QAC386 Final Report

Ping-Jung Liu

Introduction

Named Entity Recognition (NER) has emerged as an important research field for numerous natural language processing applications. It is a form of information extraction in which we seek to classify named entities in each document into person name, organization, location, and other sorts of nouns. While the World Wide Web provides a wide variety of scattered information, NER can integrate these data into structured form and create a new but tremendous source of information, thereby creating more accurate internet search engine, or generate the lists of named entities before newspapers articles. For example, when is the word “Washington” used as the name of a person instead of the name of a city.

This project specifically focused on extracting the geographical locations from a chronicle of 528 years, beginning in the First year of Diocletian, down to the Second Year of Michael and the Latter’s Son Theophylaktos. The final goal of the greater project will be to tag related people names and locations with the years they appeared. This initial project will serve as a POC to show that the concepts of Named Entities Recognition can indeed extract, if not all, a great portion of the ancient locations from the chronicle.

Dataset

The Chronicle was directly provided in txt format, which granted me the ability to freely perform data preparation on the file, including cutting, tagging, and filtering the file. There were page numbers and other appendix tags included in the file; unfortunately, I did not find the appropriate methods to track down and delete them, so at the beginning I manually deleted these numbers and collapsed the first seventy pages into one huge paragraph, but the results were proven to be identical to those generated from the original text file.

I also made uses of dataset from an online project named Pleiades, which maintains a database for ancient geographic names. This file saved the names in three different formats: Title, Id, and NameTranliterated; The 1st and 3rd column contained numerous elements with unidentified symbols, so I took the Id column and changed them all into capital letters for matching purpose.

Methods

My first thoughts were to pattern match specific phrases like “from .....”, “to ……”, “go to ……”, but the results for the first seventy pages were not ideal and contained a huge portion of unrelated nouns and people names.

I then turned back to NLP and discovered its NER utility’ first I used the Named Entity Recognition tool in the openNLP library to tag the places appeared in the text

that are recognized by the tool (The tool can also recognize People, Organizations, and others) and extracted them into a list.

Though the whole recognition had been contracted into one tool, it synthesized two main steps. First, it performed sentence and word tokenization to tag each word in the text to the correct part of speech by finding the results with highest probability through maximum entropy modeling. Given the POS tags the tool can obtain all those labeled with “NNP”, which means named entity in openNLP, and move on to the recognition process.

The named entities recognition tools in openNLP essentially treat sentences as a sequence of word Wi; the goal is then to predict the most probable sequence of entity tags Ti provided the input sequence Wi. It once again used the maximum entropy model, which I will discuss in detail in the next section.

After using NER, I looped through the database from Pleiades to calculate another list of locations containing the overlap of the database in the chronicle using the R built in function grepl. By combing the two lists and eliminating the repetitions within, a single list of locations was obtained with my method. The list contained of 600 locations and could be useful in determining the geographical flow of ancient empires and its relations with Emperors and Bishops.

Maximum Entropy Model

Both the part of speech tagging and named entity recognition implement maximum entropy modeling; it is a probability estimation technique and is widely used for natural language processing. The main motivation of maximum entropy modeling was to generate a model that assumes nothing about what is unknown to produce accurate results, so this part will provide a rudimentary but thorough introduction to maximum entropy modeling.

Entropy itself is a measure of unpredictability of a state. If a state is relatively unpredictable, then performing the activity should theoretically provide more new information. In contradiction, a coin with two heads will have zero entropy because the outcome is nonnegotiable. Entropy is then defined as, for a discrete random variable X, $H(X) = -\sum_{k \geq 1} p_k \log p_k$.

There are two main reasons behind maximum entropy. First, states in physical world tend to move toward maximum entropy configuration overtime. Second, the model with least prior knowledge about the data can make the most accurate predictions. A direct but simple demonstration of maximum entropy modeling can be shown by finding the p(head) with the best accuracy of predicting right results, given the constraint that p(head) + p(tail) = 1. It is obvious that H(X) will reach its maximum 1 when p(head) = 0.5 and all other p(head) will result in H(X) strictly less than 1. We

can then treat NER as a sort of sequence prediction: given a sequence of named entities, predict the best sequence of name tags; possible features include but not limit to: previous tag, next tag, n-grams.

Comparison Between NER and Pleiades

The results from regular expressions contain way too many noises, possibly caused by the lack of reliable knowledge about how location named entities generally occurred, so I only made a comparison between the results from NER in openNLP and the data from Pleiades.

First I would like to point out that the NER tools are not based on an existing library of known locations because in fact the results contained a small portion elements like “great”, “277”, which should never be considered locations in any case.

The NER tools recognized 232 unique locations from the text file, while the Pleiades provided 430 unique location matchings. It turned out 18 percent of the locations extracted by NER also exist in the lists found by Pleiades, so the overlap is small, which implied that, first, the NER tools have space of improvement in recognizing ancient English grammars; second, the combination of the lists can provide a tremendous boost to the amount of unique locations in the chronicle.

Applications

The NER tools in the openNLP library does not require huge amount of texts to generate accurate results, which means this method could also be used to recognize named entities in smaller articles like those from news or even twits.

Even though I did not continue after obtaining the lists of locations, I noticed that the description of each year always start with “In this year”; this feature could be extremely useful in separating the text into chunks according to years. Researchers could then figure out the locations and people and tag them to the suitable years.

Potential output includes but not limit to:

Diocletian, 6th year Narses, 2nd year Gaius	**Thebes, Alexandria**
Peter, 5th year Vitalius, 2nd year	**Africa, Egypt**
Theodosios, 7th year Vararanes	**Syria, Libya**