Assignment 2
Named Entity Recognition

Advanced Natural Languages Engineering (G5114)

Peijie Wen

CandNo. 246674

University of Sussex, School of Engineering and Informatics

1. Introduction

Named Entity Recognition (NER) is a fundamental task in the field of Natural Language Processing (NLP). Its aim is to identify named entities with specific meanings from a large amount of unstructured text data and classify them into predefined entity types[1]. NER tasks can effectively extract key information from massive data and are widely applied in fields such as relationship extraction, information retrieval, text mining, and question answering. This essay discusses the early rule- and dictionary-based approaches to NER, statistical learning-based approaches and deep learning implementations of NER under the development of machine learning, focusing on the two machine learning-based approaches and comparing their performance through the findings of related projects. Ultimately, deep machine learning (e.g. the Bert model) gives better results with relatively sufficient training data and is more widely applicable. Likewise, the statistical machine learning-based approach has some advantages in that the relatively small size of the model and the cost of building it can achieve accurate recognition quickly, which is suitable for some small-scale NER scenarios.

2. Methods

2.1. Data analysis

The difficulties in data processing in the NER task are the diversity of entities, ambiguity, the complexity of entities, and the absence of supervisory data. In general, the difficulties with NER data fall into three broad categories

1. Out-of-vocabulary (OOV) or novel entities refer to new entities that are not included in the vocabulary or training data of a model, and hence have a lower frequency of occurrence. As time passes and various fields develop, a plethora of new entities arise, such as new products, technologies, and terminology, among others. These entities do not have a unified naming convention, which requires NER models to possess strong contextual reasoning abilities, enabling them to extract context information from the text and infer the correct boundaries and categories of entities. Resolving the OOV issue is a crucial challenge in NER tasks, and requires the use of contextual information, dictionary matching, remote supervision, and other methods to fully utilize text information and thereby improve the accuracy and generalization of NER models.

2. Ambiguity and Noise: Ambiguity and noise are common issues in text, where a single word may be a named entity in one instance, and a common noun in another, or even a different type of entity. The same entity may have multiple ways of expression, and different entities may exist in similar contextual environments. Therefore, before performing named entity recognition, additional named entity disambiguation tasks are required. These tasks may involve contextual semantic analysis, entity type determination, and entity linking, among others, all of which can improve the accuracy and coverage of NER, better addressing the issues of ambiguity and noise in text.
3. Informal Text: With the popularity of social media, people are increasingly sharing their lives, thoughts, and opinions on social platforms, resulting in a vast amount of text data on platforms such as Twitter. However, social media text is typically brief and employs colloquial language, while also containing a significant amount of abbreviations, slang, and homophones. These characteristics make NER tasks more challenging, requiring the use of more efficient and accurate methods for processing such data.

2.2. Rule-based and dictionary-based methods

Early NER methods primarily relied on rule-based templates constructed by linguistic experts based on language-specific properties. These templates matched and extracted entities from text data. Different datasets usually required specific rules constructed based on statistical information, punctuation, keywords, indicators and directional words, and center words. For example, in medical texts, domain knowledge and rules can be used to extract entities such as symptoms, medications, surgeries, and hospitals.

The rule-based approach is simple to implement, requires minimal annotated data, and can handle professional terminology and vocabulary well in specialized domains. Due to the clearly defined rules, it has strong interpretability and performs well in specific corpora. However, the rule-based approach also has several limitations. Firstly, it requires a large amount of domain-specific knowledge to develop and maintain rules, which can be time-consuming and labor-intensive. Especially for complex domains with many entity types, there can be issues with rules becoming cumbersome and difficult to maintain. Secondly, the rule-based approach has low fault tolerance when facing changes in text. For example, it may not be able to recognize spelling errors or named entities with different variations. Finally, the rule-based approach may not scale well to large datasets or domains with numerous entity types. Therefore, it is suitable for handling certain fixed domains, smaller datasets, or simpler tasks.

2.3. Statistical machine learning-based methods

With the rise of machine learning techniques, some researchers have employed machine learning methods to address the NER problem. This approach can overcome some of the limitations of rule-based and dictionary-based NER methods to a certain extent. This type of method can be classified into three categories: supervised learning, semi-supervised learning, and unsupervised learning.

