CS 838 — Data Science: Principles, Algorithms, and Applications; Spring 2017

Stage 2: extracting structured information from raw data

Trang Ho, Thomas Ngo, Qinyuan Sun

Table of Contents

1.Introduction

In this project stage, our team performed information extraction (IE) from natural text documents by using a supervised learning approach. In particular, we extracted names of tertiary educational affiliations from 300 The New York Times (https://www.nytimes.com/) articles.

We manually marked up the educational afflliation names with < pos >...</ pos > to indicate positive examples for the supervised learning. Some of the positive examples including "University of California, Berkeley", "University of Arizona", "Harvard Business School" and "California State University-Los Angeles".

2.Dataset

We set aside 100 text documents as test set (https://github.com/TrangHo/cs838-code/tree/master/test-examples) to generate testing examples and use the rest of text documents as train set (https://github.com/TrangHo/cs838-code/tree/master/train-texts) to generate training examples.

	Num. of documents	Num. of positive examples	Num. of negative examples
Training Set I	200	725	1948
Testing Set J	100	359	898
Total	300	1084	2846

Subsequently, we used four main regular-expression patterns (https://github.com/TrangHo/cs838-code/blob/master/src/lib/constants/patterns.py) to create a pool of potential negative-example candidates. The patterns suggest the following characteristics of negative candidates:

- having at least 2 words and all of them are capitalized
- having 2 captialized words with a prefix of at/from/in
- consisting of 3 or 4 words with a suffix of a noun usually goes with univerisities such as professor/student/etc.
- consisting of 3 or words with a prefix of a verb usually goes with with universities such as attend/receive

The final negative examples were then randomly selected from the pool.

3.Training

To generate feature vectors from the positive and negative examples, we eventually designed 17 functions that (1) take a string and its surrounding texts, and (2) output either zero or one. Therefore our feature vector has 17 dimensions.

The machine learning algoirthms we employed are:

- Support vector machine
- Decision tree
- Random forest
- Linear regresion
- Logistic regression
- Multilayer perceptron neural network

We initially had only 16 features. The average precision and recall of 5-fold cross-validation are listed as follow. However, the results of our classifiers were close but did not meet the requirement of having (1) precision of 90% or higher and (2) recall of 50% or higher. After inspecting the false positives and false negatives, we found out that a prevalent problem was that single-word university names (such as Yale, Standford, and Columbia) were wrongly classifed as negatives. As a result, we added a dictionary of short names for popular universities for these case as feature 17. This feature significantly increases both precisons and recalls of all classifiers.

Precision & Recall with 16 Features

Machine Learning Algorithm	Ave CV Precision	Ave CV Recall	F1
Support Vector Machine	0.92	0.49	0.64
Decsion Tree	0.89	0.54	0.67
Random Forest	0.89	0.54	0.67
Logistic Regression	0.90	0.50	0.64
Neural Network	0.88	0.54	0.67

Precision & Recall with 17 Features

Machine Learning Algorithm	Ave CV Precision	Ave CV Recall	F1
Support Vector Machine	0.95	0.70	0.81
Decsion Tree	0.93	0.72	0.81
Random Forest	0.92	0.74	0.82
Linear Regression	0.97	0.67	0.79
Logistic Regression	0.95	0.70	0.81
Neural Network	0.92	0.73	0.81

We chose Support Vector Machine as our classifier. We trained the classifier with all the training examples and tested on the testing examples. The results are shown in the following table.

Type	Precision	Recall	F1
TRAIN	0.93	0.73	0.82
TEST	0.97	0.72	0.83

4.Links

link (https://github.com/TrangHo/cs838-code/tree/master/texts) to 300 text document

link (https://github.com/TrangHo/cs838-code/tree/master/train-texts) to training set

link (https://github.com/TrangHo/cs838-code/tree/master/test-examples) to test set

link (https://github.com/TrangHo/cs838-code/tree/master/src) to source code

link (https://github.com/TrangHo/cs838-spring2017/raw/master/cs838-stage2.zip) to a zip file for stage 2 related documents

```
In [ ]:
```