A Survey in the Named Entity Recognition Focus on Deep Neural Network, Active Learning and Adversarial Learning

姓　　名：朱君鹏

学　　号：52184506004

院　　系：计算机科学技术系

课　　程：文本挖掘

指导教师：兰曼

Jan 2, 2019

Overview

1 Introduction

Information extraction (IE) is the task of automatically extracting structured information from unstructured or semi-structured text. In other words information extraction can be considered as a limited form of full natural language understanding, where the information we are looking for are known beforehand.

知识抽取的任务是：从非结构化或半结构化文本中抽取结构化信息的过程。换句话说，信息抽取能够被看作自然语言理解的一种限定形式，我们想要寻找的信息在执行知识抽取之前就已经知道。

IE is one of the critical task in text mining and widely studied in different research communities such as information retrieval, natural language processing and Web mining. Similarly, It has vast application in domains such as biomedical text mining and business intelligence. See for some of the applications of information extraction. Information extraction includes two fundamental tasks, namely, name entity recognition and relation extraction. The state of the art in both tasks are statistical learning methods. In the following we briefly explain two information extraction tasks.

在文本挖掘领域，IE 是一个重要的任务，在信息检索、自然语言处理、web 挖掘等领域也被广泛地研究。相类似，IE 也有广泛的应用领域，如生物医学文本挖掘、商业智能。IE 包含两个重要的任务，分别是：命名实体识别（简称为 NERC 或者 NER）和关系抽取。在这两个任务中，当前比较优秀的算法都是基于统计的学习方法。下面我们将详细通过文献汇总近几年最新的研究成果。

【1】 Allahyari M , Pouriyeh S , Assefi M , et al. A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques[J]. 2017.

To identify articles for this survey, I searched Google, Google Scholar, Semantic Scholar and Web of Science. The terms of querying included named entity recognition, neural architectures for named entity recognition, neural network based named entity recognition models, deep learning models for named entity recognition, deep active learning for the named entity recognition and deep adversarial learning for the named entity recognition. I sorted the papers returned from each query by citation count and read at least the top five, 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. I included an article presenting a neural architecture only if it was the first article to introduce the

architecture; otherwise, I traced citations back until i found the original source of the architecture. I followed the same approach for feature-engineering NER systems. I also included articles that implemented these systems for different languages or domain. In total, 44 articles were reviewed and were selected for the survey.

2 The Proposed Features of NER

The use of an expressive and varied set of features turns out to be just as important as the choice of machine learning algorithms。

表达式和不同特征的使用被证明和机器学习算法一样重要。

Features are descriptors or characteristic attributes of words designed for algorithmic consumption. An example of a feature is a Boolean variable with the value true if a word is capitalized and false otherwise. Feature vector representation is an abstraction over text where typically each word is represented by one or many Boolean, numeric and nominal values. For example, a hypothetical NERC system may represent each word of a text with 3 attributes:

1) a Boolean attribute with the value true if the word is capitalized and false otherwise;
2) a numeric attribute corresponding to the length, in characters, of the word;
3) a nominal attribute corresponding to the lowercased version of the word.

在算法设计中，特征是一个描述符或者单词的典型属性，例如，当一个单词首字母大写时此时其特征的 Boolean 值为 True，反之则说明其特征值为 false。特征向量是一个文本的抽象表示，一般情况下，每个单词都可以通过一个 Boolean 值、数字、名词表示。例如，假设一个 NERC 系统中的每个次都包含以下三个属性：

1) 如果单词的首字母大写，则将其表示为 True，否则为 False；
2）用一个数字表示单词的长度
3) 用该单词的全部字母小写形式来表示这个单词。

In this scenario, the sentence “The president of Apple eats an apple.”, excluding the punctuation, would be represented by the following feature vectors:

```
<true, 3, “the”>, <false, 9, “president”>, <false, 2, “of”>, <true,
5, “apple”>, <false, 4, “eats”>, <false, 2, “an”>, <false, 5, “apple”>
```

在上述假设下，句子 The president of Apple eats an apple.除了标点符号之外，它将表示成下面的特征向量形式:

```
<true, 3, “the”>, <false, 9, “president”>, <false, 2, “of”>, <true,
5, “apple”>, <false, 4, “eats”>, <false, 2, “an”>, <false, 5, “apple”>
```

Usually, the NERC problem is resolved by applying a rule system over the features. For instance, a system might have two rules, a recognition rule: “capitalized words are candidate entities” and a

classification rule: “the type of candidate entities of length greater than 3 words is organization”. These rules work well for the exemplar sentence above. However, real systems tend to be much more complex and their rules are often created by automatic learning techniques.

通常情况下，NERC 系统通过一系列规则所决定。例如：一个系统也许有两个规则，其一是一个识别规则：大写的单词是候选实体，其二是一个分类规则：如果这种类型的单词长度大于 3，那么它被认为是一个组织的名字。这些规则在上面的例句中表现极佳。然而，真实的系统往往具有非常复杂的规则，并且这些规则通常情况下通过自动学习技术被创建。

In this section, we present the features most often used for the recognition and classification of named entities. We organize them along three different axes: Word-level features, List lookup features and Document and corpus features.

在这部分，我们将给出一些在识别和分类命名实体时经常被使用的规则。本文将其组织成为三种不同的类型：Word-level、List Lookup 特征和文档及语料库特征。

2.1 Word-Level Feature

Word-level features are related to the character makeup of words. They specifically describe word case, punctuation, numerical value and special characters. Table 1 lists subcategories of word-level features.

word-level 特征经常和组成单词的字符相关。它们专门描述单词的大小写（word case）、标点符号、数字特征和特殊字符。Table 1 给出了 word-level 级别特征的资分类情况。

Table 1: Word-level features

Features	Examples
Case	- Starts with a capital letter - Word is all uppercased - The word is mixed case (e.g., ProSys, eBay)
Punctuation	- Ends with period, has internal period (e.g., St., I.B.M.) - Internal apostrophe, hyphen or ampersand (e.g., O’Connor)
Digit	- Digit pattern (*see section 3.1.1*) - Cardinal and Ordinal - Roman number - Word with digits (e.g., W3C, 3M)
Character	- Possessive mark, first person pronoun - Greek letters
Morphology	- Prefix, suffix, singular version, stem - Common ending (*see section 3.1.2*)
Part-of-speech	- proper name, verb, noun, foreign word
Function	- Alpha, non-alpha, n-gram (*see section 3.1.3*) - lowercase, uppercase version - pattern, summarized pattern (*see section 3.1.4*) - token length, phrase length

2.2 List LookupFeature

Lists are the privileged features in NERC. The terms "gazetteer", "lexicon" and "dictionary" are often used interchangeably with the term "list". List inclusion is a way to express the relation "is a" (e.g., Paris is a city). It may appear obvious that if a word (Paris) is an element of a list of cities, then the probability of this word to be city, in a given text, is high. However, because of word polysemy, the probability is almost never 1 (e.g., the probability of "Fast" to represent a company is low because of the common adjective "fast" that is much more frequent).

在 NERC 系统中， Lists 是主要的特征。术语"Dictionary"、"Lexicon"经常与 List 互换，也就是 Lists Lookup 是指词典查询。list 的包含操作经常被用来表示"Is a"关系（例如 Pairs is a city）。如果单词（例如 Pairs）是城市列表中的一个元素，那么这么单词是城市的概率将会极高。然而，由于存在一词多义现象，导致这个概率几乎不可能达到 1（例如 Fast 表示一个公司的概率是极低的，因为通常情况下 fast 用作形容词会更加的常见）。

Table 2: List lookup features.

Features	Examples
General list	- General dictionary (see section 3.2.1) - Stop words (function words) - Capitalized nouns (e.g., January, Monday) - Common abbreviations
List of entities	- Organization, government, airline, educational - First name, last name, celebrity - Astral body, continent, country, state, city
List of entity cues	- Typical words in organization (see 3.2.2) - Person title, name prefix, post-nominal letters - Location typical word, cardinal point

2.3 Document and Corpus Feature

Document features are defined over both document content and document structure. Large collections of documents (corpora) are also excellent sources of features. We list in this section features that go beyond the single word and multi-word expression and include meta-information about documents and corpus statistics.

文档特征包含两个方面，分别是：文档内容特征和文档结构特征。大量的文档（语料库）也是极好的特征源。下面表 3 给出了一些例子。

Table 3: Features from documents.

Features	Examples
Multiple occurrences	- Other entities in the context - Uppercased and lowercased occurrences (see 3.3.1) - Anaphora, coreference (see 3.3.2)
Local syntax	- Enumeration, apposition - Position in sentence, in paragraph, and in document
Meta information	- Uri, Email header, XML section, (see section 3.3.3) - Bulleted/numbered lists, tables, figures
Corpus frequency	- Word and phrase frequency - Co-occurrences - Multiword unit permanency (see 3.3.4)

3 NER Evaluation Paradigms

Thorough evaluation of NERC systems is essential to their progress. Many techniques were proposed to rank systems based on their capability to annotate a text like an expert linguist. In the following section, we take a look at three main scoring techniques used for MUC, IREX, CONLL and ACE conferences. But first, let's summarize the task from the point of view of evaluation.

为了 NERC 系统的发展，彻底的评估 NERC 系统非常重要。为了排名 NERC 系统，这

些系统都能够像语言学家一样具备标注文本的能力，在此基础之上许多技术被提出。下一节，我们给出三个主要的评分技术，这些技术分别被使用在 MUC、IREX、CONLL 和 ACE 会议。首先，我们从评估的观点来总结 NERC 任务。

In NERC, systems are usually evaluated based on how their output compares with the output of human linguists. For instance, here's an annotated text marked up according to the MUC guidelines. Let's call it the solution.

