Mining Sentiment Classification from Political Web Logs

Kathleen Durant
WebKDD ‘06
August 20, 2006
Explosion of News and Opinions on the Web

- Substantial growth of people accessing the Internet for news
  - 3% in 1995, 20% in 2004
- Growth of web logs on the Web
  - 100,000 in 2002 to 4.8 million in 2004
- Growth in people reading Web logs
  - 2004 saw a 58% increase in readers of web logs
Sentiment Topic View of the Blog Space

- Web logs provide readily available opinions on a myriad of topics
- Sentiment classification separates opinions into two opposing camps
- Take advantage of opinions and tools to build a custom view of blog space by topic and opinion
Questions Investigated

- Can existing Machine learning techniques be successfully applied?
- Which techniques work well?
  - Naïve Bayes, Support Vector Machines
- What’s the effect of unbalanced class compositions on results?
  - Different camps write at different rates on particular topics
Research Statement

- **Apply sentiment classification to political web log posts**
  - Topic specific corpus
    - George W. Bush and the Iraq War
  - Domain Specific
    - Political Web log Posts

- **Judge – Joe Gandelman**
  - classified over 250 web logs

- **Classify Web log posts according to our judge’s sentiment class**
  - Right-voice
  - Left-voice
Segmentation of Data

- Data segmented by the Month
  - Leading to 25 different models
  - Small enough to limit the events discussed
  - Large enough to generate enough posts on topic
Dataset Representation via the Vector Space Model

- Feature set – terms occurring at least 5 times within the Month’s corpus
  - Unigrams with polarity of environment
    - Differentiate between “not support”, “support”
  - Bag-of-words framework
    - Order not important, “Bush is” = “Is Bush”
  - Presence Vectors
    - Given n features the post is represented as a n-dimensional vector
      - 0 feature not present in post
      - 1 feature is present
      - Example: {0,1,1,1,0}. 5 features feature 1 and feature 5 are not present, features 2,3,4 are.

WebKDD 2006 Workshop on Knowledge Discovery on the Web, Aug. 20, 2006, at KDD 2006, Philadelphia, PA, USA
Naïve Bayes Classification

Choose the category with the Maximum Posterior Probability

Prior for the red class
Calculate the product of the probabilities for each term in a post
Likelihood term, appears = total number of occurrences of term in class / total number of words in red category

Prior for the blue class

Posterior Probability = Prior * Likelihood
Support Vector Machines

\[ Wx + b = -1 \]

\[ Wx + b = 0 \]

\[ Wx + b = 1 \]

Margin = \( \frac{2}{|W|} \)
Web logs to Classifiers

- SVM
- Naive Bayes
- Balanced
- Inflated
- Small
- NB

On Topic

Unbalanced

Off Topic
Comparing Machine Learning Techniques

Off-the-shelf Machine Learning Techniques perform well

Naïve Bayes significantly outperforms Support Vector Machines

- 99.9% confidence level, CI [1.425, 3.489]
Class Composition found on the Web

- **Imbalance in the class ratio**
  - 14% of right-voice posts on topic
  - 24% of left-voice posts on topic
Unbalanced Large and Small Results by Category

Unbalanced Large

Unbalanced Small

Predictability


Right Voices  Left Voices

Predictability


Right Voices  Left Voices
Unbalanced and Balanced Results by Category

Unbalanced

Right Voices
Left Voices

Balanced

Right Voices
Left Voices

[Graph showing the predictability of unbalanced and balanced results over time by category, with red and blue lines representing right and left voices respectively.]
Conclusions

- Off-the-Shelf Machine Learning Techniques work pretty well
- Balanced Naïve Bayes significantly outperforms Support Vector Machines
  - SVM 75.47%, NB 78.06% [1.425,3.488]
- Balancing the classes helps keep the number of misclassified per category more balanced
  - Unbalanced classifiers: more Right-voices were consistently misclassified
  - Balanced classifiers: more Left-voices were misclassified 56% to 44% over time continuum