TweetStrap

Natarajan Chakrapani  
Kuldeep Kapade  
Karthik Reddy Vadde

Twitter Mentor : Stanislav Nikolov
Goals

• “Tweetstrap” a marketer to identify mini-celebrities who can provide max retweets of their test tweets.
• Value: See beyond the usual suspects.
• Predict apriori retweet counts, if a test marketing tweet is tweeted by mini-celebrities from various domains
• Rank the celebrities by predicted retweet counts, based on their recent tweeting behavior
Avg Retweet count by celebrity

Followers by celebrity domains

Retweet graph

Average number of Retweets

Celebrity IDs

Books
Business
Politics
Media
Design
Fashion
Tech
Assumptions

- Tweets span topics
- Celebrities talk about different topics. Leverage that to identify tweets that might be easy to make them tweet about for max. reach
- Retweet count = Network reach = influence
- Retweets from immediate followers are significant in influence on global retweet counts compared to retweets from non-followers (via twitter search)
Data

- Extracted tweets and retweets of recommended tweeters for different categories.
  - (84 users across 14 categories)

- Verified accounts. Why?
  - 61% of the tweets from these tweets get retweets.
Features Used : v1

- **Tweet Features**
  - Topic distribution of tweet (numeric vector of topic scores)
  - Topic Distribution of tweets aggregated by Author.

- **Social Features**
  - Local Retweet count from first level followers
  - Friend/Follower
  - Listed_count/Follower
Features Used : v2

- **Tweet Features**
  - Topic distribution of tweet (numeric vector of topic scores)
  - Topic Distribution of tweets aggregated by Author.

- **Social Features**
  - Local Retweet count from first level followers
  - Friend/Follower
  - Listed_count/Follower
Topic Modeling

- Approach 1: Latent Dirichlet Allocation (LDA) for topic modeling
- Approach 2: use uclassify API for topic modeling.

Use these topics as features for co-relating to retweet counts along with other user attributes.
Problems with LDA

- Tweet content too limited to suggest strong topics
- Semantically unrelated terms seen in topics
  - 2012-11-25 11:32:24.113 : INFO : topic #2:
    0.016*out + 0.012*check + 0.008*help + 0.006*@ +
    0.005*some + 0.005*celebrity + 0.005*one +
    0.004*need + 0.004*me + 0.004*can + 0.004*love +
    0.004*style + 0.004*$ + 0.004*news + 0.004*womens +
    0.003*years + 0.003*th + 0.003*these + 0.003*do +
    0.003*our
- Hashtag terms overlap across topics
- Twitter spam hard to control
Uclassify

- A web service providing a public topic model classifier
- Built on top of the Open Directory project
- Multi level SVM Classification done over millions of documents
- Ten high level categories identified: (Arts, Business, Computers, Games, Health, Home, Recreation, Science, Society, Sports)
Approach 1

- Predict the retweet count given a text or topic with user as the centre using Multi-linear regression.

\[ E(Y | X) = \alpha + \beta_1 X_1 + \cdots + \beta_p X_p \]

- Goal: For a given tweet, rank users based on predicted retweet counts.
Model 1: Multi Linear regression

<table>
<thead>
<tr>
<th>variable</th>
<th>coefficient</th>
<th>std. Error</th>
<th>t-statistic</th>
<th>prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>-365.478526</td>
<td>1016.165069</td>
<td>-0.359665</td>
<td>0.719138</td>
</tr>
<tr>
<td>x1</td>
<td>452.602204</td>
<td>1024.552589</td>
<td>0.441756</td>
<td>0.658716</td>
</tr>
<tr>
<td>x2</td>
<td>445.124916</td>
<td>1026.104080</td>
<td>0.433801</td>
<td>0.664482</td>
</tr>
<tr>
<td>x3</td>
<td>379.665338</td>
<td>1024.600601</td>
<td>0.370550</td>
<td>0.711014</td>
</tr>
<tr>
<td>x4</td>
<td>427.162107</td>
<td>1024.552589</td>
<td>0.441756</td>
<td>0.658716</td>
</tr>
<tr>
<td>x5</td>
<td>438.522748</td>
<td>1026.104080</td>
<td>0.433801</td>
<td>0.664482</td>
</tr>
<tr>
<td>x6</td>
<td>437.297918</td>
<td>1024.600601</td>
<td>0.370550</td>
<td>0.711014</td>
</tr>
<tr>
<td>x7</td>
<td>406.350527</td>
<td>1024.402410</td>
<td>0.396671</td>
<td>0.691654</td>
</tr>
<tr>
<td>x8</td>
<td>468.469618</td>
<td>1026.450134</td>
<td>0.456398</td>
<td>0.648156</td>
</tr>
<tr>
<td>x9</td>
<td>420.150035</td>
<td>1025.998174</td>
<td>0.409504</td>
<td>0.682216</td>
</tr>
<tr>
<td>x10</td>
<td>480.435207</td>
<td>1027.406405</td>
<td>0.467619</td>
<td>0.640101</td>
</tr>
<tr>
<td>x11</td>
<td>-591.189392</td>
<td>1076.731942</td>
<td>-0.549059</td>
<td>0.583029</td>
</tr>
<tr>
<td>x12</td>
<td>-2279.465626</td>
<td>335.733960</td>
<td>-6.789500</td>
<td>0.000000</td>
</tr>
<tr>
<td>x13</td>
<td>947855.916033</td>
<td>26979.168879</td>
<td>35.132880</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

Models stats

- R-squared: 0.483405
- Adjusted R-squared: 0.479900
- F-statistic: 137.915497
- Prob (F-statistic): 0.000000
- Durbin-Watson stat: 1.514446
- Omnibus stat: 2662.954024
- Prob (Omnibus stat): 0.000000
- JB stat: 828587.886089
- Prob (JB): 0.000000
- Log likelihood: -13696.033782
- Prob (Log likelihood): 0.000000
- AIC criterion: 14.207289
- Skew: 7.695159
- BIC criterion: 14.247659
- Kurtosis: 103.333552
Approach 2 : Support Vector Regression

- Develop models per-user.
- Non linear relationship among the variables.
- For test tweet, estimate local retweet count based on retweet counts seen from “similar” tweets from user model.
- Use a Gaussian kernel (rbf) to estimate a best fit curve.
- \text{Retweet\_count} = F(\text{tweet topic scores , local retweet count})
- Compare the relative retweet counts across users
Prediction process

- Topic model of tweets
- Local retweet counts
- Gaussian kernel

Per-user Retweet Prediction Model

Test Tweet

Retweet Count
Prediction error margins
Demo
Questions??