在 NERC 系统中，通常的评估方法会对比机器与人类语言学家的输出结果。例如，这里给出一个根据 MUC 指南标注过的文本，让我们来看看它的解决方案。

```
Unlike <ENAMEX TYPE="PERSON">Robert</ENAMEX>, <ENAMEX TYPE="PERSON">John Briggs Jr</ENAMEX> contacted <ENAMEX TYPE="ORGANIZATION">Wonderful Stockbrockers Inc</ENAMEX> in <ENAMEX TYPE="LOCATION">New York</ENAMEX> and instructed them to sell all his shares in <ENAMEX TYPE="ORGANIZATION">Acme</ENAMEX>.
```

Let's now hypothesize a system producing the following output:

下面我们假设系统给出了如下的输出结果：

```
<ENAMEX TYPE="LOCATION">Unlike</ENAMEX> Robert, <ENAMEX TYPE="ORGANIZATION">John Briggs Jr</ENAMEX> contacted Wonderful <ENAMEX TYPE="ORGANIZATION">Stockbrockers</ENAMEX> Inc <ENAMEX TYPE="PERSON">in New York</ENAMEX> and instructed them to sell all his shares in <ENAMEX TYPE="ORGANIZATION">Acme</ENAMEX>.
```

The system produced five different errors3, explained in Table 4. In this example, the system gives one correct answer: (<Organization> Acme </Organization>). Ultimately, the question is "What score should we give to this system?" In the following sections, we survey how the question was answered in various evaluation forums.

对比两个结果，我们可以知道该系统产生了 5 个不同的错误，下表给出了详细的说明。在这个例子中，系统给出了一个正确的答案 (<Organization> Acme </Organization>). 最终，问题是：我们给这个系统什么样的评分？下面部分，我们将给出在不同的评估方法里，如何回答这个问题。

Correct solution	System output	Error
Unlike	<ENAMEX TYPE="LOCATION"> Unlike </ENAMEX>	The system hypothesized an entity where there is none.
<ENAMEX TYPE="PERSON"> Robert </ENAMEX>	Robert	An entity was completely missed by the system.
<ENAMEX TYPE="PERSON"> John Briggs Jr </ENAMEX>	<ENAMEX TYPE="ORGANIZATION"> John Briggs Jr </ENAMEX>	The system noticed an entity but gave it the wrong label.
<ENAMEX TYPE="ORGANIZATION"> Wonderful Stockbrockers Inc </ENAMEX>	<ENAMEX TYPE="ORGANIZATION"> Stockbrockers </ENAMEX>	A system noticed there is an entity but got its boundaries wrong.
<ENAMEX TYPE="LOCATION"> New York </ENAMEX>	<ENAMEX TYPE="PERSON"> in New York </ENAMEX>	The system gave the wrong label to the entity and got its boundary wrong.

3.1 MUC Evaluations

In MUC events (R. Grishman & Sundheim 1996, N. Chinchor 1999), a system is scored on two axes: its ability to find the correct type (TYPE) and its ability to find exact text (TEXT). A correct TYPE is credited if an entity is assigned the correct type, regardless of boundaries as long as there is an overlap. A correct TEXT is credited if entity boundaries are correct, regardless of the type. For both TYPE and TEXT, three measures are kept: the number of correct answers (COR), the number of actual system guesses (ACT) and the number of possible entities in the solution (POS).

在 MUC 1996 年和 1999 年的论文中，系统在两个维度上评分：一是寻找正确类型的能力；二是寻找精确文本的能力。何谓正确的类型？何谓精确的文本？正确的类型是指：一个实体被指定为正确的类型，不管边界是什么，只要有重叠就行。精确的文本是指：实体边界是正确的，但是忽略类型。对于 TYPE 和 TEXT，有三个度量标准：**正确答案的数目（COR）**，**实际系统猜测到的实体的数目（ACT）**和**在问题中可能存在的实体的数目（POS）**。

The final MUC score is the micro-averaged f-measure (MAF), which is the harmonic mean of precision and recall calculated over all entity slots on both axes. A micro-averaged measure is performed on all entity types without distinction (errors and successes for all entity types are summed together). The harmonic mean of two numbers is never higher than the geometrical mean. It also tends toward the least number, minimizing the impact of large outliers and maximizing the impact of small ones. The F-measure therefore tends to privilege balanced systems.

最终的 MUC 度量标准是 MAF，它是一个在所有实体的两个维度上精确度和召回率调和平均值。MAF 度量被用于所有实体类型上。召回率和精确度的调和平均值从来都比起几个平均值小。它也趋向于最小的数，最小化了离群点的影响，同时也最大化了小值的影响。因此通常情况下 MAF 度量倾向于在平衡占主导地位的系统中。

In MUC, precision is calculated as COR / ACT and the recall is COR / POS. For the previous example, COR = 4 (2 TYPE + 2 TEXT), ACT = 10 (5 TYPE + 5 TEXT) and POS = 10 (5 TYPE + 5 TEXT). The precision is therefore 40%, the recall is 40% and the MAF is 40%.

在 MUC 中，精确度被计算通过 COR/ACT，召回率是通过计算 COR/POS 计算得到。对

于前面例子,精确度是 40%，召回率和 MAF 都是 40%。

$$\frac{1}{\frac{1}{2} * \frac{1}{40\%} * \frac{1}{40\%}} = 40\%$$

This measure has the advantage of taking into account all possible types of errors of Table 4. It also gives partial credit for errors occurring on one axis only. Since there are two evaluation axes, each complete success is worth two points. The worst errors cost this two points (missing both TYPE and TEXT) while other errors cost only one point.

这种度量方法充分地考虑了所有可能出现的错误。它还为只在一个轴上发生的错误提供部分信用（比如可能实体在预测的过程中，只是实体的类型预测错误，但是确实它是一个实体，比如上述例子中的 John Briggs Jr，确实是一个实体，但是实体的类型预测错误）。因为存在两个评估维度，每一个完全成功的实体预测都会共享两个点。最严重的错误是两个维度的预测全部不正确，其它情况都能保证一个维度正确。

3.2 Exact-match Evaluations

IREX and CONLL share a simple scoring protocol. We can call it “exact-match evaluation”. Systems are compared based on the micro-averaged f-measure (MAF) with the precision being the percentage of named entities found by the system that are correct and the recall being the percentage of named entities present in the solution that are found by the system. A named entity is correct only if it is an exact match of the corresponding entity in the solution.

IREX 和 CONLL 共享一个简单的评分标准，我们被该评分标准称为“精确匹配评估”。根据 MAF 对系统进行比较，其精度为系统找到正确的命名实体的百分比，召回率为系统找到的解决方案中出现的命名实体的百分比。一个命名实体正确当且仅当它精确的匹配问题中对应的实体。

For the previous example, there are 5 true entities, 5 system guesses and only one guess that exactly matches the solution. The precision is therefore 20%, the recall is 20% and the MAF is 20%.

对于之前的例子，有 5 个真实的实体，系统检测到存在 5 个命名实体，但是只有一个命名实体是正确的，因此精确度为 20%，召回率为 20%，并且 MAF 也为 20%（MAF 计算方法同上）。

For some application, the constraint of exact match is unnecessarily stringent. For instance, in some bioinformatics work, the goal is to determine whether or not a particular sentence mentions a specific gene and its function. Exact NE boundaries are not required: all is needed is to determine if the sentence does refer to the gene (R. Tzong-Han Tsai et al. 2006).

对于一些应用，不必太严格限制精确匹配。例如：在生物信息学中，主要的目标是确定是/否在一个特殊的句子中提到了一个特定的基因和它的功能。精确的命名实体边界不被需要，因为：如果一个句子包含一个基因，那么整个句子就会被选出。

【2】 Tsai T H , Wu S H , Chou W C , et al. Various criteria in the evaluation of biomedical named entity recognition[J]. Bmc Bioinformatics, 2006, 7(1):92-0.

3.3 ACE Evaluations

ACE has a complex evaluation procedure. It includes mechanisms for dealing various evaluation issues (partial match, wrong type, etc.). The ACE task definition is also more elaborated than previous tasks at the level of named entity "subtypes", "class" as well as entity mentions (coreferences), and more, but these supplemental elements will be ignored here.

ACE 拥有一个复杂的评估程序，它包含了评估不同问题的机制，这些问题包括：部分匹配、错误，错误类型等。ACE 在任务定义上比之前的情况都更为详细。本文不再详述，具体参见文献。

3.4 F1 Score Evaluations

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; Deleger 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).

3.5 NER Datasets

- Shared task: https://www-nlpir.nist.gov/related_projects/muc/
- Shared task: https://www.clips.uantwerpen.be/conll2002/ner/
- Shared task: https://www.clips.uantwerpen.be/conll2003/ner/
- Shared task: http://bsnlp.cs.helsinki.fi/shared_task.html
- hared task: https://www.i2b2.org/NLP/Relations/
- Shared task: https://www.cs.york.ac.uk/semeval-2013/task9/index.html
- Similar datasets can be found here: http://www.biocreative.org
- Shared task: http://2016.bionlp-st.org/tasks/bb2

4 NER Systems for the Different Languages and Domains

4.1 The NER Research of Different Languages

1. The first paper introduces the Named Entity Recognition, and review the evaluation of precision and recall using the English Language.
【3】Grishman R, Sundheim B. Message understanding conference-6: A brief history[C]//COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics. 1996, 1.

2. Language-independent named entity recognition for using the English and Germen Languages.
【4】Tjong Kim Sang E F, De Meulder F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition[C]//Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4\. Association for Computational Linguistics, 2003: 142-147.