Supervised learning for NER involves transforming the NER task into a classification problem, by using machine learning methods to construct feature vectors from labeled corpus data, and establishing a classification model to identify entities. The general process of this method typically includes: (1) acquiring raw data, (2) preprocessing the data, (3) selecting appropriate features based on the text information in the data, (4) assigning different weights to different features and choosing an appropriate classifier to train the feature vector to obtain a model, and (5) using the model to perform entity recognition tasks and evaluating the results. Common supervised classification models include Hidden Markov Models (HMM), **(Figure2.1)** Support Vector Machines (SVM), and Conditional Random Fields (CRF).

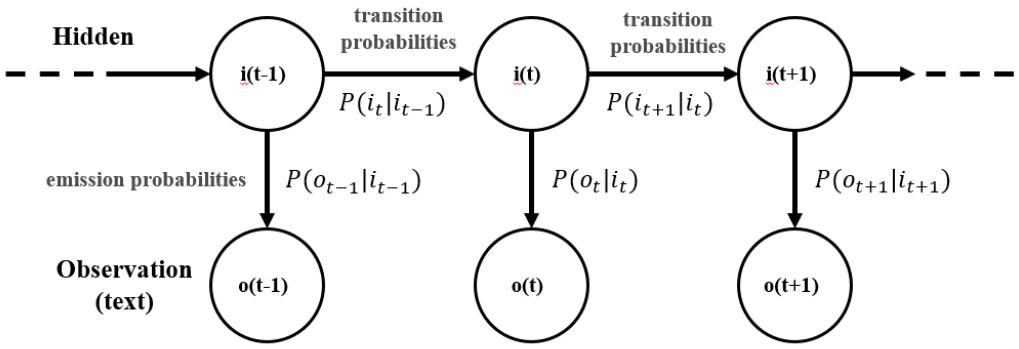

Figure 2.1 Hidden Markov Models

When the joint probability $P(y, x)$ is factorised in the form of the HMM, the associated conditional distribution $P(x|y)$ is a specific Linear-chain CRF

Semi-supervised NER methods were proposed to address the problem of requiring a large amount of hand-labeled training data for supervised learning methods. This approach utilizes a small number of labeled examples and a large amount of unlabeled data for NER research. The general process typically involves the following stages: (1) manually constructing an initial seed set, (2) generating related patterns based on named entity context information, (3) matching the generated patterns with test data to identify new named entities and generate new patterns to facilitate iteration, and (4) adding newly recognized named entities to the entity set.

Unsupervised learning techniques have been proposed to address the lack of annotated text for cross-domain and cross-lingual NER tasks. Unsupervised learning algorithms

do not require labeled data and make decisions based on unlabeled data. For example, clustering-based methods use similarity measures, such as distance measures, to cluster entities together, and then assign a label to each cluster manually or semi-automatically. These methods aim to discover the category information of data from their structural and distributional features without the need for any human-labeled data.

In general, this approach typically involves a combination of feature engineering and statistical learning techniques to learn patterns and structures in text corresponding to entities. Then, the trained model can be used to recognize and extract entities in new, unseen textual data.

One of the advantages of machine learning-based methods is that they can learn from data and generalize to new instances. The model can learn to identify patterns in text that are difficult or impossible to capture with handcrafted rules. Furthermore, machine learning-based methods are effective in complex domains or datasets with a large number of entity types.

Indeed, there are also some limitations to machine learning-based methods. Firstly, they require a large amount of labeled data for training, which can be time-consuming and expensive to obtain. Secondly, the performance of the model is highly dependent on the quality and representativeness of the labeled data. If the data is biased or incomplete, the model may not perform well on new, unseen data. Lastly, interpreting and explaining machine learning-based methods can be challenging, as the behavior of the model is determined by complex interactions between features and statistical learning algorithms.

