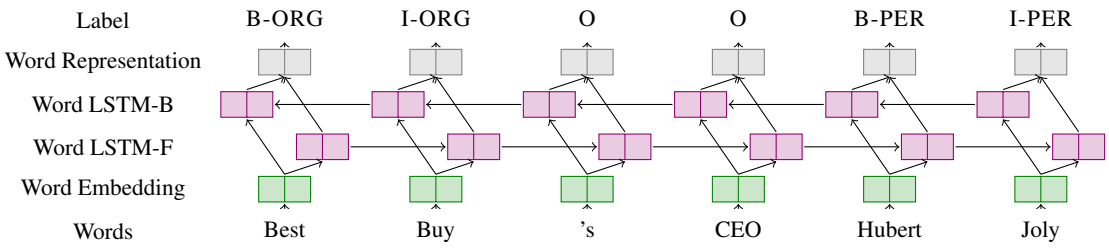

Figure 1: Word level NN architecture for NER

the current word. The best classifier for each auxiliary task was selected based on its confidence. This model had achieved 89.31% and 75.27% F score on English and German, respectively.

Agerri and Rigau (2016) developed a semi-supervised system[9] by presenting NER classifiers with features including orthography, character n-grams, lexicons, prefixes, suffixes, bigrams, trigrams, and unsupervised cluster features from the Brown corpus, Clark corpus and k-means clustering of open text using word embeddings (Mikolov et al., 2013). They achieved near state of the art performance on CoNLL datasets: 84.16%, 85.04%, 91.36%, 76.42% on Spanish, Dutch, English, and German, respectively.

In DrugNER (Segura Bedmar et al., 2013), Liu et al. (2015) achieved state-of-the-art results by using a CRF with features like lexicon resources from Food and Drug Administration (FDA), DrugBank, Jochem (Hettne et al., 2009) and word embeddings (trained on a MedLine corpus). For the same task, Rocktäschel et al. (2013) used a CRF with features constructed from dictionaries (e.g., Jochem (Hettne et al., 2009)), ontologies (ChEBI ontologies), prefixes-suffixes from chemical entities, etc.

6.4 Feature-inferring neural network systems

Collobert and Weston (2008) proposed one of the first neural network architectures for NER, with feature vectors constructed from orthographic features (e.g., capitalization of the first character), dictionaries and lexicons. Later work replaced these manually constructed feature vectors with word embeddings (Collobert et al., 2011), which are representations of words in n-dimensional space, typically learned over large collections of unlabeled data through an unsupervised process such as the skip-gram model (Mikolov et al., 2013). Studies have shown the importance of such pre-trained word embeddings for neural network based NER systems (Habibi et al., 2017), and similarly for pre-trained character embeddings in character-based languages like Chinese (Li et al., 2015; Yin et al., 2016).

Modern neural architectures for NER can be broadly classified into categories depending upon their representation of the words in a sentence. For example, representations may be based on words, characters, other sub-word units or any combination of these.

6.4.1 Word level architectures

In this architecture, the words of a sentence are given as input to Recurrent Neural Networks (RNN) and each word is represented by its word embedding, as shown in Figure 1.

The first word-level NN model was proposed by Collobert et al. (2011)[10]. The architecture was similar to the one shown in Figure 1, but a convolution layer was used instead of the Bi-LSTM layer and the output of the convolution layer was given to a CRF layer for the final prediction. The authors achieved 89.59% F1 score on English CoNLL 2003 dataset by including gazetteers and SENNA embeddings.

Huang et al. (2015) presented a word LSTM model (Figure 1) and showed that adding a CRF layer to the top of the word LSTM improved performance, achieving 84.26% F1 score on English CoNLL 2003 dataset. Similar systems were applied to other domains: DrugNER by Chalapathy et al. (2016) achieving 85.19% F1 score (under an unofficial evaluation) on MedLine test data (Segura Bedmar et al., 2013), and medical NER by Xu et al. (2017) achieving 80.22% F1 on disease NER corpus using this architecture. In similar tasks, Plank et al. (2016) implemented the same model for multilingual POS tagging.

