Popularity Prediction of Hacker News Posts
Zhenglin Geng, Yang Yuan, Chao Wang \{zhenglin, yyuan16, cwang17\}@stanford.edu

Summary
A large amount of news are produced every day but only a small portion become popular. News popularity prediction is a meaningful task that can benefit content providers. And trying to predict in advance using only news content is even more challenging. We’re building a popularity prediction system by analyzing the corpus of Hacker News posts.

Features
N-gram: A contiguous sequence of n tokens are used as n-gram.
Counts & tf-idf: We both count n-gram patterns and transform it into term frequency-inverse document frequency.
Word2vec: Token is represented as a vector and Document vector is calculated by taking mean or tf-idf of word vectors.
Topics: We use non-negative matrix factorization to extract topics in the corpus and use distance to top 10 topics as features of each document.

Table1: Some Selected Topics When Topic Number Set To 20

Table: Topic 1 js javascript node react component python config
Topic 2 apple iPhone iPad FBI mac cook sir headphone
Topic 3 invest startup fund venture capital entrepreneur
Topic 4 game Pokémon VR Nintendo virtual oculus
Topic 5 email Clinton website inbox Gmail slack click
Topic 7 tesla vehicle musk autopilot elect autonomous
Topic 8 Facebook twitter advertise Zuckerberg messenger
Topic11 docker deploy cloud kubernetes cluster config
Topic 12 ai artificial neuronal deep alphago deepmind
Topic 14 uber ride city lyft vehicle hail self did transport

Results
Binary classification
Since predicting the popularity based on news text is difficult in nature[2], statistical significance is used to measure the effectiveness of our classifiers against random classifiers. We focus on precision on predicting popular news. We choose α = 0.05, the corresponding p-value = 0.1098, which means any classifier that achieves precision higher than p-value should be considered better than random guess.

Table2: Best Models We Get

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-EN-2gram</td>
<td>0.18</td>
<td>0.04</td>
</tr>
<tr>
<td>SVM-EN-2gram-tfidf-o</td>
<td>0.16</td>
<td>0.10</td>
</tr>
<tr>
<td>Bernoulli-NB-tfidf-o</td>
<td>0.16</td>
<td>0.20</td>
</tr>
<tr>
<td>Bynoulli-NB-tfidf-o</td>
<td>0.16</td>
<td>0.18</td>
</tr>
<tr>
<td>PA-w2v(500)-mean</td>
<td>0.15</td>
<td>0.13</td>
</tr>
<tr>
<td>SVM-w2v(200)-EN-tfidf-o</td>
<td>0.14</td>
<td>0.32</td>
</tr>
<tr>
<td>SVM-w2v(1500)-EN-tfidf-o</td>
<td>0.14</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Multinomial classification
Not better. Precision for each class is approximately the ratio of that class in our dataset.

Discussion
Despite the fact that predicting the popularity using news text is in general a difficult task[2] because of the noise of data, we are still able to find predictors that achieve significantly better precision than random predictors. To better understand our dataset, we also build topic visualization tools, which automatically generate meaningful topics. Our experiment has yielded very promising results on this part.

Future Work
We can obtain larger data set, incorporate more features such as title and publication time and further fine tune different classifiers to get better prediction results.

Reference