Comparison of NLP Tools on Named-Entity Recognition of Resumes

Artsiom Strok
NetID: astrok2
astrok2@illinois.edu

***Abstract*—Named entity recognition (NER) – is an information extraction technique that automatically identifies named entities in a text and classifies them into predefined categories such as person name, organization, location. In this article, I am going to compare open-source APIs SpaCy and NLTK (Stanford NER Tagger) as well as SaaS APIs Google Cloud Natural Language API, Amazon Comprehend Text Analysis API, Microsoft Azure Text Analytics API on Named-Entity Recognition in the text of resumes.**

***Index Terms*—Resume, NER, SpaCy, NLTK, Stanford NER Tagger, Google Cloud Natural Language API, Amazon Comprehend Text Analysis API, Microsoft Azure Text Analytics API**

I. Introduction

This document is a model and instructions for LaTeX. Please observe the conference page limits.

A. SpaCy

SpaCy[1] is an open-source software library for advanced natural language processing, written in the programming languages Python and Cython The library is published under the MIT license and its main developers are Matthew Honnibal and Ines Montani, the founders of the software company Explosion.

Unlike NLTK, which is widely used for teaching and research, spaCy focuses on providing software for production usage. As of version 1.0, spaCy also supports deep learning workflows that allow connecting statistical models trained by popular machine learning libraries like TensorFlow, PyTorch or MXNet through its own machine learning library Thinc. Using Thinc as its backend, spaCy features convolutional neural network models for part-of-speech tagging, dependency parsing, text categorization and named entity recognition (NER). Prebuilt statistical neural network models to perform these task are available for English, German, Greek, Spanish, Portuguese, French, Italian, Dutch, Lithuanian and Norwegian, and there is also a multi-language NER model. Additional support for tokenization for more than 50 languages allows users to train custom models on their own datasets as well

B. Stanford NER

Stanford NER[2] is a Java implementation of a Named Entity Recognizer. Named Entity Recognition (NER) labels sequences of words in a text which are the names of things, such as person and company names, or gene and protein names. It comes with well-engineered feature extractors for Named Entity Recognition, and many options for defining feature extractors. Included with the download are good named entity recognizers for English, particularly for the 3 classes (PERSON, ORGANIZATION, LOCATION), and we also make available on this page various other models for different languages and circumstances, including models trained on just the CoNLL 2003 English training data.

Stanford NER is also known as CRFClassifier. The software provides a general implementation of (arbitrary order) linear chain Conditional Random Field (CRF) sequence models. That is, by training your own models on labeled data, you can actually use this code to build sequence models for NER or any other task. (CRF models were pioneered by Lafferty, McCallum, and Pereira (2001); see Sutton and McCallum (2006) or Sutton and McCallum (2010) for more comprehensible introductions.)

C. Google Cloud Natural Language API

The Cloud Natural Language API[3] provides natural language understanding technologies to developers, including sentiment analysis, entity analysis, entity sentiment analysis, content classification, and syntax analysis. This API is part of the larger Cloud Machine Learning API family.

D. Amazon Comprehend Text Analysis API

Amazon Comprehend[4] uses natural language processing (NLP) to extract insights about the content of documents. Amazon Comprehend processes any text file in UTF-8 format. It develops insights by recognizing the entities, key phrases, language, sentiments, and other common elements in a document. Use Amazon Comprehend to create new products based on understanding the structure of documents. For example, using Amazon Comprehend you can search social networking feeds for mentions of products or scan an entire document repository for key phrases.

E. Microsoft Azure Text Analytics API

The Text Analytics API[5] is a cloud-based service that provides Natural Language Processing (NLP) features for

[1]https://en.wikipedia.org/wiki/SpaCy
[2]https://nlp.stanford.edu/software/CRF-NER.html
[3]https://cloud.google.com/natural-language/docs
[4]https://docs.aws.amazon.com/comprehend/latest/dg/what-is.html
[5]https://docs.microsoft.com/en-us/azure/cognitive-services/text-analytics/overview

text mining and text analysis, including: sentiment analysis, opinion mining, key phrase extraction, language detection, and named entity recognition. The API is a part of Azure Cognitive Services, a collection of machine learning and AI algorithms in the cloud for your development projects. You can use these features with the REST API, or the client library.

II. Dataset and Analysis Methodology

This dataset is a document annotation dataset used to perform NER on resumes from indeed.com, which was obtained from https://www.kaggle.com/dataturks/resume-entities-for-ner/home.

III. Results