[9]Code: `https://github.com/ixa-ehu/ixa-pipe-nerc`
[10]Code: `https://ronan.collobert.com/senna/`

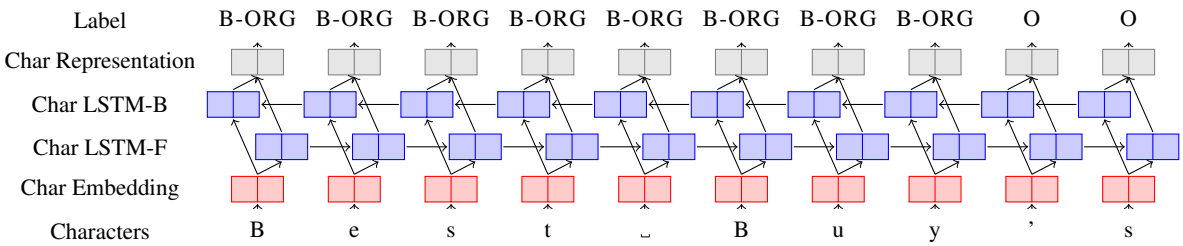

Figure 2: Character level NN architecture for NER

With slight variations, Yan et al. (2016) implemented word level feed forward NN, bi-directional LSTM (bi-LSTM) and window bi-LSTM for NER of English, German and Arabic. They also highlighted the performance improvement after adding various features like CRF, case, POS, word embeddings and achieved 88.91% F1 score on English and 76.12% on German.

6.4.2 Character level architectures

In this model, a sentence is taken to be a sequence of characters. This sequence is passed through an RNN, predicting labels for each character (Figure 2). Character labels transformed into word labels via post processing. The potential of character NER neural models was first highlighted by Kim et al. (2016) using highway networks over convolution neural networks (CNN) on character sequences of words and then using another layer of LSTM + softmax for the final predictions.

This model was implemented by Pham and Le-Hong (2017) for Vietnamese NER and achieved 80.23% F-score on Nguyen et al. (2016)'s Vietnamese test data. Character models were also used in various other languages like Chinese (Dong et al., 2016) where it has achieved near state of the art performance.

Kuru et al. (2016) proposed CharNER [11] which implemented the character RNN model for NER on 7 different languages. In this character model, tag prediction over characters were converted to word tags using Viterbi decoder(Forney, 1973) achieving 82.18% on Spanish, 79.36% on Dutch, 84.52% on English and 70.12% on German CoNLL datasets. They also achieved 78.72 on Arabic, 72.19 on Czech and 91.30 on Turkish. Ling et al. (2015) proposed word representation using RNN (Bi-LSTM) over characters of the word and achieved state of the art results on POS task using this representation in multiple languages including 97.78% accuracy on English PTB(Marcus et al., 1993).

Gillick et al. (2015) implemented sequence to sequence model (Byte to Span- BTS) using encoder decoder architecture over sequence of characters of words in a window of 60 characters. Each character was encoded in bytes and BTS achieved high performance on CoNLL 2002 and 2003 dataset without any feature engineering. BTS achieved 82.95%, 82.84%,86.50%,76.22% Fscore on Spanish, Dutch, English and German CoNLL datasets respectively.

6.4.3 Character+Word level architectures

Systems combining word context and the characters of a word have proved to be strong NER systems that need little domain specific knowledge or resources. There are two base models in this category. The **first type of model** represents words as a combination of a word embedding and a convolution over the characters of the word, follows this with a Bi-LSTM layer over the word representations of a sentence, and finally uses a softmax or CRF layer over the Bi-LSTM to generate labels. The architecture diagram for this model is same as Figure 3 but with the character Bi-LSTM replaced with a CNN[12].

Ma and Hovy (2016) implemented this model to achieve 91.21% F1 score on the CoNLL 2003 English dataset and 97.55% POS-tagging accuracy on the WSJ portion of PTB (Marcus et al., 1993). They also showed lower performance by this model for out of vocabulary words.