The focus research topic of above papers is the PER (person), LOC (location), ORG (organization) and MISC (miscellaneous including all other types of entities).

3. The first paper includes Indian Language.
【5】Begum R, Husain S, Dhwaj A, et al. Dependency annotation scheme for Indian languages[C]//Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II. 2008.

4. The first paper includes Chinese Lanauage.
【6】Che W, Li Z, Liu T. Ltp: A chinese language technology platform[C]//Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations. Association for Computational Linguistics, 2010: 13-16. (COLING'10)

5. The first paper includes Arabic language.
【7】Shaalan K, Oudah M. A hybrid approach to Arabic named entity recognition[J]. Journal of Information Science, 2014, 40(1): 67-87.

6. Germen language
【8】Benikova D, Biemann C, Kisselew M, et al. Germeval 2014 named entity recognition shared task: companion paper[J]. 2014.

7. The first Cross-Lingual Challenge on Recognition, Normalization and Matching of Named Entities in Slavic Languages
【9】Piskorski J, Pivovarova L, Šnajder J, et al. The First Cross-Lingual Challenge on Recognition, Normalization and Matching of Named Entities in Slavic Languages[C]//Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing. Association for Computational Linguistics, 2017.

4.2 The NER Research of Different Domains

1. Bioinfomatic Domain
【10】Song Y, Kim E, Lee G G, et al. POSBIOTM-NER in the shared task of BioNLP/NLPBA 2004[C]//Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications. Association for Computational Linguistics, 2004: 100-103.

2. Clinical Medical Informatics Domain
【11】Uzuner Ö, Luo Y, Szolovits P. Evaluating the state-of-the-art in automatic de-identification[J].

Journal of the American Medical Informatics Association, 2007, 14(5): 550-563.

3. The Improvement of the Previous Paper 2

【12】Uzuner Ö, South B R, Shen S, et al. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text[J]. Journal of the American Medical Informatics Association, 2011, 18(5): 552-556.

4. Medicine Domain

【13】Segura-Bedmar I, Martínez P, Zazo M H. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013)[C]//Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). 2013, 2: 341-350.

5. Chemicals Domain

【14】Krallinger M, Rabal O, Leitner F, et al. The CHEMDNER corpus of chemicals and drugs and its annotation principles[J]. Journal of cheminformatics, 2015, 7(1): S2.

6. Social Media（person, company, facility, band, sport steam, movie, TV show, etc.）

【15】Baldwin T, de Marneffe M C, Han B, et al. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition[C]//Proceedings of the Workshop on Noisy User-generated Text. 2015: 126-135.

5 The Traditional Systems for the Named Entity Recognition and Classification

Identifying references to these entities in text was recognized as one of the important sub-tasks of IE and was called “Named Entity Recognition and Classification (NERC)”

IE 中一个重要的子任务就是标识实体在文本中的引用，这个过程被称为命名实体识别和分类.

【16】Nadeau D , Sekine S . A survey of named entity recognition and classification[J]. Lingvisticae Investigationes, 2007, 30(1):3-26.

While early systems were making use of handcrafted rule-based algorithms, modern systems most often resort to machine learning techniques.

早期的用于处理 NERC 任务的系统/算法主要是充分利用了纯手工的基于规则的算法，而现代系统/算法大都借助机器学习技术。

We survey these techniques as well as other critical aspects of NERC such as features and evaluation methods. It was indeed concluded in a recent conference that the choice of features is at least as important as the choice of technique for obtaining a good NERC system (E.Tjong Kim Sang & De Meulder 2003).

我们也汇总了 NERC 任务中一些其他重要的方面，比如特征和评估方法。在最近（2003

年）会议中指出一个重要的结论：在得到一个好的 NERC 系统中，特征的选择和技术的选择一样重要。

【17】Sang E F T K , De Meulder F . Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition[J]. 2003.

A named entity is a sequence of words that identifes some real world entity, e.g. “Google Inc”, “United States”, “Barack Obama”. The task of named entity recognition is to locate and classify named entities in free text into predefned categories such as person, organization, location, etc. NER can not be completely done simply by doing string matching against a dictionary, because a) dictionaries are usually incomplete and do not contain all forms of named entities of a given entity type. b) Named entities are frequently dependent on context, for example “big apple” can be the fruit, or the nickname of New York. Named entity recognition is a preprocessing step in the relation extraction task and also has other applications such as in question answering. Most of the named entity recognition techniques are statistical learning methods such as hidden Markov models [13], maximum entropy models, support vector machines and conditional random felds.

每个命名实体都是一个单词序列，每个这样的序列都能表示现实世界中的一个实体，例如：“Google Inc”（谷歌公司），“United States”（美国），“Barack Obama” （美国前总统奥巴马）。命名实体识别的任务是：定位并将任意文本中的命名实体分类到预先定义的类型中，比如人、组织、位置等等。命名实体识别不能仅仅通过字符串的简单匹配实现，这主要基于下面的两点原因：（a）字典通常不完整，这通常会导致不能包含一个给定实体类型的所有命名实体形式。（b）命名实体自身依赖于上下环境，例如“big apple”可以表示水果，也可以表示美国纽约的昵称。在关系抽取任务或其它诸如 Q&A 系统中，命名实体识别是其中一个重要的预处理步骤，大部分命名实体识别技术都属于统计学习方法范畴，比如隐马尔科夫模型、最大熵模型、SVM 支持向量机模型和条件随机菲尔德模型等。

【18】sun Zhen, Wang Huilin. Overview on the Advance of the Research on Named Entity Recognition. New Technology of Library and Information Service, 2010, 26(6): 42-47

【19】Allahyari M , Pouriyeh S , Assefi M , et al. A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques[J]. 2017.

While early studies were mostly based on handcrafted rules, most recent ones use supervised machine learning (SL) as a way to automatically induce rule-based systems or sequence labeling algorithms starting from a collection of training examples.

在 2006 年之前的研究主要分为两个派别： 分别是基于 handcrafted 规则的方法，另外一大类型是基于学习算法（包括监督学习、半监督学习、无监督学习）。下面简要介绍学习算法在 NERC 中的应用，由于基于 handcrafted 规则的方法比较陈旧，本文中将不再赘述。

5.1 The NER Systems based on the Knowledge and Feature Engineering

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 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.

基于知识的方法不要求带有标注的训练数据，因为他们仅仅依赖于词典或者领域特定的知识。当词典十分详细时，这种算法的性能将会十分好，反之，则会十分差。因为有字典的存在，因此这些算法的精确度通常情况下会非常高，召回率通常情况下比较低，**这主要是因为领域或者特定的语言规则以及不完整的字典导致。另外一个显著的缺点是：基于知识的 NER 系统需要领域专家去构建并维护相应的知识源。**

【20】Knox C, Law V, Jewison T, et al. DrugBank 3.0: a comprehensive resource for 'omics' research on drugs[J]. Nucleic acids research, 2010, 39(suppl_1): D1035-D1041. **引用：1551**

5.2 The NER Systems based on the Machine Learning Algorithms

5.2 1 Supervised Learning Algorithms

监督学习算法的缺点是：需要大量标注的数据。出自： Supervised learning is used, a prerequisite is the availability of a large collection of annotated data.

The current dominant technique for addressing the NERC problem is supervised learning. SL techniques include Hidden Markov Models (HMM) (D. Bikel et al. 1997), Decision Trees (S. Sekine 1998), Maximum Entropy Models (ME) (A. Borthwick 1998), Support Vector Machines (SVM) (M. Asahara & Matsumoto 2003), and Conditional Random Fields (CRF) (A. McCallum & Li 2003). These are all variants of the SL approach that typically consist of a system that reads a large annotated corpus, memorizes lists of entities, and creates disambiguation rules based on discriminative features.

目前（2007 年之前），解决 NERC 问题使用的主流技术是监督学习算法。其中包括：隐马尔科夫模型（HMM）、决策树模型（DT）、最大熵模型（ME）、支持向量机模型（SVM）和条件随机菲尔德模型（CRF）。它们都是监督学习的不同变种，它们的典型的思路是：读取一个大量的带标注的语料库、存储实体列表并且基于不同的特征创建一些消歧规则。

【21】Bikel D M , Schwartz R , Weischedel R M . An Algorithm that Learns What's in a Name[J]. Machine Learning, 1999, 34(1-3):211-231.

【22】A. Borthwick, J. Sterling, E. Agichtein, R. Grishman. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In Proceedings of the sixth workshop on very large corpora, 1998.

【23】Asahara M , Matsumoto Y . Japanese Named Entity extraction with redundant morphological analysis[C]// Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology. Association for Computational Linguistics, 2003.

【24】Mccallum A , Li W . Early results for Named Entity Recognition with Conditional Random

Fields, Feature Induction and Web-Enhanced Lexicons.[C]// Conference on Natural Language Learning at Hlt-naacl. Association for Computational Linguistics, 2003.

A baseline SL method that is often proposed consists of tagging words of a test corpus when they are annotated as entities in the training corpus. The performance of the baseline system depends on the vocabulary transfer, which is the proportion of words, without repetitions, appearing in both training and testing corpus.Vocabulary transfer is a good indicator of the recall (number of entities identified over the total number of entities) of the baseline system but is a pessimistic measure since some entities are frequently repeated in documents. A. Mikheev et al. (1999) precisely calculated the recall of the baseline system on the MUC-7 corpus. They report a recall of 76% for locations, 49% for organizations and 26% for persons with precision ranging from 70% to 90%. Whitelaw and Patrick (2003) report consistent results on MUC-7 for the aggregated enamex class. For the three enamex types together, the precision of recognition is 76% and the recall is 48%.

