Beyond Trigram: Learning to Distinguish Fake Article 11-761 Project Report

Yanran Hao
yanranh@andrew.cmu.edu

Yulan Huang
yulanh@andrew.cmu.edu

Ang Lu
alu1@andrew.cmu.edu

Heqing Ya
heqingy@andrew.cmu.edu

April 24, 2016

Abstract

1 Introduction

With the development of Language Technology, people have construct models to formally describe the natural language we are using in daily life. Interestingly, some of the models have been powerful enough to generate fake articles that could even mix the false with the genuine. Three MIT students developed a program called SCIgen[1] that can automatically generate an SCI article. They eventually fooled reviewers in several IEEE conferences with the generated articles. Therefore, a challenge is before us: how can we tell the machine generated articles from human written ones. Finding such methods will not only avoid being fooled, but can also help us build more powerful language model to make fun of the incompetent paper reviewers.

In this project, we have extracted several features, and built several classification models to distinguish the fake generated by trigram models and true articles. According to our experiment result , we have reached about 90% accuracy in cross validation and development set.

2 Feature Generation

3 Experiments

3.1 Data Description

The training set is 1000 articles, 500 real and 500 fake, of varying length. The development set is 200 articles, 100 fake, 100 real. Articles in development set is truncated to meet the length distribution in Table 3.1. Besides these two sets, we also use a 100 million word corpus of Broadcast News articles as external source for generation of specific feature.

3.2 Data Preprocessing

Articles from training set are truncated following the document length distribution in Table 3.1. The number of truncated training articles is 10065. Sentence per article distributions of original training set, truncated

[1]website: https://pdos.csail.mit.edu/archive/scigen/

#sentence	#article
1	20
2	10
3	10
4	10
5	10
7	10
10	10
15	10
20	10

Table 1: Article length distribution of dev set

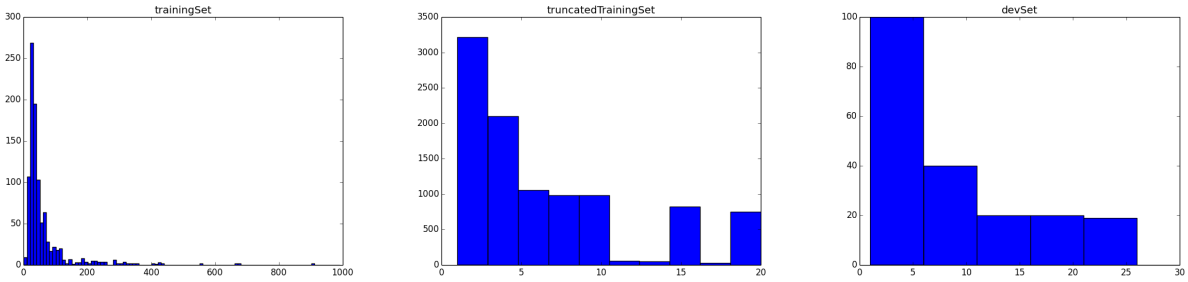
training set and development set are shown in Figure 1.

3.3 Classifier Choice

Popular classifiers for binary classification task are chosen as candidates, including KNN, logistic regression, SVM, gradient boosting (xgboost). Testing individual feature on development set and cross validation on training set, xgboost outperforms all the other algorithm by about 5%. Also it is very fast comparing to SVM. So we choose xgboost as the final classier.

3.4 Result

4 Conclusions



(a) Training set (original) (b) Training set (truncated) (c) Dev set

Figure 1: Article length distribution of different sets