Chiu and Nichols (2015) achieved 91.62% F1 score on the CoNLL 2003 English dataset and 86.28% F score on Onto notes 5.0 dataset (Pradhan et al., 2013) by adding lexicons and capitalization features to

[11]Code: `https://github.com/ozanarkancan/char-ner`
[12]Code: `https://github.com/LopezGG/NN_NER_tensorFlow`

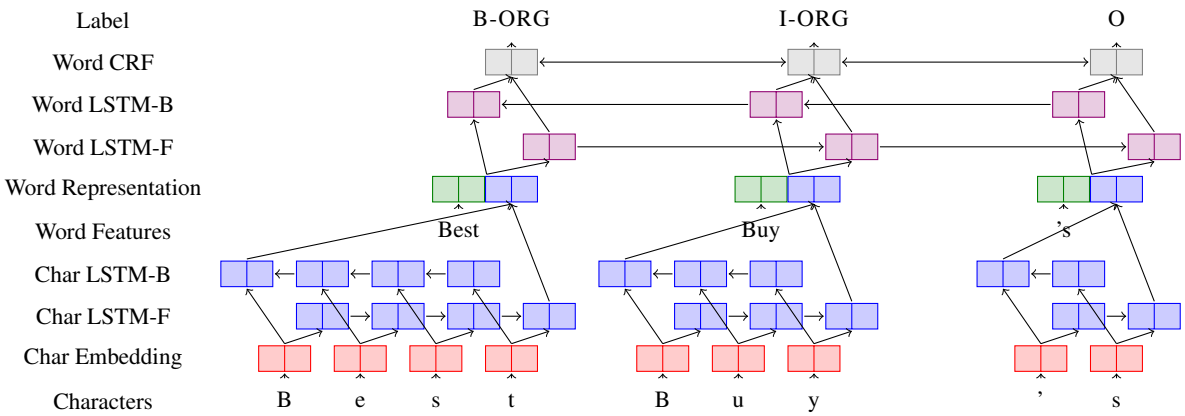

Figure 3: Word+character level NN architecture for NER

this model. Lexicon feature were encoded in the form or B(begin), I(inside) or E(end) PER, LOC, ORG and MISC depending upon the match from the dictionary.

This model has also been utilized for NER in languages like Japanese where Misawa et al. (2017) showed that this architecture outperformed other neural architectures on the *organization* entity class.

Limsopatham and Collier (2016) implemented a character+word level NER model for Twitter NER (Baldwin et al., 2015) by concatenating a CNN over characters, a CNN over orthographic features of characters, a word embedding, and a word orthographic feature embedding. This concatenated representation is passed through another Bi-LSTM layer and the output is given to CRF for predicting. This model achieved 65.89% F score on segmentation alone and 52.41% F score on segmentation and categorization.

Santos and Guimaraes (2015) implemented a model with a CNN over the characters of word, concatenated with word embeddings of the central word and its neighbors, fed to a feed forward network, and followed by the Viterbi algorithm to predict labels for each word. The model achieved 82.21% F score on Spanish CoNLL 2002 data and 71.23% F score on Portuguese NER data (Santos and Cardoso, 2007).

The **second type of model** concatenates word embeddings with LSTMs (sometimes bi-directional) over the characters of a word, passing this representation through another sentence-level Bi-LSTM, and predicting the final tags using a final softmax or CRF layer (Figure 3). Lample et al. (2016)[13] introduced this architecture and achieved 85.75%, 81.74%, 90.94%, 78.76% Fscores on Spanish, Dutch, English and German NER dataset respectively from CoNLL 2002 and 2003.

Dernoncourt et al. (2017) implemented this model in the NeuroNER toolkit[14] with the main goal of providing easy usability and allowing easy plotting of real time performance and learning statistics of the model. The BRAT annotation tool[15] is also integrated with NeuroNER to ease the development of NN NER models in new domains. NeuroNER achieved 90.50% F score on the English CoNLL 2003 data.

Habibi et al. (2017) implemented the model for various biomedical NER tasks and achieved higher performance than the majority of other participants. For example, they achieved 83.71 F-score on the CHEMDNER data (Krallinger et al., 2015).

Bharadwaj et al. (2016)[16] utilized phonemes (from Epitran) for NER in addition to characters and words. They also utilize attention knowledge over sequence of characters in word which is concatenated with the word embedding and character representation of word. This model achieved state of the art performance (85.81% F score) on Spanish CoNLL 2002 dataset.