被提出的监督学习算法由测试语料库的标注词组成，在训练语料库时，他们被标准作为实体。监督学习算法的性能依赖于语料库训练和测试中出现的词汇迁移（vocabulary transfer），即词汇没有重复。vocabulary transfer 是召回率（召回率=被识别的实体的总数/实体总数，其中可能存在某些实体未被识别）的良指示器，但其实它是一个比较保守的度量标准，因为在文档中一些实体可能会存在大量重复。

5.2.2 Semi-Supervised Learning Algorithms

The term “semi-supervised” (or “weakly supervised”) is relatively recent. The main technique for SSL is called “bootstrapping” and involves a small degree of supervision, such as a set of seeds, for starting the learning process. For example, a system aimed at “disease names” might ask the user to provide a small number of example names. Then the system searches for sentences that contain these names and tries to identify some contextual clues common to the five examples. Then, the system tries to find other instances of disease names that appear in similar contexts. The learning process is then reapplied to the newly found examples, so as to discover new relevant contexts. By repeating this process, a large number of disease names and a large number of contexts will eventually be gathered. Recent experiments in semi-supervised NERC (Nadeau et al. 2006) report performances that rival baseline supervised approaches. Here are some examples of SSL approaches.

半监督学习（也称为弱监督学习）是近来出现较新的术语。半监督学习的主要技术叫做“bootstrapping”，并且半监督学习技术会涉及到一小部分的监督过程，**比如需要一小部分的种子点用于整个学习过程的开始**。例如，某个基于半监督学习算法的系统主要是寻找疾病名称，此时需要用户提供少量的疾病名称实例。接着该系统会搜索包含实例中给出的那些名字的文本，并且会尝试去识别给出实例中的上下文提示。接着该系统会尝试去寻找其他出现在相似上下文中的疾病名称。接着，对于新找到的疾病名称以及上下文，上述学习过程会被再次使用，一遍能够发现新的相关的上下文。通过重复上述过程，大量的疾病名称和大量与疾病名称相关的上上下文就最终被聚集到一起。

5.2.3 Unsupervised Learning Algorithms

The typical approach in unsupervised learning is clustering. For example, one can try to gather named entities from clustered groups based on the similarity of context. There are other unsupervised methods too. Basically, the techniques rely on lexical resources (e.g., WordNet), on lexical patterns and on statistics computed on a large unannotated corpus.

典型的无监督学习技术是聚类。例如，一个可行的办法是根据上下文中的相似性度量标准可以将命名实体聚集成簇。当然还有其他的非无监督学习算法。这些算法基本上都依赖词典，它们都在大规模无标准的语料库中进行统计计算和词典模式匹配。文献【10】是无监督学习的第一篇文章，该文完全不需要监督学习和人类的干预就能完成。

【16】Nadeau, David; Turney, P.; Matwin, S. 2006. Unsupervised Named Entity Recognition: Generating Gazetteers and Resolving Ambiguity. In Proc. Canadian Conference on Artificial Intelligence. **引用次数： 193**

【25】Zhang S, Elhadad N. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts[J]. Journal of biomedical informatics, 2013, 46(6): 1088-1098.

6 The NER Systems based on the Deep Neural Network, Deep Active Learning and Adversarial Learning

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.

命名实体识别是 NLP 领域中重要的研究主题，其中包括 Q&A、信息检索、关系抽取领域。NER 系统已经研究并且发展了几十年，但是使用深度神经网络技术的 NER 系统研究在最近几年初现端倪。本文将给出基于深度神经网络模型的 NER 系统的架构，并且对比了早期基于特征工程和其它监督、半监督学习算法和基于深度神经网络算法之间的差异。我们的结果强调了深度神经网络在其中的改进机制，并且给出了如果在早期的基于特征工程的 NER 系统中加入深度神经网络以便进一步改进性能。

【26】Vikas Yadav. Steven Bethard. A Survey on Recent Advances in Named Entity Recognition from Deep Learning models[C]//Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018) , pages 2145－2158.

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.

命名实体识别是一个重要的任务，目的是标记文本中的实体，如人、位置、组织、药物、时间、蛋白质等。NER 系统经常被作为 Q&A 系统、信息检索嘻嘻、主题模型中的第一步。因此，强调进来在命名实体技术上的进展，尤其是在 Neural Network 上的体系结构。

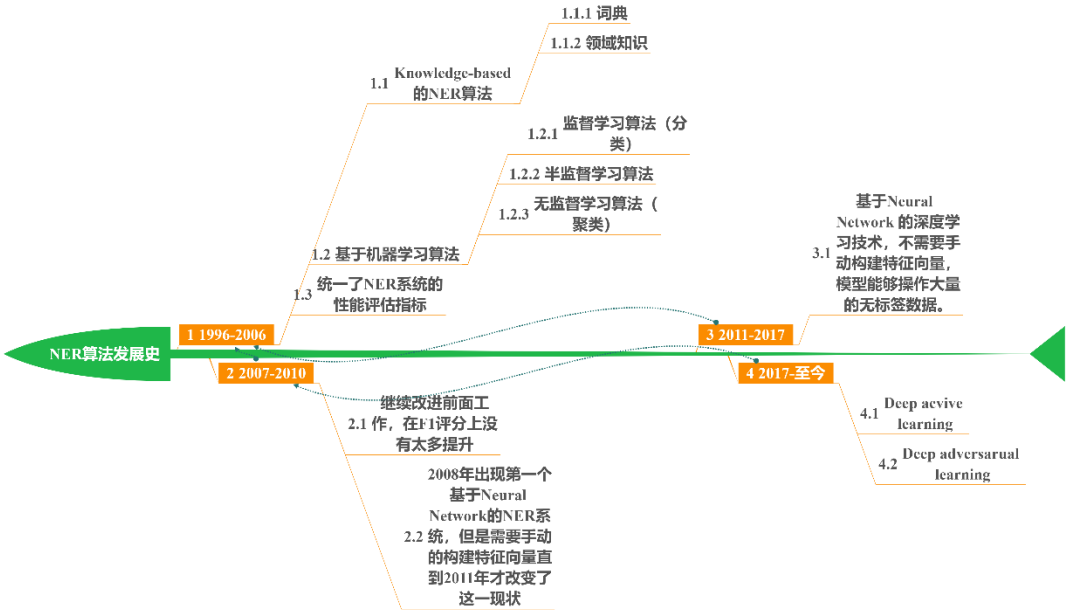
The first NER task was organized by Grishman and Sundheim (1996) in the Sixth Message Understanding Conference. Since then, there have been numerous NER tasks (Tjong Kim Sang and De Meulder, 2003; Tjong Kim Sang, 2002; Piskorski et al., 2017; Segura Bedmar et al., 2013; Bossy et al., 2013; Uzuner et 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 achine learning (Nadeau and Sekine, 2007). Starting with Collobert et al. (2011), neural network NER ystems with minimal feature engineering have become popular. Such models are appealing because hey 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 roposed, mostly based on some orm of recurrent neural networks (RNN) over characters, sub-words and/or word embeddings.

第一个 NER 任务在 1996 年 MUC 会议上被提出。从那之后，涌现了大量的命名实体识别算法。**（1996-2006）的命名实体识别系统主要是基于手动的规则、词典、形态学特征和本体、特征工程等**。紧随其后**（2007-2011）**出涌现了大量**基于机器学习的算法**。经过了一段时间的沉寂之后，在 2011 年 Collobert 等人提出了**基于 Neural Network 的 NER 系统**，使得基于 **Neural Network 的 NER** 系统变得**极其流行**。为什么这样的模型会出现？因为这些模型通常都不需要特殊的领域知识，比如像字典和本体等，也就是说他们是领域独立的。不同的神经网络架构被提出，但大部分都是基于 RNN（Recurrent Neural Network）。

In recently, it is focus on the NER which is based on the deep active learning and adversarial learning. This is mainly due to the technology of active learning and GAN. (**It is my opinion**.)

近来由于深度主动学习技术、对抗学习等技术的出现，使得 NER 重新变得热门。

As a concluded, the evolution history of NER systems may be classified 4 steps, including in the systems based on the orthographic features, the systems based on the feature-engineers, the systems based on the machine learning algorithms, the systems based on the deep neural network and deep active learning and deep adversarial learning. The following picture gives the steps for the changes of NER system. (**It's my opinion**.)



【27】Tjong Kim Sang E F, De Meulder F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition[C]//Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4\. Association for Computational Linguistics, 2003: 142-147.

【28】Piskorski J, Pivovarova L, Šnajder J, et al. The First Cross-Lingual Challenge on Recognition, Normalization and Matching of Named Entities in Slavic Languages[C]//Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing. Association for Computational Linguistics, 2017.

【29】Nadeau D, Sekine S. A survey of named entity recognition and classification[J]. Lingvisticae Investigationes, 2007, 30(1): 3-26.

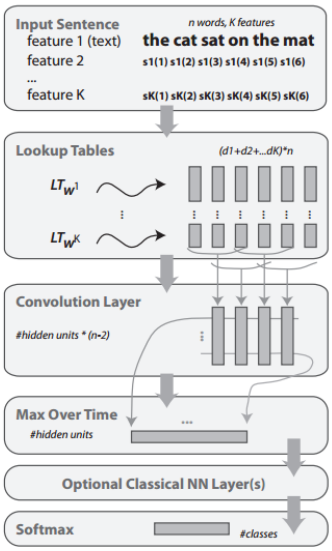
【30】Collobert R, Weston J, Bottou L, et al. Natural language processing (almost) from scratch[J]. Journal of Machine Learning Research, 2011, 12(Aug): 2493-2537. **3855 次引用**

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.

