IIIT Hyderabad Team at TAC-2008-Opinion Tasks

Team: IIITSUM08,
Presented by: VasudevaVarma.
Outline

- Introduction – Tracks and Tasks
- Data preprocessing
- Approaches
- Results
- Observations
Aims at mining opinions from blog posts.
Tasks

- **Rigid List Questions**
  - Exact strings containing a list item
    - Expects a list of named entities as an answer
    - Evaluated using F-Measure
  - **Example**: Which countries would like to build nuclear power plants?

- **Squishy List Questions**
  - Strings (sentences) containing an answer to the question
  - **Example**: What features do people like in vista?
Data preprocessing

- Answers must be retrieved from Blog06 corpus
- Used top 50 document set (subset of Blog06)

Challenges

- Encoding
  - Different character encodings to UTF-8 encoding
- Identifying post and Extraction of Author
  - Different domains has different templates
    - Parser based on the domain
  - For blogs without proper template
    - Html to text conversion & regular expressions to extract author
Approaches

- Question Answering Track
  - Rigid List: Includes four steps
    - Question Classification
    - Post Retrieval
    - Answer Extraction
    - Answer Ranking
  - Squishy List: Includes three major steps
    - Question Analysis
    - Sentence opinion & polarity determination
    - Sentence Ranking

- Summarization Track
  - Similar to Squishy list approach in QA
Rigid List approach

Question Classification

Keywords

Polarity

Post Retrieval

Ranked Posts

Answer Extraction

Answer Candidates

Answer Ranking

Answer List

Question

Docs

IIT Hyderabad at TAC-2008
Rigid List approach

Question Classification

- Answer type
  - Classifier trained on labeled question set provided by UIUC
  - Using SVM to classify the question into coarse grained category
    - HUMAN, LOCATION, ORGANIZATION, NUMBER, ENTITY
  - Person -> Person & Author
- Polarity of the question is determined using Naïve Bayes.
- Ex: *Who likes Windows Vista?*
  - Answer type: Person, Polarity: Positive

Post Retrieval

- Post as a unit
- Lucene for indexing and retrieval
- Naïve Bayes to estimate the relevance of the post
  - Using $P(\text{post} | \text{question polarity})$ estimate
Rigid List approach

Answer Extraction
- Stanford Named Entity Recognizer
  - PERSON, LOCATION & ORGANIZATION
- Rule based NER
  - NUMBER & ENTITY
- Authors extracted during preprocessing

Answer Ranking
- Two features with equal weights
  - Relevance of the post to the question
  - Relevance of the post to the question polarity
Squishy List approach

- Squishy list QA is similar to descriptive QA

- In-house summarization system
  - Topped answering why, what & how questions
  - Query dependent (QD) Feature
    - Boosts the sentence which has question key words in it
  - Query Independent (QI) Feature
    - Boosts the most informative sentences using KL-Divergence
Squishy List approach

1. Docs
2. Sentence Breaker
3. Sentence Ranking
4. Duplicate Detector
5. Top N sentences
6. Question
7. Question Analysis
   - Polarity
Sentence Ranking

- Query Dependent
- Query Independent
- Opinion & Polarity

List of Sentences
Question
Polarity

Weighted Linear

List of Ranked Sentences
Squishy List approach

- Opinion & polarity determination as a feature (OPS)
  - Focuses on mining opinion sentences in the interest of question
  - Boosts the opinion sentences whose polarity matches with expected polarity
  - A two class classifier in two phases
    - Opinion/Non-opinion classification
    - Positive/Negative classification
  - OpinionScore = 0.3 \( p(\text{sentence, opinion}) \) + 0.7 \( p(\text{sentence, polarity class predicted}) \)
Training Data

- **Training data**
  - IMDB movie review data for opinion-non opinion classification
    - 5,000 opinion sentences
    - 5,000 non-opinion sentences
  - 130,000 reviews on products from Amazon for polarity classification
    - Review with rating $\geq 4 \Rightarrow$ positive else negative
    - 98,000 positive reviews
    - 32,000 negative reviews
# Model Generation

<table>
<thead>
<tr>
<th>Task</th>
<th>Opinion/Non opinion classification</th>
<th>Polarity determination</th>
</tr>
</thead>
<tbody>
<tr>
<td>QA Run 1</td>
<td>Naïve Bayes</td>
<td></td>
</tr>
<tr>
<td>QA Run 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Summarization Run 1</td>
<td>SVM-HMM</td>
<td>Probabilistic indexing model</td>
</tr>
<tr>
<td></td>
<td>Unigram, bag of words as features</td>
<td></td>
</tr>
<tr>
<td>Summarization Run 2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
QA Runs

- Run 1
  - Rigid List (approach described earlier)
  - Squishy List: Opinion score is used as a feature
    - QD, QI & OPS weights are 0.275, 0.325 & 0.4

- Run 2
  - Rigid List (same as run 1)
  - Squishy List: Opinion score is used as a filter
    - Opinion score <= 0.4, drop the sentence while ranking
    - QD & QI weights are 0.3 & 0.7
<table>
<thead>
<tr>
<th>Type</th>
<th>Run 1</th>
<th>Run 2</th>
<th>Best Run</th>
<th>Median of Runs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rigid List</td>
<td>0.131</td>
<td>0.131</td>
<td>0.156</td>
<td>0.063</td>
</tr>
<tr>
<td>Squishy List</td>
<td>0.186</td>
<td>0.165</td>
<td>0.186</td>
<td>0.091</td>
</tr>
<tr>
<td>Total</td>
<td>0.164</td>
<td>0.154</td>
<td>0.168</td>
<td>0.093</td>
</tr>
</tbody>
</table>
Summarization Runs

- Run 1: SentiWordNet (SWN) score as a feature
  - QD, QI & SWN weights are 0.4, 0.3 & 0.3

- Run 2: Opinion score is used as a feature
  - QD, QI & OPS weights are 0.5, 0.3 & 0.2

<table>
<thead>
<tr>
<th>Runs</th>
<th>F-Measure</th>
<th>Coherence</th>
<th>Readability</th>
<th>Responsiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>0.101</td>
<td>2.045</td>
<td>3.545</td>
<td>2.364</td>
</tr>
<tr>
<td>Run 2</td>
<td>0.102</td>
<td>2.045</td>
<td>3.545</td>
<td>2.500</td>
</tr>
</tbody>
</table>
Observations

■ Possible decrease in F-measure for Rigid List questions
  ■ Person -> Person & Author
    ■ Results in picking extra candidate answers
    ■ Decrease in precision

■ Possible reasons for failure of summarization
  ■ Not using the optional answer snippets provided
  ■ Improper weighting of features
Post TAC Experiment on Summarization Track (Run2)

- No change in the model
- Used snippets provided along with blog posts,
- Experimented with different weights for each of the three parameters. Evaluated our summaries manually using nugget judgments

Description of Experiment:

- Weights: 0.25, 0.35, 0.4 for Query Dependent (QD), Query Independent (QI), Opinion Feature (OF) respectively.

- Length of Summary is limited to 2500 characters for each query. (Previously we tried to fill total 7000 characters in the summary)

The Average F-Measure ($\beta=1$) score over 22 summaries improved from 0.102 $\rightarrow$ 0.199
Thank You

Questions/Comments: vv@iiit.ac.in