A slightly improved system focusing on multi-task and multi-lingual joint learning was proposed by Yang et al. (2016) where word representation given by GRU (Gated Recurrent Unit) cell over characters plus word embedding was passed through another RNN layer and the output was given to CRF models trained for different tasks like POS, chunking and NER. Yang et al. (2017) further proposed transfer

[13]Code: `https://github.com/glample/tagger`
[14]Code: `http://neuroner.com`
[15]Code: `http://brat.nlplab.org/`
[16]Code: `https://github.com/dmort27/epitran`

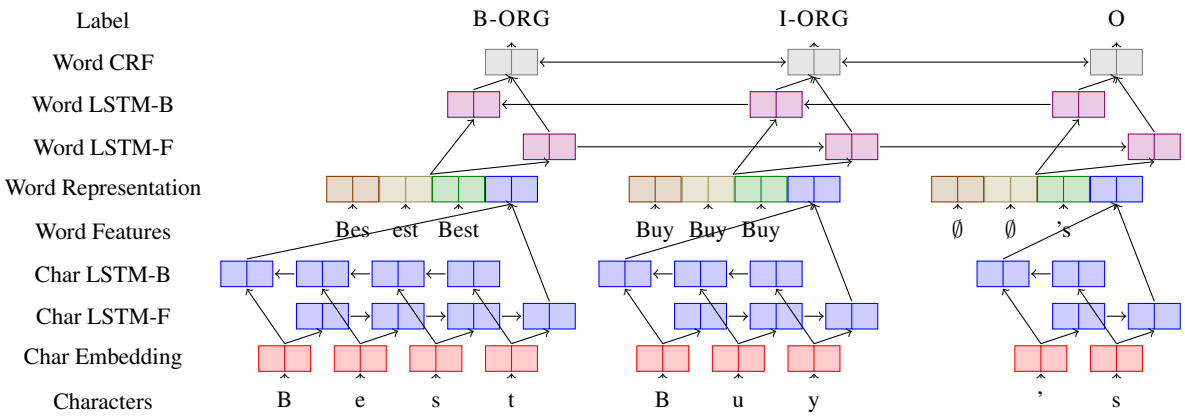

Figure 4: Word+character+affix level NN architecture for NER

learning for multi-task and multi-learning, and showed small improvements on CoNLL 2002 and 2003 NER data, achieving 85.77%, 85.19%, 91.26% F scores on Spanish, Dutch and English, respectively.

6.4.4 Character + Word + affix model

Yadav et al. (2018) implemented a model that augments the character+word NN architecture with one of the most successful features from feature-engineering approaches: affixes. Affix features were used in early NER systems for CoNLL 2002 (Tjong Kim Sang, 2002; Cucerzan and Yarowsky, 2002) and 2003 (Tjong Kim Sang and De Meulder, 2003) and for biomedical NER (Saha et al., 2009), but had not been used in neural NER systems. They extended the Lample et al. (2016) character+word model to learn affix embeddings[17] alongside the word embeddings and character RNNs (Figure 4). They considered all n-gram prefixes and suffixes of words in the training corpus, and selected only those whose frequency was above a threshold, T. Their word+character+affix model achieved 87.26%, 87.54%, 90.86%, 79.01% on Spanish, Dutch, English and German CoNLL datasets respectively. Yadav et al. (2018) also showed that affix embeddings capture complementary information to that captured by RNNs over the characters of a word, that selecting only high frequency (realistic) affixes was important, and that embedding affixes was better than simply expanding the other embeddings to reach a similar number of hyper-parameters.

7 Discussion

Table 1 shows the results of all the different categories of systems discussed in section 6 on the CoNLL 2002 and 2003 datasets. The table also indicates, for each model, whether it makes use of external knowledge like a dictionary or gazetteer. Table 2 presents a similar analysis on the DrugNER dataset from SemEval 2013 task 9 (Segura Bedmar et al., 2013).