本文将给出一个综合性的关于命名实体识别最近的研究进展。我们将描述基于知识和特征工程的 NER 系统，这些系统将领域知识等特征和监督学习算法、非监督学习算法、半监督学习算法结合在一起。我们将对比那些系统和基于 NN 架构的系统之间的差异，并且比较不同的 NN 模型。在表 1 和表 2 给出了 NN 模型如何和之前的技术相结合来改进 NER 系统

的性能。

6.1 The First NER System based on the Deep Neural Network and Feature Engineering



1）The first layer extracts features for each word. The first layer has to map words into real-valued vectors.

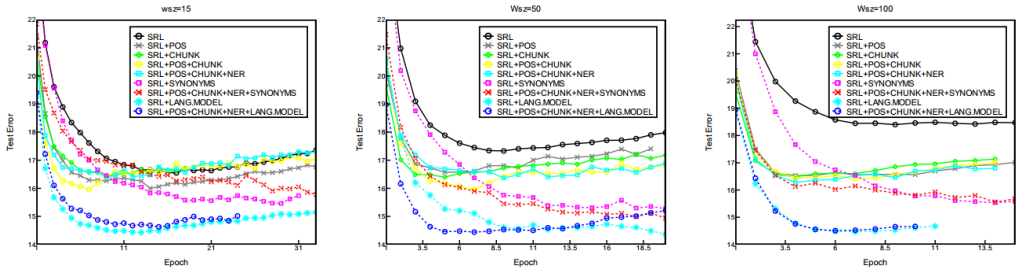
2）The second layer extracts features from the sentence treating it as a sequence with local and global structure (i.e., it is not treated like a bag of words).

3）The following layers are classical NN layers. A general deep NN architecture for NLP. Given an input sentence, the NN outputs **class probabilities** for one chosen word.

Datasets NER labeled data was obtained by running the Stanford Named Entity Recognizer over the PropBank dataset（https://propbank.github.io/） version 1 (about 1 million words). It uses the dictionary of the 30, 000 most common words from Wikipedia, converted to lower case. Other words were considered as unknown and mapped to a special word.

Experimental Results

	wsz=15	wsz=50	wsz=100
SRL	16.54	17.33	18.40
SRL + POS	15.99	16.57	16.53
SRL + Chunking	16.42	16.39	16.48
SRL + NER	16.67	17.29	17.21
SRL + Synonyms	15.46	15.17	15.17
SRL + Language model	14.42	14.30	14.46
SRL + POS + Chunking	16.46	15.95	16.41
SRL + POS + NER	16.45	16.89	16.29
SRL + POS + Chunking + NER	16.33	16.36	16.27
SRL + POS + Chunking + NER + Synonyms	15.71	14.76	15.48
SRL + POS + Chunking + NER + Language model	14.63	14.44	14.50



Conclusions The paper proposed a general deep NN architecture for NLP. The architecture is extremely fast enabling us to take advantage of huge databases (e.g. 631 million words from Wikipedia). We showed our deep NN could be applied to various tasks such as SRL, NER, POS, chunking and language modeling. We demonstrated that learning tasks simultaneously can improve generalization performance. In particular, when training the SRL task jointly with our language model our architecture achieved state-of-the-art performance in SRL without any explicit syntactic features. This is an important result, given that the NLP community considers syntax as a mandatory feature for semantic extraction (Gildea & Palmer, 2001).

Advantages

1）Making full use of huge data sets；

2）The model is the better for generalization than others which based on the combination of feature-engineering and machine learning algorithms.

3）This is an important result, given that the NLP community considers syntax as a mandatory feature for semantic extraction

Disadvantages

1）The model is not avoid to use the hand-constructed feature vector which based on the feature-engineer, which manually constructed feature vectors from orthographic features (e.g., capitalization of the first character), dictionaries and lexicons.

2）There is no F1 score results for the combination of NER and deep neural network

3）A large number of hand-labeled data is necessary for the model, which is limited.

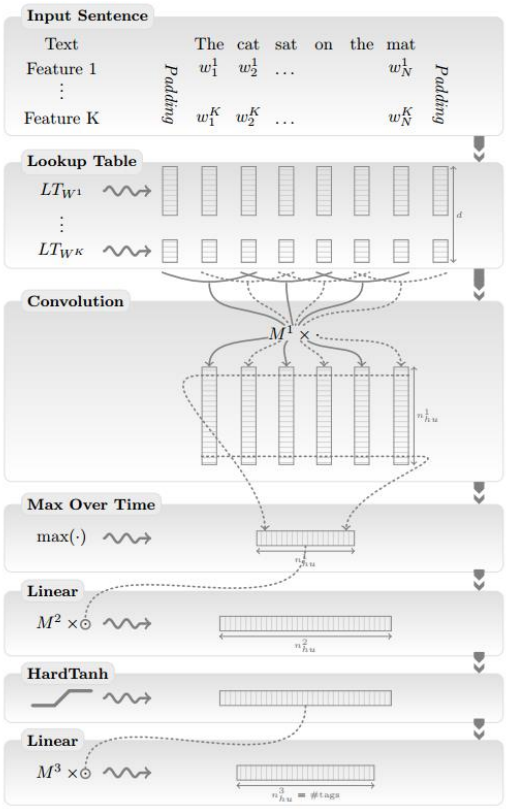
References

【31】Collobert R, Weston J. A unified architecture for natural language processing: Deep neural networks with multitask learning[C]//Proceedings of the 25th international conference on Machine learning （ICML'08）. ACM, 2008: 160-167.

【32】Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. Journal of machine learning research (JMLR'03), 2003, 3(Feb): 1137-1155.

【33】Collobert R, Weston J. Fast semantic extraction using a novel neural network architecture[C]//Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics （ACL'07）. 2007: 560-567.

6.2 The NER Systems for the Combination of Word Embedding and Neural Network



1) The first layer extracts features for each word.
2) The second layer extracts features from a window of words or from the whole sentence, treating it as a sequence with local and global structure (i.e., it is not treated like a bag of words).
3) The following layers are standard NN layers.

Datasets

1）Our first English corpus is the entire English Wikipedia. We have removed all paragraphs containing non-roman characters and all MediaWiki markups. The resulting text was tokenized using the Penn Treebank tokenizer script.14 The resulting data set contains about 631 million words. As in our previous experiments, we use a dictionary containing the 100,000 most common words in WSJ, with the same processing of capitals and numbers. Again, words outside the dictionary were replaced by the special “RARE” word.

2）Our second English corpus is composed by adding an extra 221 million words extracted from the Reuters RCV1 (Lewis et al., 2004) data set.15 We also extended the dictionary to 130,000 words by adding the 30,000 most common words in Reuters. This is useful in order to determine whether improvements can be achieved by further increasing the unlabeled data set size.

Experimental Results

Task		Benchmark	SENNA
Part of Speech (POS)	(Accuracy)	97.24 %	97.29 %
Chunking (CHUNK)	(F1)	94.29 %	94.32 %
Named Entity Recognition (NER)	(F1)	89.31 %	89.59 %
Parse Tree level 0 (PT0)	(F1)	91.94 %	92.25 %
Semantic Role Labeling (SRL)	(F1)	77.92 %	75.49 %

Conclusions

1）The authors achieved 89.59% F1 score on English CoNLL 2003 dataset by including gazetteers and SENNA embeddings.

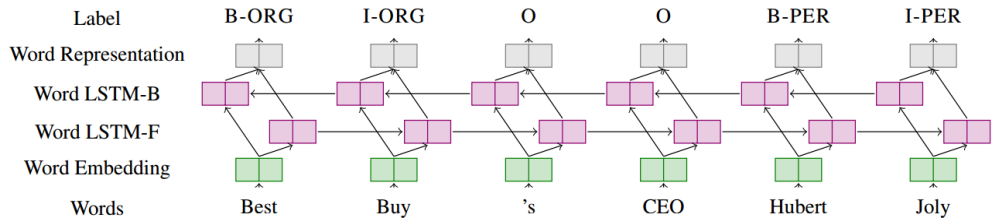
2）The paper rely on large unlabeled data sets and let the training algorithm discover internal representations that prove useful for all the tasks of interest.

Advantages

1）The model make full use of huge data set and unlabeled data, which is almost from scratch.

2）The model is the better for generalization than others.

3）Using the word embeddings vector instead of hand-constructed feature vectors, which is represented by n-dimension vector space.



4）The RAM and computational speed are most fast between systems based on the word embeddings, which is minimal computational requirements.

POS System	RAM (MB)	Time (s)
Toutanova et al. (2003)	800	64
Shen et al. (2007)	2200	833
SENNA	32	4

SRL System	RAM (MB)	Time (s)
Koomen et al. (2005)	3400	6253
SENNA	124	51

5）avoid task-specific engineering and disregarding a lot of prior knowledge

Disadvantages

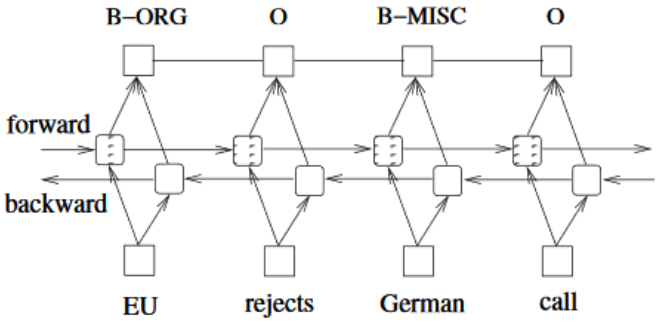
1）It is necessary to add others languages new corpora for F1 score.

2）The model is dependent to the word embedding.

3）It is necessary to resort the word embedding.