2.4. Methods based on deep learning

In recent years, deep learning-based NER models have dominated and achieved good results. [2] Compared to feature-based methods, deep learning methods help to automatically discover hidden features. The general process of deep learning-based NER methods is as follows: (1) preprocess the input sequence, (2) vectorize the words in the input sequence, (3) perform semantic encoding on the word embeddings, and (4) perform decoding predictions based on the embedded context. **(Figure 2.2)**

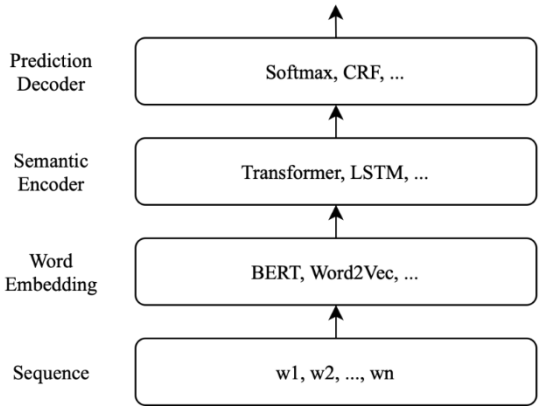

Figure 2.2 Method based on deep learning processing

There are various specific approaches for this type of task, and in this article, we will focus on the method based on the BERT model. BERT, [3] short for Bidirectional Encoder Representations from Transformers, is a pre-trained language model that utilizes Transformer structures to better capture the relationships between contexts. The BERT-based NER method generally adopts the Fine-tuning approach, **(Figure 2.3)** which involves fine-tuning the pre-trained BERT model using a specific NER dataset to extend its performance to the corresponding field. Specifically, the input words or characters are first converted into corresponding vector representations, which are then fed into the BERT model. After passing through multiple layers of Transformer encoders, context-aware word vector representations are generated. Finally, using classifiers such as Softmax, the NER labels for each word are predicted based on these word vector representations.

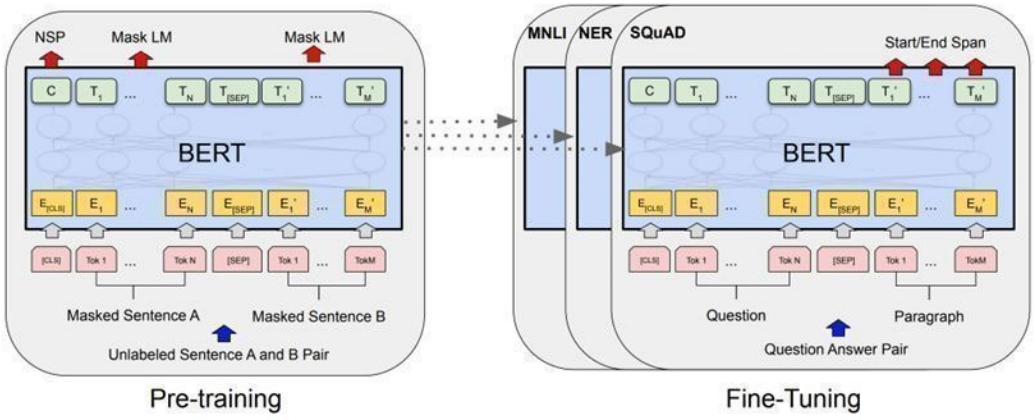

Figure 2.3 Bert model Fine-tuning

Compared to the first two methods, deep learning-based NER methods have achieved significant improvements in model performance. By utilizing the information contained in pre-trained models, these models can better understand natural language. Furthermore, deep learning models such as BERT can process longer texts and have better contextual understanding abilities. However, this approach also has some drawbacks. Firstly, deep learning models tend to be large and require a significant amount of computing resources for training and fine-tuning. Secondly, training deep learning models takes a long time, and they are highly dependent on training data and sensitive to hyperparameter settings. These are all significant issues that cannot be ignored.

3. Comparison

Theoretical comparison

The three methods differ in the techniques and approaches they use. Rule-based and dictionary-based methods rely on manually created rules and do not require training data. Statistical machine learning methods start with statistical analysis at the data level, and their models have strong generalization ability. Deep learning-based methods deepen the model's role in this process, by using the parameters in a deep neural network to save the features and patterns learned from the data. Although this approach requires a large amount of training data and computing resources, it also improves the effectiveness of the model.

Practical comparison

In terms of model performance, rule-based and dictionary-based methods show good results in specific domains, but their accuracy is limited by the quality of rules. The statistical machine learning approach has greatly improved the accuracy of models, but it still lags behind deep learning-based methods. Deep learning-based methods can achieve the best accuracy performance in many tasks. For example, on the CoNLL2003 dataset, deep learning-based methods can achieve F1 scores of 93.9[4] and 92.55[5], while statistical machine learning-based methods can achieve F1 scores of 82.57[6] and 87.24[7].