Our first finding from the survey is that feature-inferring NN systems outperform feature-engineered systems, despite the latter's access to domain specific rules, knowledge, features, and lexicons. For example, the best feature-engineered system for Spanish, Agerri and Rigau (2016), is 1.59% below the best feature-inferring neural network system, (Lample et al., 2016), and 1.65% below the best neural network system that incorporates lexical resources (Bharadwaj et al., 2016). Similarly, the best feature-engineered system for German, Agerri and Rigau (2016), is 2.34% below the best feature-inferring neural network system, Lample et al. (2016). The differences are smaller for Dutch and English, but in neither case is the best feature-engineered model better than the best neural network model. In DrugNER, the word+character NN model outperforms the feature engineered system by 8.90% on MedLine test data and 3.50% on the overall dataset.

Our next finding is that word+character hybrid models are generally better than both word-based and character-based models. For example, the best hybrid NN model for English, Chiu and Nichols (2015), is 0.52% better than the best word-based model, Huang et al. (2015), and 5.12% better than the best character-based model, (Kuru et al., 2016). Similarly, the best hybrid NN model for German, Lample et

[17]Code: `https://github.com/vikas95/Pref_Suff_Span_NN`

Feature-engineered machine learning systems	Dict	SP	DU	EN	GE
Carreras et al. (2002) binary AdaBoost classifiers	Yes	81.39	77.05	-	-
Malouf (2002) - Maximum Entropy (ME) + features	Yes	73.66	68.08	-	-
Li et al. (2005) SVM with class weights	Yes	-	-	88.3	-
Passos et al. (2014) CRF	Yes	-	-	90.90	-
Ando and Zhang (2005a) Semi-supervised state of the art	No	-	-	89.31	75.27
Agerri and Rigau (2016)	Yes	**84.16**	**85.04**	**91.36**	**76.42**
Feature-inferring neural network word models					
Collobert et al. (2011) Vanilla NN +SLL / Conv-CRF	No	-	-	81.47	-
Huang et al. (2015) Bi-LSTM+CRF	No	-	-	84.26	-
Yan et al. (2016) Win-BiLSTM (English), FF (German) (Many fets)	Yes	-	-	88.91	**76.12**
Collobert et al. (2011) Conv-CRF (SENNA+Gazetteer)	Yes	-	-	89.59	-
Huang et al. (2015) Bi-LSTM+CRF+ (SENNA+Gazetteer)	Yes	-	-	**90.10**	-
Feature-inferring neural network character models					
Gillick et al. (2015) – BTS	No	**82.95**	**82.84**	**86.50**	**76.22**
Kuru et al. (2016) CharNER	No	82.18	79.36	84.52	70.12
Feature-inferring neural network word + character models					
Yang et al. (2017)	Yes	85.77	**85.19**	91.26	-
Luo (2015)	Yes	-	-	91.20	-
Chiu and Nichols (2015)	Yes	-	-	**91.62**	-
Ma and Hovy (2016)	No	-	-	91.21	-
Santos and Guimaraes (2015)	No	82.21	-	-	-
Lample et al. (2016)	No	85.75	81.74	90.94	**78.76**
Bharadwaj et al. (2016)	Yes	**85.81**	-	-	-
Dernoncourt et al. (2017)	No	-	-	90.5	-
Feature-inferring neural network word + character + affix models					
Re-implementation of Lample et al. (2016) (100 Epochs)	No	85.34	85.27	90.24	78.44
Yadav et al. (2018)(100 Epochs)	No	86.92	87.50	90.69	78.56
Yadav et al. (2018) (150 Epochs)	No	**87.26**	**87.54**	90.86	**79.01**

Table 1: Comparison of NER systemsin four languages: CoNLL 2002 Spanish (SP), CoNLL 2002 Dutch (DU), CoNLL 2003 English (EN), and CoNLL 2003 German (GE). Dict indicates whether or not the approach makes use of dictionary lookups. Best performance in each category is highlighted in bold.

