A Few Bad Votes Too Many? Towards Robust Ranking in Social Media

Jiang Bian
Yandong Liu
Eugene Agichtein
Hongyuan Zha

Georgia Tech
Emory University
Emory University
Georgia Tech
Outline

• Background and Motivation
• Learning Ranking Functions in Social Media
• Vote Spam in Social Media
• Experiments on Community Question Answering
Online Social Media

User Interactions:
Voting/Rating the content

Information need → Online Social Media

Yahoo! Answers
Thumbs up or down votes to answers

digg
votes to news or comments

YouTube
Broadcast Yourself
votes to videos or comments

share Information

...
Community Question Answering (CQA)

- users can express specific information needs by posting questions, and get direct responses authored by other web users.
- Both questions and answers are stored for future use
- Allow searchers to attempt to locate an answer to their question
- Existing answers can be voted on by any users who wants to share her evaluations of the answers

The quality of the content in this QA portals varies drastically [Agichtein et al. 2008]
User votes can provide crucial indicators into the quality and reliability of the content
User votes can help to improve the quality of ranking CQA content [Bian et al. 2008]
Vote Spam

• Not all user votes are reliable
  – Many “thumbs up” or “thumbs down” votes are generated without much thought
  – In some cases, users intend to game the system by promoting specific answers for fun or profit
  – We refer those bad or fraudulent votes as vote spam

• How to handle vote spam for robust ranking of social media content?
  – Yahoo! Team semi-automatically removes some of more obvious vote spam after the fact
  – It is not adequate
    • The amount and the patterns of vote spam evolve
    • Vote spam methods can change significantly due to varying popularity of content, specifics of media and topic

• Challenge
  – A robust method to train a ranking function that remains resilient to evolving vote spam attacks
Outline

• Background and Motivation
• Learning Ranking Functions in Social Media
• Vote Spam in Social Media
• Experiments on Community Question Answering
Social Content and User Votes in Social Media

- Topic thread poster
- Responses creator
- Voter

- Response 1
  - User Votes: \( p_1 \times p_1 \) \( n_1 \times n_1 \)

- Response 2
  - User Votes: \( p_2 \times p_2 \) \( n_2 \times n_2 \)

- ... 

- Response \( n \)
  - User Votes: \( p_n \times p_n \) \( n_n \times n_n \)

User Votes
Learning-based Approach

<query, topic, response>

Content features

Community interaction Features

User Votes

relevance

Quality

Preference

GBrank
Outline

• Background and Motivation
• Learning Ranking Functions in Social Media
• Vote Spam in Social Media
• Experiments on Community Question Answering
Vote Spam Attack Models

- Two main types of vote spam
  - Incorrect votes – not an expert
  - Malicious votes – promote some specific responses

Choose $\beta$ % topic threads to attack

Choose number of attackers based on $N(\mu, \sigma^2)$ for each chose thread

Choose one response to promote for each chosen thread

thumbs up vote spam
one thumb up vote to chosen response

Thumbs down vote spam
one thumb up vote to chosen response
AND one thumb down vote to each others
Outline

• Background and Motivation
• Learning Ranking Functions in Social Media
• Vote Spam in Social Media
• Experiments on Community Question Answering
Robust Ranking Method

- **GBrank** in QA retrieval [Bian et al. 2008]
  - Promising performance
  - User vote information provides much contribution to the high accuracy (no vote spam)

- Robust ranking method – **GBrank-robust**
  - Apply the general vote spam model to generate vote spam into unpolluted QA data
  - Train the ranking function based on new polluted data
  - Transfer more weight to other content and community interaction features

```
< qr, qst, ans >
```

- Content features
- Community interaction Features
- User Votes
  - relevance
  - Quality
  - Preference

Ranking function
Experimental Setup

• Dataset
  – Factoid questions from the TREC QA benchmarks:
    • Total question set: 3000 factoid questions from 1999 to 2006
    • 1250 factoid questions from total question set—have at least one similar question in the Yahoo! Answers archive

  – Question-answer collection dataset
    • To simulate a user’s experience with a community QA site
    • Submit each TREC query to Yahoo! Answers and retrieve up to 10 top-ranked questions according to Yahoo! Answer ranking
    • For each of Yahoo! Questions, we retrieve all of its answers
    • 89,642 \(<qr, qst, ans>\) tuples

  – Relevance Judgments
    • Automatically labels using the TREC factoid answer patterns
    • 17,711 tuples (19.8%) are labeled as “relevant”
    • 71,931 tuples (81.2%) are labeled as “non-relevant”
Experimental Setup

• Evaluation Metrics
  – Precision at $K$
    • For a given query, $P(K)$ reports the fraction of answers ranked in the top $K$ results that are labeled as relevant

  – Mean Reciprocal Rank (MRR)
    • The MRR of each individual query in the reciprocal of the rank at which the first relevant answer was returned

  – Mean Average of Precision (MAP)
    • The mean of average precision of all queries in the test set
Ranking Methods Compared

• Baseline:
  – Let “best answer” always be on top
  – Following answers are ranked in decreasing order by number of
    (thumbs up votes – thumbs down votes)

• GBrank:
  – Ranking function with textual and community interaction features
    and preference extracted from voting information

• GBrank-robust:
  – Similar to GBrank
  – The training data is polluted according to the chosen spam model
Experimental Results

• QA Retrieval
  – Vote spam model: $\beta\% = 10\%; N(\mu, \sigma^2) = N(3, 1^2)$
  – Training data: randomly select 800 TREC queries and all related QA
  – Testing data (polluted): remainder 450 TREC queries and all related QA
Robustness to Vote Spam

Thumbs up vote spam

Thumbs up&down vote spam

GBrank-robust

GBrank-robust

GBrank

GBrank

Baseline

Baseline
Analyzing Feature Contribution

<table>
<thead>
<tr>
<th>Info Gain</th>
<th>Feature Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.048</td>
<td>Similarity between query and question</td>
</tr>
<tr>
<td>0.045</td>
<td>Number of resolved questions of the answerer</td>
</tr>
<tr>
<td>0.043</td>
<td>Length ratio between query and answer</td>
</tr>
<tr>
<td>0.003</td>
<td>Number of thumbs down vote</td>
</tr>
<tr>
<td>0.029</td>
<td>Number of stars for the answerer</td>
</tr>
<tr>
<td>0.002</td>
<td>Number of thumb up vote</td>
</tr>
<tr>
<td>0.026</td>
<td>Similarity between query and qst+ans</td>
</tr>
<tr>
<td>0.018</td>
<td>Number of answer terms</td>
</tr>
</tbody>
</table>

No textual features
No community interaction features
Contribution and Future Work

• Contributions
  – A parameterized vote spam model to describe and analyze some common forms of vote spam
  – A method for increasing the robustness of ranking by injecting noise at training
  – A comprehensive evaluation on ranking performance for community question answering under a variety of simulated vote spam attacks, demonstrating robustness of our ranking

• Future work
  – Explore further the different spam strategies and corresponding robust ranking methods
Thank you!
Related Work

• Robustness of web search ranking to click spam
  – [Jansen 2006] reveal the influence of malicious clicks on online advertising
  – [Radlinkski 2006] present how click spam bias the ranking results
  – [Immorlica et al. 2005] demonstrate that a particular class of learning algorithm are resistant to click fraud in some sense

• Ranking the content in social media site [Bian et al. 2008]
  – Present a ranking framework to utilize user interaction information (including user votes) to retrieve high quality relevant content in social media