Review of Stanford's Name Entity Recognizer Toolkit and Its Usage for E-Commerce Queries

Prepared by Weijie Wang (weijiew2@illinous.edu)

Introduction

Name entity recognition (NER) is an interesting task under the category of information extraction. Its main goal is to locate potential entities, such as personal name, business name, location, from unstructured text data. Due to the nature of e-commerce, users' search queries are often ambiguous and full of noise. NER can help provide a new set of features from queries and can largely improve the performance of the recommendation system providing search feeds.

The Stanford name entity recognizer is a toolkit implemented in Java and is capable of extracting entities from general text [1]. It performs particularly well in English addressing entities in the categories of location, organization, person, and time [1][2].

Usage

The toolkit is provided in Java executables, so it is easy to import it as a part of your project or use it as a service. To try it out, you can load the GUI .bat or .sh and use one of their provided classifiers for 3 classes (Location, Person, Organization), 4 classes (+Misc), and 7 classes (+Money, Percent, Date, Time) recognition. You can also try it on their demo website [2]. The toolkit is also included in the NLTK library, which you can import by `from nltk.tag.stanford import StanfordNERTagger`.

Architecture and implementation detail

The general idea behind the Stanford NER is the Linear chain Conditional Random Field (CRF) model [3]. A CRF [4] is a sequential model representing the probability of a hidden state sequence given selected features like the surrounding words, the surrounding POS tags, and the current word n-grams. The output will be later fed into a Gibbs sampling with annealing [5]. They chose Gibbs sampling over the Viterbi algorithm which is often used in this scenario because the Viterbi algorithm puts an assumption on traceable model inference which is often broken by the nature of natural language. They also did a comparison in their paper. The output of the Gibbs sampling can be used to predict the entity types.

E-commerce application

A good NER can be a great help for users' search queries for any e-commerce website. For example, considering the query "Nike shoes", if we just use it to match against our product titles, we will for sure miss "Nike Air Max 720 for men" or "Nike Air Jordan 1 mid size". Even if we break the query and do "Nike OR shoes", we are giving these 2 words the same weight and will get a combination of Nike products and random shoes. However, if we use Stanford's NER which finds out that Nike is an entity of type organization name [2], we can search specifically for products tagged as shoes and with Nike in their title. We can even pre-generate product collections for each notable organization entity and show the collection tile on the side.

The NER can also be used to improve the performance of search autocomplete. When we generate candidates, we can put entities like "Adidas" as the top choices for "adi-"even if the count of "Adibas" is larger (it is possible if your company does not sell many Adidas products).

Conclusion

The Stanford NER provides a good tool for entity extraction and it can be useful in the context of e-commerce.

Related Work

This article provides details regarding how to train your own Stanford NER with e-commerce data, but it did not cover the details of the Stanford NER architecture and how we can use Stanford NER to improve our recommendation system.

Reference

[1] Stanford CRF-NER website https://nlp.stanford.edu/software/CRF-NER.shtml
[2] Stanford ner demo http://nlp.stanford.edu:8080/ner/process
[3] Incorporating Non-local Information into Information
Extraction Systems by Gibbs Sampling
https://nlp.stanford.edu/~manning/papers/gibbscrf3.pdf
[4] Conditional Random Fields: Probabilistic Models for Segmenting
and Labeling Sequence Data
https://repository.upenn.edu/cgi/viewcontent.cgi?article=1162&context=cis_papers
[5] Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images
https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1083.4723&rep=rep1&type=pdf