		MedLine (80.10%)			DrugBank (19.90%)			Complete dataset		
	Dict	P	R	F1	P	R	F1	P	R	F1
Feature-engineered machine learning systems										
Rocktäschel et al. (2013)	Yes	60.70	55.80	58.10	88.10	87.50	87.80	73.40	69.80	71.50
Liu et al. (2015) (baseline)	No	-	-	-	-	-	-	78.41	67.78	72.71
Liu et al. (2015) (MED. emb.)	No	-	-	-	-	-	-	82.70	69.68	75.63
Liu et al. (2015) (state of the art)	Yes	78.77	60.21	68.25	90.60	88.82	**89.70**	84.75	72.89	**78.37**
NN word model										
Chalapathy et al. (2016) (relaxed performance)	No	52.93	52.57	52.75	87.07	83.39	85.19	-	-	-
NN word + character model										
Yadav et al. (2018)	No	73	62	67	87	86	87	79	72	75
NN word + character + affix model										
Yadav et al. (2018)	No	74	64	**69**	89	86	87	81	74	77

Table 2: DrugNER results on the MedLine and DrugBank test data (80.10% and 19.90% of the test data, respectively). The Yadav et al. (2018) experiments report no decimal places because they were run after the end of shared task, and the official evaluation script outputs no decimal places.

al. (2016), is 2.64% better than the best word-based model, Yan et al. (2016), and 2.54% better than the best character-based model, (Kuru et al., 2016). In DrugNER, the word+character hybrid model is better than the word model by 14.25% on MedLine test data and 1.81% on DrugBank test data.

Our final finding is that there is still interesting progress to be made by incorporating key features of past feature-engineered models into modern NN architectures. Yadav et al. (2018)'s simple extension

of Lample et al. (2016) to incorporate affix features yields a very strong new model, achieving a new state-of-the-art in Spanish, Dutch, and German, and performing within 1% of the best model for English.

8 Conclusion

Our survey of models for named entity recognition, covering both classic feature-engineered machine learning models, and modern feature-inferring neural network models has yielded several important insights. Neural network models generally outperform feature-engineered models, character+word hybrid neural networks generally outperform other representational choices, and further improvements are available by applying past insights to current neural network models, as shown by the state-of-the-art performance of our proposed affix-based extension of character+word hybrid models.

References

Rodrigo Agerri and German Rigau. 2016. Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

Enrique Alfonseca and Suresh Manandhar. 2002. An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st international conference on general WordNet, Mysore, India*, pages 34–43.

Rie Kubota Ando and Tong Zhang. 2005a. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853.

Rie Kubota Ando and Tong Zhang. 2005b. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853.

Timothy Baldwin, Marie-Catherine de Marneffe, Bo Han, Young-Bum Kim, Alan Ritter, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. Nosta-d named entity annotation for german: Guidelines and dataset. In *LREC*, pages 2524–2531.

Akash Bharadwaj, David R. Mortensen, Chris Dyer, and Carlos de Juan Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *EMNLP*.

Daniel M Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201. Association for Computational Linguistics.

Robert Bossy, Wiktoria Golik, Zorana Ratkovic, Philippe Bessières, and Claire Nédellec. 2013. Bionlp shared task 2013–an overview of the bacteria biotope task. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 161–169.

Xavier Carreras, Lluís Màrquez, and Lluís Padró. 2002. Named entity extraction using adaboost, proceedings of the 6th conference on natural language learning. *August*, 31:1–4.

Raghavendra Chalapathy, Ehsan Zare Borzeshi, and Massimo Piccardi. 2016. An investigation of recurrent neural architectures for drug name recognition. *arXiv preprint arXiv:1609.07585*.

Nancy Chinchor and Patricia Robinson. 1997. Muc-7 named entity task definition. In *Proceedings of the 7th Conference on Message Understanding*, volume 29.

Jason PC Chiu and Eric Nichols. 2015. Named entity recognition with bidirectional lstm-cnns. *arXiv preprint arXiv:1511.08308*.

Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Silviu Cucerzan and David Yarowsky. 2002. Language independent ner using a unified model of internal and contextual evidence. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–4. Association for Computational Linguistics.

Louise Delèger, Robert Bossy, Estelle Chaix, Mouhamadou Ba, Arnaud Ferrè, Philippe Bessieres, and Claire Nèdellec. 2016. Overview of the bacteria biotope task at bionlp shared task 2016. In *Proceedings of the 4th BioNLP Shared Task Workshop*, pages 12–22.