References

【34】Collobert R, Weston J, Bottou L, et al. Natural language processing (almost) from scratch[J]. Journal of Machine Learning Research （JMLR'11）, 2011, 12(Aug): 2493-2537.



Algorithm 1 Bidirectional LSTM CRF model training procedure

```
for each epoch do
    for each batch do
        1) bidirectional LSTM-CRF model forward pass:
            forward pass for forward state LSTM
            forward pass for backward state LSTM
        2) CRF layer forward and backward pass
        3) bidirectional LSTM-CRF model backward pass:
            backward pass for forward state LSTM
            backward pass for backward state LSTM
        4) update parameters
    end for
end for
```

1）In each epoch, we divide the whole training data to batches and process one batch at a time. Each batch contains a list of sentences which is determined by the parameter of batch size. For each batch, we first run bidirectional LSTM-CRF model forward pass which includes the forward pass for both forward state and backward state of LSTM. As a result, we get the output score for all tags at all positions.

2）We then run CRF layer forward and backward pass to compute gradients for network output and state transition edges. After that, we can back propagate the errors from the output to the input, which includes the backward pass for both forward and backward states of LSTM. 3）Finally we update the network parameters which include the state transition matrix, and the original bidirectional LSTM parameters.

Datasets CoNLL 2003 named entity tagging

Table 1: Size of sentences, tokens, and labels for training, validation and test sets.

		POS	CoNLL2000	CoNLL2003
training	sentence #	39831	8936	14987
	token #	950011	211727	204567
validation	sentence #	1699	N/A	3466
	token #	40068	N/A	51578
test	sentences #	2415	2012	3684
	token #	56671	47377	46666
	label #	45	22	9

Experimental Results 84.26% F1 score on English CoNLL 2003 dataset

Table 6: Comparison of F1 scores of different models for NER.

System	accuracy
Combination of HMM, Maxent etc. (Florian et al., 2003)	88.76
MaxEnt classifier (Chieu., 2003)	88.31
Semi-supervised model combination (Ando and Zhang., 2005)	89.31
Conv-CRF (Collobert et al., 2011)	81.47
Conv-CRF (Senna + Gazetteer) (Collobert et al., 2011)	89.59
CRF with Lexicon Infused Embeddings (Passos et al., 2014)	**90.90**
BI-LSTM-CRF (ours)	84.26

Conclusions The model is the first work of applying a BI-LSTM-CRF model to NLP benchmark sequence tagging data. The model is robust and it has less dependence on word embedding as compared to the observation in (Collobert et.al., 2011). It can achieve accurate tagging accuracy without resorting to word embedding.

Advantages

1) The experiments show that BI-LSTM-CRF model is robust

2) It has less dependence on word embedding as compared to previous observations (Collobert et al., 2011). It can produce accurate tagging performance without resorting to word embedding.

3) The first work of applying a BI-LSTM-CRF model to NLP benchmark sequence tagging data

4) Without resorting to word embedding

Disadvantages

1）The F1 score is less than previous models, which is necessary improved.

2）It is not enough for experiments which use English corpus only. The experimental results should add the new corpora.

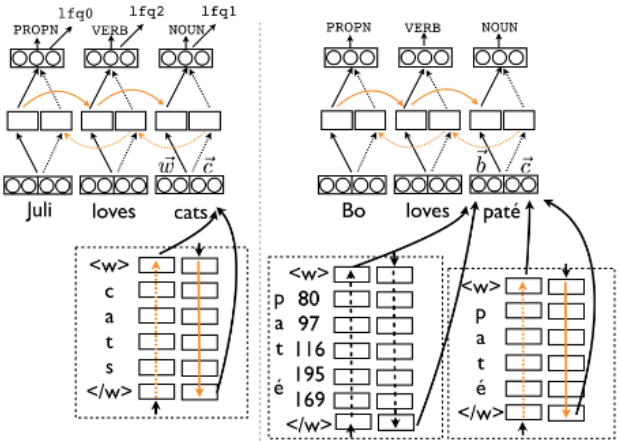
3）The research is not widely accepted by others.

References

【35】Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging[J]. arXiv preprint arXiv:1508.01991, 2015.

【36】Yao K, Peng B, Zhang Y, et al. Spoken language understanding using long short-term memory neural networks[C]//Spoken Language Technology Workshop (SLT), 2014 IEEE. IEEE, 2014: 189-194.

In this paper, their a) evaluate the effectiveness of different representations in bi-LSTMs, b) compare these models across a large set of languages and under varying conditions(data size, label noise) and c) propose a novel bi-LSTM model with auxiliary loss，which combines the POS tagging loss function with an auxiliary loss function that accounts for rare words.



Datasets For the multilingual experiments, we use the data from the Universal Dependencies project v1.2 (Nivre et al., 2015) (17 POS) with the canonical data splits. We consider all languages that have at least 60k tokens and are distributed with word forms, resulting in 22 languages. We also report accuracies on WSJ (45 POS) using the standard splits (Collins, 2002; Manning, 2011).

	COARSE	FINE		COARSE	FINE
ar	non-IE	Semitic	he	non-IE	Semitic
bg	Indoeuropean	Slavic	hi	Indoeuropean	Indo-Iranian
cs	Indoeuropean	Slavic	hr	Indoeuropean	Slavic
da	Indoeuropean	Germanic	id	non-IE	Austronesian
de	Indoeuropean	Germanic	it	Indoeuropean	Romance
en	Indoeuropean	Germanic	nl	Indoeuropean	Germanic
es	Indoeuropean	Romance	no	Indoeuropean	Germanic
eu	Language isolate		pl	Indoeuropean	Slavic
fa	Indoeuropean	Indo-Iranian	pt	Indoeuropean	Romance
fi	non-IE	Uralic	sl	Indoeuropean	Slavic
fr	Indoeuropean	Romance	sv	Indoeuropean	Germanic

Experimental Results

	BASELINES		BI-LSTM using:				$\vec{w}+\vec{c}$+POLYGLOT		OOV ACC		BTS
	TNT	CRF	$\vec{w}$	$\vec{c}$	$\vec{c}+\vec{b}$	$\vec{w}+\vec{c}$	bi-LSTM	FREQBIN	bi-LSTM	FREQBIN	
avg	94.61	94.27	96.00†	94.29	94.01	92.37	**96.50**	**96.52**	83.48	87.98	95.70
Indoeur.	94.70	94.58	96.15†	94.58	94.28	92.72	**96.63**	**96.63**	82.77	87.63	–
non-Indo.	94.57	93.62	95.67†	93.51	93.16	91.97	96.21	**96.28**	87.44	90.39	–
Germanic	93.27	93.21	95.09†	92.89	92.59	91.18	**95.55**	95.49	81.22	85.45	–
Romance	95.37	95.53	96.51†	94.76	94.49	94.71	**96.93**	**96.93**	81.31	86.07	–
Slavic	95.64	94.96	96.91†	96.45	96.26	91.79	97.42	**97.50**	86.66	91.69	–
ar	97.82	97.56	**98.91**	98.68	98.43	95.48	98.87	**98.91**	95.04	96.21	–
bg	96.84	96.36	98.02	97.89	97.78	95.12	**98.23**	97.97	87.40	90.56	97.84
cs	96.82	96.56	97.80	96.38	96.08	93.77	98.02	**98.24**	89.02	91.30	98.50
da	94.29	93.83	96.19	95.12	94.88	91.96	96.16	**96.35**	77.09	86.35	95.52
de	92.64	91.38	92.64	90.02	90.11	90.33	**93.51**	93.38	81.95	86.77	92.87
en	92.66	93.35	94.46	91.62	91.57	92.10	**95.17**	95.16	71.23	80.11	93.87
es	94.55	94.23	95.12	93.06	92.29	93.60	95.67	**95.74**	71.38	79.27	95.80
eu	93.35	91.63	94.70	92.48	92.72	88.00	95.38	**95.51**	79.87	84.30	–
fa	95.98	95.65	97.19	95.82	95.03	95.31	**97.60**	97.49	80.00	89.05	96.82
fi	93.59	90.32	94.85	90.25	89.15	87.95	95.74	**95.85**	86.34	88.85	95.48
fr	94.51	95.14	95.80	94.39	93.69	94.44	**96.20**	96.11	78.09	83.54	95.75
he	93.71	93.63	95.79	93.74	93.58	93.97	96.92	**96.96**	80.11	88.83	–
hi	94.53	96.00	96.23	93.40	92.99	95.99	96.97	**97.10**	81.19	85.27	–
hr	94.06	93.16	94.76	95.32	94.47	89.24	96.27	**96.82**	84.62	92.71	–
id	93.16	92.96	93.11	91.37	91.46	90.48	93.32	**93.41**	88.25	87.67	92.85
it	96.16	96.43	97.59	95.62	95.77	96.57	97.90	**97.95**	83.59	89.15	97.56
nl	88.54	90.03	93.32	89.11	87.74	84.96	**93.82**	93.30	76.62	75.95	–
no	96.31	96.21	97.57	95.87	95.75	94.39	**98.06**	98.03	92.05	93.72	–
pl	95.57	93.96	96.41	95.80	96.19	89.73	**97.63**	97.62	91.77	94.94	–
pt	96.27	96.32	97.53	95.96	96.20	94.24	**97.94**	97.90	92.16	92.33	–
sl	94.92	94.77	**97.55**	96.87	96.77	91.09	**96.97**	96.84	80.48	88.94	–
sv	95.19	94.45	96.36	95.57	95.50	93.32	96.60	**96.69**	88.37	89.80	95.57

Conclusions

1) Their evaluated token and sub-token-level representations for neural network-based part-of-speech tagging across 22 languages and proposed a novel multi-task bi-LSTM with auxiliary loss.
2) The auxiliary loss is effective at improving the accuracy of rare words.