Regarding model efficiency, the first two methods are generally more efficient compared to the deep learning-based approach, thanks to their smaller model sizes and less computational resources required.

The rule-based and dictionary-based methods require significant effort in rule design and dictionary construction, making the initial implementation more difficult, but the subsequent usage more convenient. On the other hand, both the statistical machine learning and deep learning methods require training based on annotated data, which can be costly. In particular, the deep learning method requires more computational resources and training time, and also requires a certain amount of GPU resources to support the computation during model deployment.

4. Conclusion

These three techniques each have their own strengths and weaknesses, and their effectiveness in practical applications depends on specific situations and use cases.

Rule-based and dictionary-based methods are suitable for named entity recognition in specific domains and languages, such as medicine, law, and so on. If there is a need to quickly implement a small-scale NER system, this is a good choice.

Statistical machine learning-based methods are suitable for scenarios that require high accuracy and have relatively sufficient training data, such as named entity recognition in news articles.

Deep learning-based methods are suitable for handling named entity recognition tasks in multiple languages and domains, and this approach often achieves the best model performance when there are sufficient training resources available.

Therefore, when choosing a specific approach, it is necessary to consider the requirements for effectiveness in a particular scenario, the amount of annotated data, training device resources, and select the most suitable approach.

5. Future Work

At present, NER technology has reached a high level of maturity. By summarizing existing research on NER, future studies can explore several areas that still hold promising and intriguing ideas for further investigation.

Multi-task joint learning. NER, as a fundamental task, has various downstream applications. The traditional pipeline model, however, has limitations, where entity annotation errors in NER can propagate to subsequent tasks, leading to errors. Moreover, different tasks theoretically share information, but traditional independent training cannot exploit such potential information sharing. Multi-task learning, on the other hand, allows for information sharing and mutual influence among different tasks, which may further improve the effectiveness of NER. For instance, in the case of NER and entity linking tasks, entity linking aims to assign a unique identifier to entities mentioned in the text, with reference to knowledge bases such as Wikipedia for general domains, and Unified Medical Language System (UMLS) for biomedical domains. Linked entities can help to detect entity boundaries accurately and classify entity types correctly, and each sub-task can benefit from outputs of other sub-tasks.

Prompt-based learning for low-resource NER research. In recent years, NER tasks have extended in scope to include cross-domain, cross-task, and cross-language tasks. In general domains, most advanced NER models require extensive annotated data for training, which makes it difficult to extend them to new and resource-limited domains. Prompt-based learning can bridge the gap between low-resource and high-resource settings, facilitating knowledge transfer and enabling the easy acquisition of efficient NER models in different languages and domains.

Combining deep learning methods with auxiliary resources. Deep learning methods based on large-scale pre-trained word embedding have achieved excellent results in NER tasks. However, their performance in informal text is still not ideal. Some studies have shown that NER models benefit significantly from available auxiliary resources, such as geographical name dictionaries in user language. Therefore, this paper suggests that introducing auxiliary resources into deep learning methods may better understand user-generated content, as it combines pre-trained general knowledge background with specific domain resources. The challenge is how to obtain matching auxiliary resources for NER tasks in user-generated content or domain-specific text and how to effectively incorporate these resources into deep learning-based NER.

Reference

[1] Li, Jing, et al. "A survey on deep learning for named entity recognition." IEEE Transactions on Knowledge and Data Engineering 34.1 (2020): 50-70.

[2] Yadav, Vikas, and Steven Bethard. "A survey on recent advances in named entity recognition from deep learning models." arXiv preprint arXiv:1910.11470 (2019).

[3] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[4] Chen, Xiang, et al. "LightNER: a lightweight tuning paradigm for low-resource NER via pluggable prompting." arXiv preprint arXiv:2109.00720 (2021).

[5] Cui, Leyang, et al. "Template-based named entity recognition using BART." arXiv preprint arXiv:2106.01760 (2021).

[6] Thenmalar, S., J. Balaji, and T. V. Geetha. "Semi-supervised bootstrapping approach for named entity recognition." arXiv preprint arXiv:1511.06833 (2015).

[7] Krishnan, Vijay, and Christopher D. Manning. "An effective two-stage model for exploiting non-local dependencies in named entity recognition." Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the association for computational linguistics. 2006.