Franck Dernoncourt, Ji Young Lee, and Peter Szolovits. 2017. Neuroner: an easy-to-use program for named-entity recognition based on neural networks. *arXiv preprint arXiv:1705.05487*.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications*, pages 239–250. Springer.

Safaa Eltyeb and Naomie Salim. 2014. Chemical named entities recognition: a review on approaches and applications. *Journal of cheminformatics*, 6(1):17.

Wael Etaiwi, Arafat Awajan, and Dima Suleiman. 2017. Statistical arabic name entity recognition approaches: A survey. *Procedia Computer Science*, 113:57–64.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial intelligence*, 165(1):91–134.

G David Forney. 1973. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2015. Multilingual language processing from bytes. *arXiv preprint arXiv:1512.00103*.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, volume 1.

Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. 2017. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48.

Kristina M Hettne, Rob H Stierum, Martijn J Schuemie, Peter JM Hendriksen, Bob JA Schijvenaars, Erik M van Mulligen, Jos Kleinjans, and Jan A Kors. 2009. A dictionary to identify small molecules and drugs in free text. *Bioinformatics*, 25(22):2983–2991.

Lynette Hirschman, Alexander Yeh, Christian Blaschke, and Alfonso Valencia. 2005. Overview of biocreative: critical assessment of information extraction for biology.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Association for Computational Linguistics.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *AAAI*, pages 2741–2749.

Craig Knox, Vivian Law, Timothy Jewison, Philip Liu, Son Ly, Alex Frolkis, Allison Pon, Kelly Banco, Christine Mak, Vanessa Neveu, et al. 2010. Drugbank 3.0: a comprehensive resource for omics research on drugs. *Nucleic acids research*, 39(suppl_1):D1035–D1041.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(S1):S2.

Onur Kuru, Ozan Arkan Can, and Deniz Yuret. 2016. Charner: Character-level named entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 911–921.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Robert Leaman and Graciela Gonzalez. 2008. Banner: an executable survey of advances in biomedical named entity recognition. In *Biocomputing 2008*, pages 652–663. World Scientific.

Jianbo Lei, Buzhou Tang, Xueqin Lu, Kaihua Gao, Min Jiang, and Hua Xu. 2013. A comprehensive study of named entity recognition in chinese clinical text. *Journal of the American Medical Informatics Association*, 21(5):808–814.

Yaoyong Li, Kalina Bontcheva, and Hamish Cunningham. 2005. Svm based learning system for information extraction. In *Deterministic and statistical methods in machine learning*, pages 319–339. Springer.

Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. 2015. Component-enhanced chinese character embeddings. *arXiv preprint arXiv:1508.06669*.

Nut Limsopatham and Nigel Henry Collier. 2016. Bidirectional lstm for named entity recognition in twitter messages.

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Finding function in form: Compositional character models for open vocabulary word representation. *arXiv preprint arXiv:1508.02096*.

Shengyu Liu, Buzhou Tang, Qingcai Chen, and Xiaolong Wang. 2015. Effects of semantic features on machine learning-based drug name recognition systems: word embeddings vs. manually constructed dictionaries. *Information*, 6(4):848–865.

2015. *Joint Named Entity Recognition and Disambiguation*, September.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Robert Malouf. 2002. Markov models for language-independent named entity recognition, proceedings of the 6th conference on natural language learning. *August*, 31:1–4.

Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Shotaro Misawa, Motoki Taniguchi, Yasuhide Miura, and Tomoko Ohkuma. 2017. Character-based bidirectional lstm-crf with words and characters for japanese named entity recognition. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 97–102.

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

David Nadeau, Peter D Turney, and Stan Matwin. 2006. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 266–277. Springer.

TS Nguyen, LM Nguyen, and XC Tran. 2016. Vietnamese named entity recognition at vlsp 2016 evaluation campaign. In *Proceedings of The Fourth International Workshop on Vietnamese Language and Speech Processing*.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The genia corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the second international conference on Human Language Technology Research*, pages 82–86. Morgan Kaufmann Publishers Inc.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *arXiv preprint arXiv:1404.5367*.

