

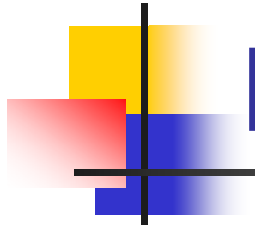


# Article Filtering for Conflict Forecasting

Benedict Lee and Cuong Than

Comp 540

4/25/2006



# Motivation

---

- One goal of the Ares Project is to predict conflict from events data extracted from various news sources
  - Sources: Reuters, BBC, The Associated Press
- Sources contain many irrelevant articles
  - We'd like to distinguish relevant articles from irrelevant articles



# Problem

---

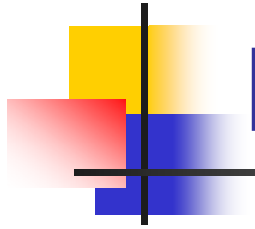
- Text Classification Problem
  - Relevant – International political interactions
  - Irrelevant – Everything else
- Additional context/information not necessarily available, so we classify solely on text of article



# General Approach

---

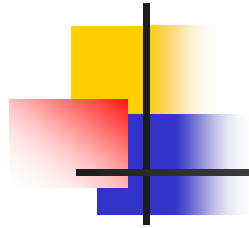
- Assertion: Relevant/irrelevant articles are similar to other relevant/irrelevant articles
- For each article, generate a Relevance and Irrelevance Rating based on “similarity” with training articles.
  - “Similarity” derived from the OKAPI BM25 Ranking formula (with Inflectional and Synonym Generation)



# Direct Comparison Classifier

---

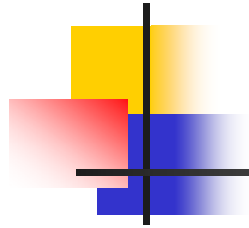
- The top N OKAPI scores are summed together to form Relevance/Irrelevance Ratings.
  - N is a tweakable parameter
- Article is classified as Relevant if  $\text{Relevance Rating} \geq \text{Irrelevance Rating}$



# Logistic Regression Classifier

---

- Input: relevance/irrelevance ratio.
- Output: 1 or 0, representing the Relevant and Irrelevant categories.
- Model:  $\Pr(y = 1 | x) = e^{h(x)} / (1 + e^{h(x)})$ , where  $h(x) = \theta^T x$ .
- Decision rule:  $x$  is classified as 1 if  $\Pr(y = 1 | x) \geq 0.5$ , or equivalently  $h(x) \geq 0$ .



# Fitting the Model

---

- Compute the likelihood over the training dataset.
- Maximize the likelihood to obtain a set of non-linear equations.
- Solve this set of equations by the IRLS method to find the parameter vector  $\theta$ .



# Dealing with Costs

---

- Motivation: misclassifying a relevant article costs more than misclassifying an irrelevant one.

	actual neg.	actual pos.
predict neg.	$c_{00}$	$c_{01}$
predict pos.	$c_{10}$	$c_{11}$

- Normally,  $c_{10} > c_{00}$  and  $c_{01} > c_{11}$ .





## Dealing with Costs (cont'd)

---

- Making decision: classify  $x$  as in category  $i$  if the risk function

$$\sum_j \Pr(j | x) c(i, j)$$

is minimized.

- $x$  is classified as in class 1 if  $\Pr(y = 1 | x) \geq p^*$ , where  $p^* = (c_{10} - c_{00}) / (c_{10} - c_{00} + c_{01} - c_{11})$ .



## Dealing with Costs (cont'd)

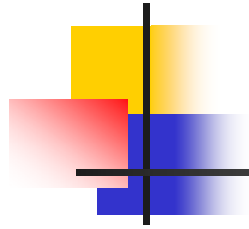
---

- Folk Theorem. By altering the example distribution, an error-minimizing classifier solves cost-sensitive problems.

- For binary output space, the number of negative examples is multiplied by

$$p^*(1 - p_0) / (1 - p^*)p_0$$

- Intuition: Changing the example distribution will change the posterior probability.



# Extend Classifiers with Voting

---

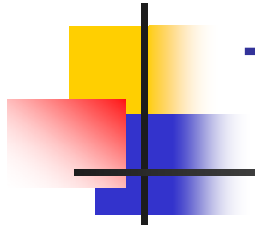
- We can increase the classifier's accuracy by learning several models and predict new cases by voting.
- Build four models for four different pairs of relevance/irrelevance ranks.



# Working Dataset

---

- ~180,000 categorized Reuters articles from 9/1996 – 7/1997
  - Relevant Categories: GVIO, G13, GDIP
  - Irrelevant Categories: 1POL, 2ECO, 3SPO, ECAT, G12, G131, GDEF, GPOL



# Test Methodology

---

- 10-Fold Cross-Validation on Reuters Dataset
  - 5 Trials
  - Approaches:
    - Naïve Bayes (NB)
    - Weight-based Complement Naïve Bayes (WNB)
    - OKAPI Direct Comparison (ODC)
    - OKAPI Logistic Regression (OLR)
    - OKAPI Cost-Sensitive LR (OCLR)



## ODC Tests with Varying N

N	Recall	Precision	Accuracy
5	0.926	0.868	0.935
10	0.931	0.875	0.939
25	0.931	0.874	0.939
50	0.931	0.868	0.937
Comp.	0.942	0.863	0.937

- Different N values do not significantly affect results



# Classifier Comparison Results

Classifier	Recall	Precision	Accuracy
NB	0.859	0.806	0.895
WNB	0.867	0.798	0.893
ODC	0.931	0.874	0.939
OLR	0.888	0.914	0.941
OCLR	0.929	0.875	0.939

- ODC and OLR:  $N = 25$
- OCLR:  $c_{00} = c_{11} = 0$ ,  $c_{01} = 1$ ,  $c_{10} = 0.7$

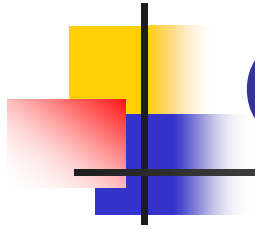


# Analysis

---

- All OKAPI classifiers performed better than NB and WNB in our tests.
- OLR has worse recall because it gives equal weights to false positives and false negatives.
- Adding cost-sensitivity improved performance.





# Conclusion

---

- The OKAPI classifiers are suitable for text classification.
- OLR doesn't perform as well as ODC.
- The cost table in OCLR can be adjusted to the appropriate trade-off necessary between recall and precision.