Advantages

1) It is not sensitive to data set size and label noise
2) Across 22 languages, and the F1 score is better than previous systems.
3) It works especially well for morphologically complex languages

Disadvantages

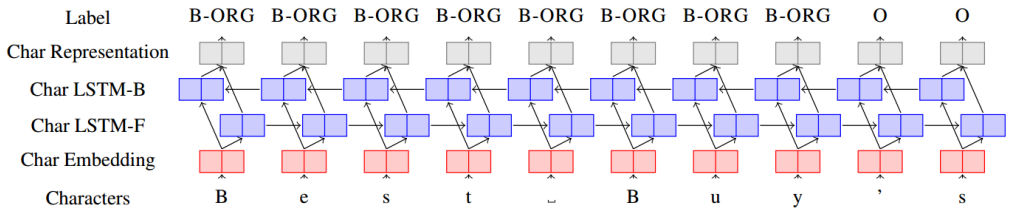
1）The model is increasingly complex , but the change of F1 score is not much obvious.

References

【37】Plank B, Søgaard A, Goldberg Y. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss[J]. arXiv preprint arXiv:1604.05529, 2016. **引用：127**

6.3 The NER Systems for the Combinations of Character Embedding and Neural Network

Sentence is represented as a sequence of characters.



Datasets

	Arabic	Czech	Dutch	English	German	Spanish	Turkish
Train	3988	4644	15806	14041	12152	8323	30000
Dev.	-	572	2895	3250	2867	1915	2237
Test	797[2]	577	5195	3453	3005	1517	3336

Experimental Results

	Arabic	Czech	Dutch	English	German	Spanish	Turkish
Best	84.30 [1] 79.90	75.61 [2] 68.38	82.84 [3] 78.08	91.21 [4] 80.79	78.76 [5] -	85.75 [5] -	91.94 [6] 82.28
Best w/o External	81.00 [7]	68.38 [2]	78.08 [3]	84.57 [3]	72.08 [3]	81.83 [3]	89.73 [2]
CharNER	78.72	72.19	79.36	84.52	70.12	82.18	91.30

Conclusions CharNER 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).

Advantages

1) character-level model. Taking characters as the primary representation is superior to considering words as the basic input unit.

2) The main contribution is to show that the same deep character level model is able to achieve good performance on multiple languages without hand engineered features or language specific external resources.

3）The F1 score of model is more better than previous models in English.

Disadvantages

1) There is nothing specific to NER in the model. It should be evaluate on other tasks such as part-of-speech tagging and shallow parsing. (**Multi-tasks**)

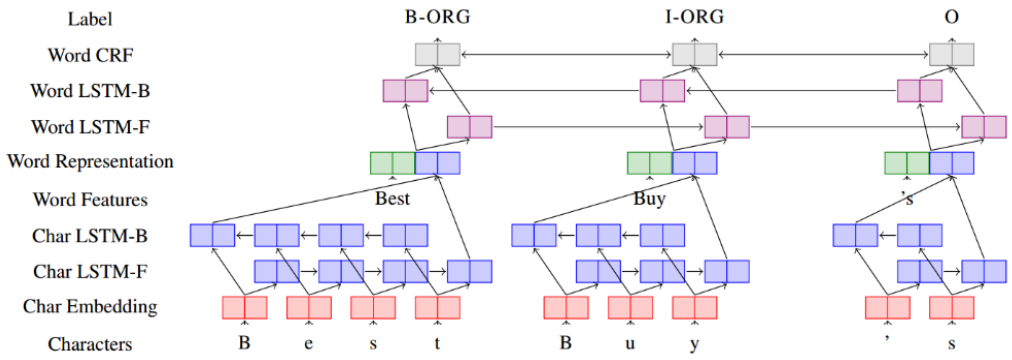
2) The robust experiment is lack.

3) It is necessary to add the new corpora for supporting the F1 score.

References

【 38 】 Kuru O, Can O A, Yuret D. Charner: Character-level named entity recognition[C]//Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics （COLING'16）: Technical Papers. 2016: 911-921.

6.4 The NER systems for the Combination of Character Embedding, Word Embedding and Neural Network



Datasets

Dataset		WSJ	CoNLL2003
Train	SENT	38,219	14,987
	TOKEN	912,344	204,567
Dev	SENT	5,527	3,466
	TOKEN	131,768	51,578
Test	SENT	5,462	3,684
	TOKEN	129,654	46,666

Experimental Results The F1 score of the model achieves 91.21%

Conclusions

1）It is a truly end-to-end model **relying on no** task-specific resources, feature engineering or data pre-processing.

Advantages

1）The system is truly end-to-end, requiring no feature engineering feature-engineer or data pre-processing.

Disadvantages

1） The model can be further improved by exploring multi-task learning approaches to combine more useful and correlated information. In a word, it is not support for multi-tasks learning.

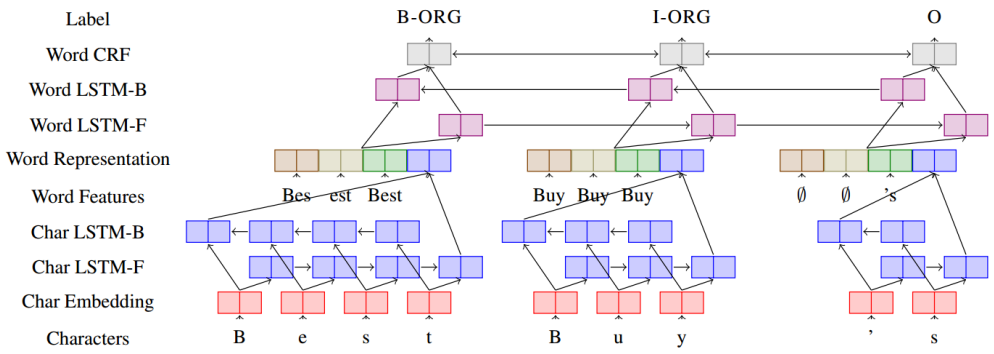
2） The model can be further explored in the different application areas such as bioinformatics, medical.

3） It is necessary to add the new corpora for supporting the F1 score.

References

【39】Ma X, Hovy E. End-to-end sequence labeling via bi-directional lstm-cnns-crf[J]. arXiv preprint arXiv:1603.01354, 2016. **引用：435**

6.5 The NER Systems for the combinations of Character Embedding, Word Embedding, affix model and Neural Network



数据集

	Dict	ES	NL	EN	DE
Gillick et al. (2016) – Byte-to-Span (BTS)	No	82.95	82.84	86.50	76.22
Yang et al. (2016)	No	85.77	85.19	91.26	-
Luo et al. (2015)	Yes	-	-	91.20	-
Chiu and Nichols (2016)	Yes	-	-	**91.62 (±0.33)**	-
Ma and Hovy (2016)	No	-	-	91.21	-
Lample et al. (2016)	No	85.75	81.74	90.94	78.76
Our base model (100 Epochs)	No	85.34	85.27	90.24	78.44
Our model (with Affixes) (100 Epochs)	No	86.92	87.50	90.69	78.56
Our model (with Affixes) (150 Epochs)	No	**87.26**	**87.54**	90.86	**79.01**

Experimental Results

	Dict	ES	NL	EN	DE
Gillick et al. (2016) – Byte-to-Span (BTS)	No	82.95	82.84	86.50	76.22
Yang et al. (2016)	No	85.77	85.19	91.26	-
Luo et al. (2015)	Yes	-	-	91.20	-
Chiu and Nichols (2016)	Yes	-	-	**91.62 (±0.33)**	-
Ma and Hovy (2016)	No	-	-	91.21	-
Lample et al. (2016)	No	85.75	81.74	90.94	78.76
Our base model (100 Epochs)	No	85.34	85.27	90.24	78.44
Our model (with Affixes) (100 Epochs)	No	86.92	87.50	90.69	78.56
Our model (with Affixes) (150 Epochs)	No	**87.26**	**87.54**	90.86	**79.01**

Model	drug	brand	group	drug_n	ML	drug	brand	group	drug_n	DB	Both
Unanue et al. (2017)	75.57	28.57	64.37	37.19	60.66	91.83	87.27	84.67	0	88.38	-
BASE	72	41.67	75.86	4.88	60.86	89.92	79.12	86.13	0	86.52	72.31
BASE+Affix(10)	79.25	44.44	85.39	32.73	69.71	92.09	86.60	87.41	20	88.93	78.39

Conclusions Straight-forward and language-independent approach shows performance gains compared to other neural systems for NER, achieving a new state of the art on Spanish, Dutch, and German NER as well as the MedLine portion of DrugNER

Advantages

1）The model is tested in multi-languages, which shows up the ability of generalization.

2）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.

Disadvantages

1) The model adds the most successful features from feature-engineering approaches：affixes, which is necessary to the labeled datasets. The pro-process is different to us.
2) It is necessary to add the new corpora for supporting the F1 score.