Nita Patil, Ajay S Patil, and BV Pawar. 2016. Survey of named entity recognition systems with respect to indian and foreign languages. *International Journal of Computer Applications*, 134(16).

Thai-Hoang Pham and Phuong Le-Hong. 2017. End-to-end recurrent neural network models for vietnamese named entity recognition: Word-level vs. character-level. *arXiv preprint arXiv:1705.04044*.

Jakub Piskorski, Lidia Pivovarova, Jan Šnajder, Josef Steinberger, Roman Yangarber, et al. 2017. The first cross-lingual challenge on recognition, normalization and matching of named entities in slavic languages. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*. Association for Computational Linguistics.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. *arXiv preprint arXiv:1604.05529*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152.

Lawrence Rabiner and B Juang. 1986. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16.

Dipti Misra Sharma Rajeev Sangal and Anil Kumar Singh, editors. 2008. *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*. Asian Federation of Natural Language Processing, Hyderabad, India, January.

Tim Rocktäschel, Torsten Huber, Michael Weidlich, and Ulf Leser. 2013. Wbi-ner: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs. In *SemEval@ NAACL-HLT*, pages 356–363.

Sujan Kumar Saha, Sudeshna Sarkar, and Pabitra Mitra. 2009. Feature selection techniques for maximum entropy based biomedical named entity recognition. *Journal of Biomedical Informatics*, 42(5):905 – 911. Biomedical Natural Language Processing.

Diana Santos and Nuno Cardoso. 2007. Reconhecimento de entidades mencionadas em português: Documentação e actas do harem, a primeira avaliação conjunta na área.

Cicero Nogueira dos Santos and Victor Guimaraes. 2015. Boosting named entity recognition with neural character embeddings. *arXiv preprint arXiv:1505.05008*.

Robert E Schapire. 2013. Explaining adaboost. In *Empirical inference*, pages 37–52. Springer.

Isabel Segura Bedmar, Paloma Martínez, and María Herrero Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). Association for Computational Linguistics.

Khaled Shaalan. 2014. A survey of arabic named entity recognition and classification. *Computational Linguistics*, 40(2):469–510.

Rahul Sharnagat. 2014. Named entity recognition: A literature survey. *Center For Indian Language Technology*.

Stephanie Strassel, Alexis Mitchell, and Shudong Huang. 2003. Multilingual resources for entity extraction. In *Proceedings of the ACL 2003 workshop on Multilingual and mixed-language named entity recognition-Volume 15*, pages 49–56. Association for Computational Linguistics.

Koichi Takeuchi and Nigel Collier. 2002. Use of support vector machines in extended named entity recognition. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–7. Association for Computational Linguistics.

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 the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Erik F Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: language-independent named entity recognition, proceedings of the 6th conference on natural language learning. *August*, 31:1–4.

Özlem Uzuner, Yuan Luo, and Peter Szolovits. 2007. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Kai Xu, Zhanfan Zhou, Tianyong Hao, and Wenyin Liu. 2017. A bidirectional lstm and conditional random fields approach to medical named entity recognition. In *International Conference on Advanced Intelligent Systems and Informatics*, pages 355–365. Springer.

Vikas Yadav, Rebecca Sharp, and Steven Bethard. 2018. Deep affix features improve neural named entity recognizers. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 167–172.

Shao Yan, Christian Hardmeier, and Joakim Nivre. 2016. Multilingual named entity recognition using hybrid neural networks. In *The Sixth Swedish Language Technology Conference (SLTC)*.

Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *arXiv preprint arXiv:1603.06270*.

Zhilin Yang, Ruslan Salakhutdinov, and William W Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. *arXiv preprint arXiv:1703.06345*.

Rongchao Yin, Quan Wang, Peng Li, Rui Li, and Bin Wang. 2016. Multi-granularity chinese word embedding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 981–986.

Shaodian Zhang and Noémie Elhadad. 2013. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of biomedical informatics*, 46(6):1088–1098.

GuoDong Zhou and Jian Su. 2002. Named entity recognition using an hmm-based chunk tagger. In *proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 473–480. Association for Computational Linguistics.