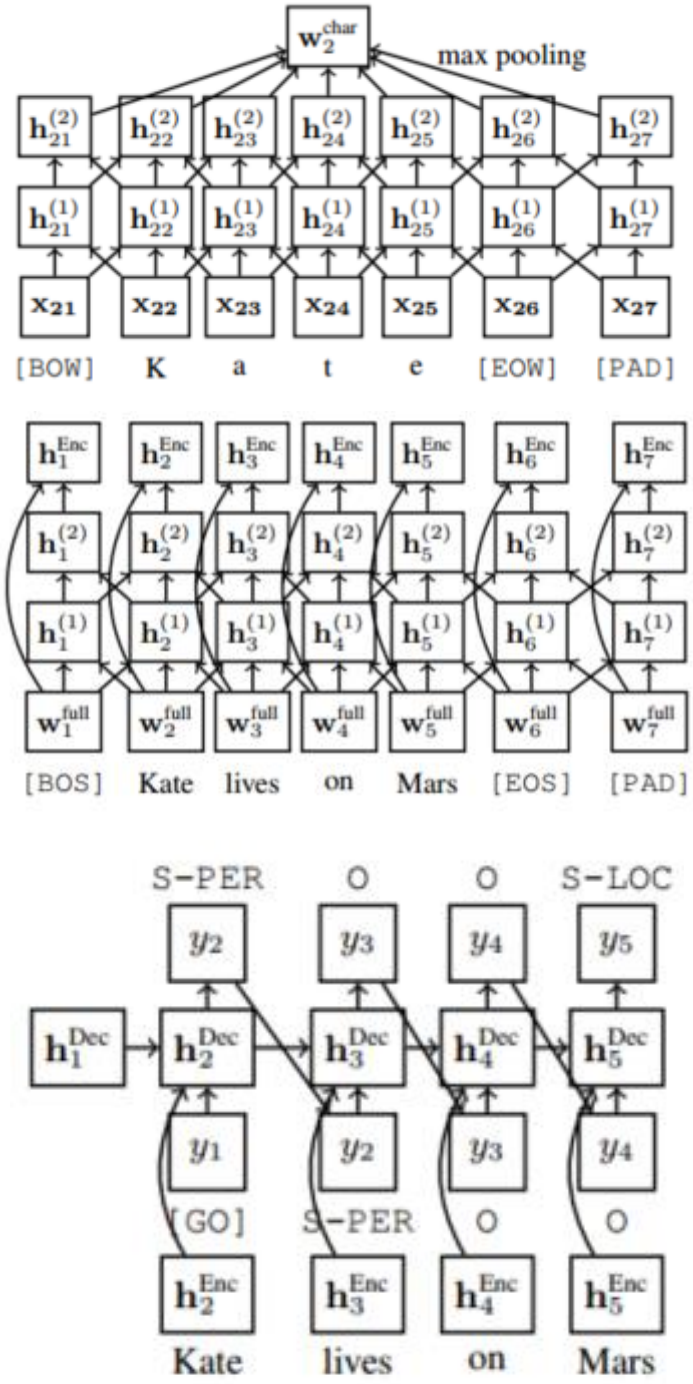
References

【40】Yadav V, Sharp R, Bethard S. Deep Affix Features Improve Neural Named Entity Recognizers[C]//Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics. 2018: 167-172.

6.6 The NER systems based on the Deep Active Learning

In this work, the combination of deep learning and active learning drastically reduced the number of labeled data. The CNN-CNN-LSTM model consisting of convolutional character and word encoders and a long short term memory (LSTM) tag decoder. The model achieves nearly state-of-the-art performance on standard datasets for the task while being computationally much more efficient than best performing models.

在这个工作中，深度学习和主动学习将大大减少标记数据的数量。模型的名称为：CNN-CNN-LSTM。这个模型 F1 评分上和其它模型接近，但是在计算性能上比当前最好的模型要好。



主动学习：通过“选择策略”主动从未标注的样本集中挑选部分（1 个或 N 个）样本让相关领域的专家进行标注；然后将标注过的样本增加到训练数据集给“学习模块”进行训练；当“学习模块”满足终止条件时即可结束程序，否则不断重复上述步骤获得更多的标注样本进行训练。

Datasets On the CoNLL-2003 English dataset

Experimental Results

Char	Word	Tag	Reference	F1	Sec/Epoch
None	CNN	CRF	Collobert et al. (2011)	88.67	-
None	LSTM	CRF	Huang et al. (2015)	90.10	-
LSTM	LSTM	CRF	Lample et al. (2016)	90.94	-
CNN	LSTM	CRF	Chiu & Nichols (2016)	90.91 ± 0.20	-
GRU	GRU	CRF	Yang et al. (2016)	90.94	-
None	Dilated CNN	CRF	Strubell et al. (2017)	90.54 ± 0.18	-
LSTM	LSTM	LSTM		90.89 ± 0.19	49
CNN	LSTM	LSTM		90.58 ± 0.28	11
CNN	CNN	LSTM		90.69 ± 0.19	11
CNN	CNN	CRF		90.35 ± 0.24	12

Conclusions The model use deep active learning algorithms for NER and empirically demonstrated that it achieves state-of-the-art performance with much less data than models trained in the standard supervised fashion

Advantages

1） The model achieves the **incremental training** with each batch of new labels: mix newly annotated samples with the older ones, and update the neural network weights for a small number of epochs, before querying for labels in a new round. This modification drastically **reduces the computational requirements of active methods and makes it practical to deploy them**.

2) **The model contains** convolutional character-level encoder, convolutional word-level encoder, and long short term memory (LSTM) tag decoder, which **trains much faster than other deep models**.

3) The model **introduces a simple uncertainty-based heuristic for active learning.**

Disadvantages

1） The model is necessary to the labeled data sets is used in training steps, but it is necessary to sample a little of data only.

2） It is necessary to add the new corpora for supporting the F1 score.

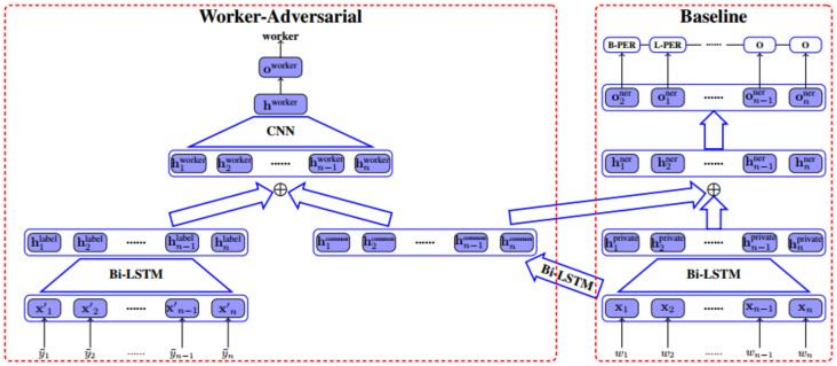
References

【41】Shen Y, Yun H, Lipton Z C, et al. Deep Active Learning for Named Entity Recognition[J]. arXiv preprint arXiv:1707.05928, 2017. **引用：21**

6.7 The NER Systems based on the Adversarial Learning

To quickly obtain new labeled data, we can choose crowd-sourcing as an alternative way at lower

cost in a short time. But as an exchange, crowd annotations from non-experts may be of lower quality than those from experts. To make full use of noisy sequence labels, the following model is proposed. Inspired by adversarial learning, the model uses a common Bi-LSTM and a private Bi-LSTM for representing annotator-generic and -specific information.



Datasets The datasets include labeled datasets and unlabeled datasets.

	#Sent	AvgLen	Kappa
DL-PS	16,948	9.21	0.6033
UC-MT	2,337	34.97	0.7437
UC-UQ	2,300	7.69	0.7529

Experimental Results

Model	Data: EC-MT		
	P	R	F1
CRF	75.12	66.67	70.64
LSTM-CRF	75.02	72.84	73.91
LSTM-Crowd	73.81	**75.18**	74.49
ALCrowd	**76.33**	74.00	**75.15**
	Data: EC-UQ		
CRF	65.45	55.33	59.96
LSTM-CRF	71.96	66.55	69.15
LSTM-Crowd	67.51	**71.10**	69.26
ALCrowd	**74.72**	68.60	**71.53**

Conclusions The experimental results show that the proposed approach outperforms strong baseline systems.

Advantages

1) The adversarial learning makes full use of noisy sequence labels.

2) The model achieves better scores than strong baseline systems.

Disadvantages

1) The model cannot handle entities separated by other entities or non-entity words.

2) It is necessary to add the new corpora for supporting the F1 score.

References

【42】Yang Y S, Zhang M, Chen W, et al. Adversarial Learning for Chinese NER from Crowd Annotations[C]// Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'18), 2018.

【43】Bekoulis G, Deleu J, Demeester T, et al. Adversarial training for multi-context joint entity and relation extraction[C]. EMNLP 2018.
【44】Bekoulis G, Deleu J, Demeester T, et al. Joint entity recognition and relation extraction as a multi-head selection problem[J]. arXiv preprint arXiv:1804.07847, 2018.

7 Conclusions

1）Neural network models generally outperform feature-engineered models.
2）The combination of character and word hybrid neural networks generally outperform other representational choices.
3）The current methods make full use of unlabeled, large number of data sets.
4）Another interesting direction is to apply models to data from other domains such as social media (Twitter and Weibo).
5）Can other supervised learning methods replace affixes of feature engineering methods in the Section 6.5?
6）There is still interesting progress to be made by incorporating key features of past feature-engineered models into modern Neural Network architectures.
7）The combination of LSTM and CRF is still the focused research topic, especially in the different ways of using small corpus and training sets.
8）The adversarial learning makes full use of noisy sequence labels.
9）It should be explored for transfer learning, active learning and joint learning in NER systems.

8 The Personality Opinions

In my opinion, with the evolution of the computer sciences technologies, such as knowledge engineer, machine learning, deep neural network, active learning and adversarial learning, the progress of the NER systems. The **F1 score** and **other evaluation paradigms** achieve **much higher** on some corpora and new corpora. It is trends to combine NER systems and focused methods. It is common to get the best performance for the algorithms. But it is **weak** to the NER systems for the **specific domains**, it should be explored for us. It may be future work for the NER systems for the combinations of **character embedding**, **word embedding**, and **neural network** adds the **unsupervised learning**, **supervised learning** or **semi-supervised learning algorithms**.