What Users Care about: A Framework for Social Content Alignment

Lei Hou[1], Juanzi Li[1], Xiaoli Li[2], Jiangfeng Qu[1], Xiaofei Guo[1], Ou Hui[1], Jie Tang[1]

[1] *Knowledge Engineering Group, Dept. of Computer Science and Technology, Tsinghua University*

[2] *Institute for Infocomm Research, A*STAR, Singapore*

Outline

- Motivation & Challenges
- Related Work
- Approach
- Experiment
- Conclusion & Future Work

Motivation

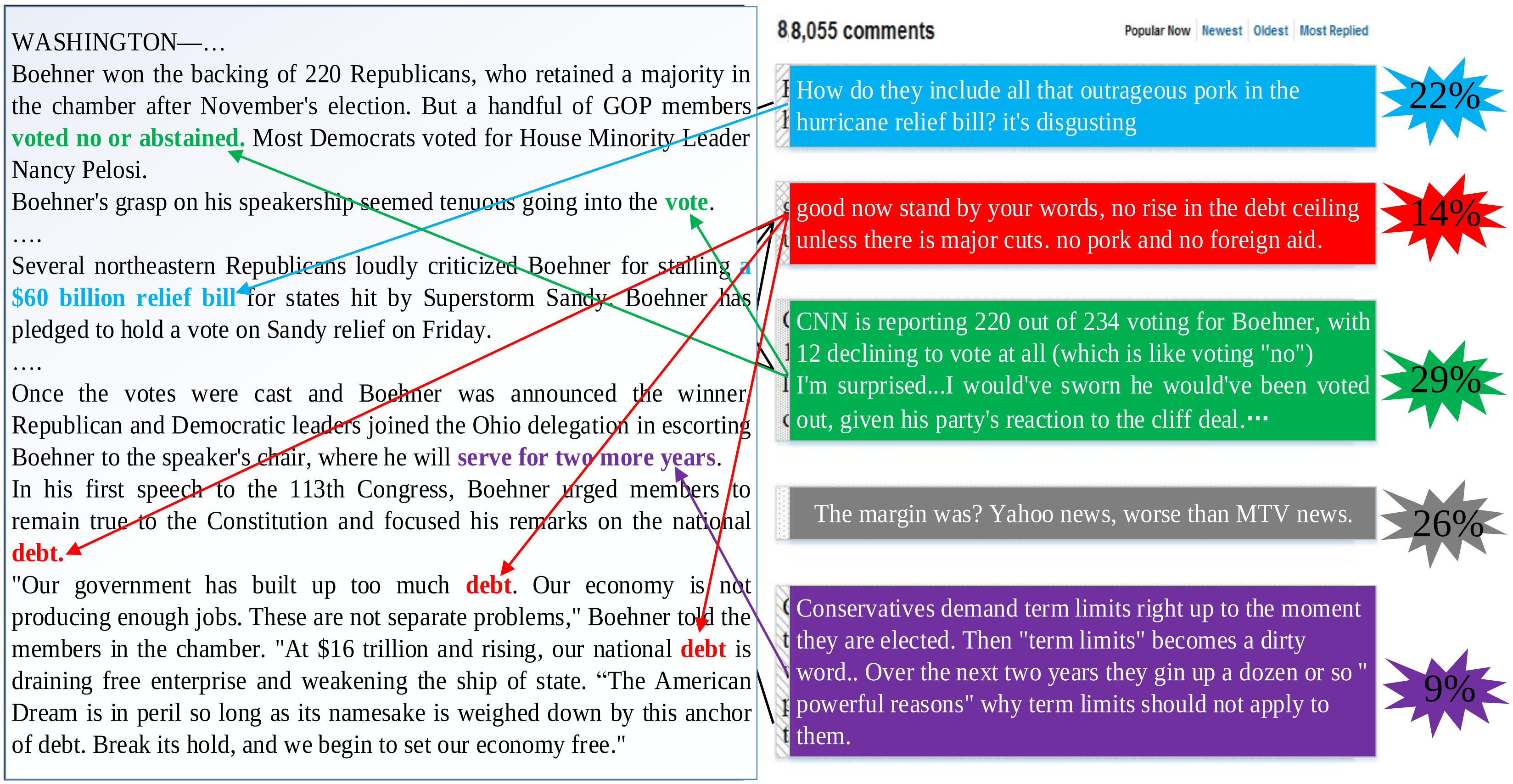
78% of Internet users in China (461 million) read news online[Jun, 2013, CNNIC]

The average numbers of comments for top news in Yahoo! and Sina are 5684.6 and 9205.4 respectively (on Nov, 2012)

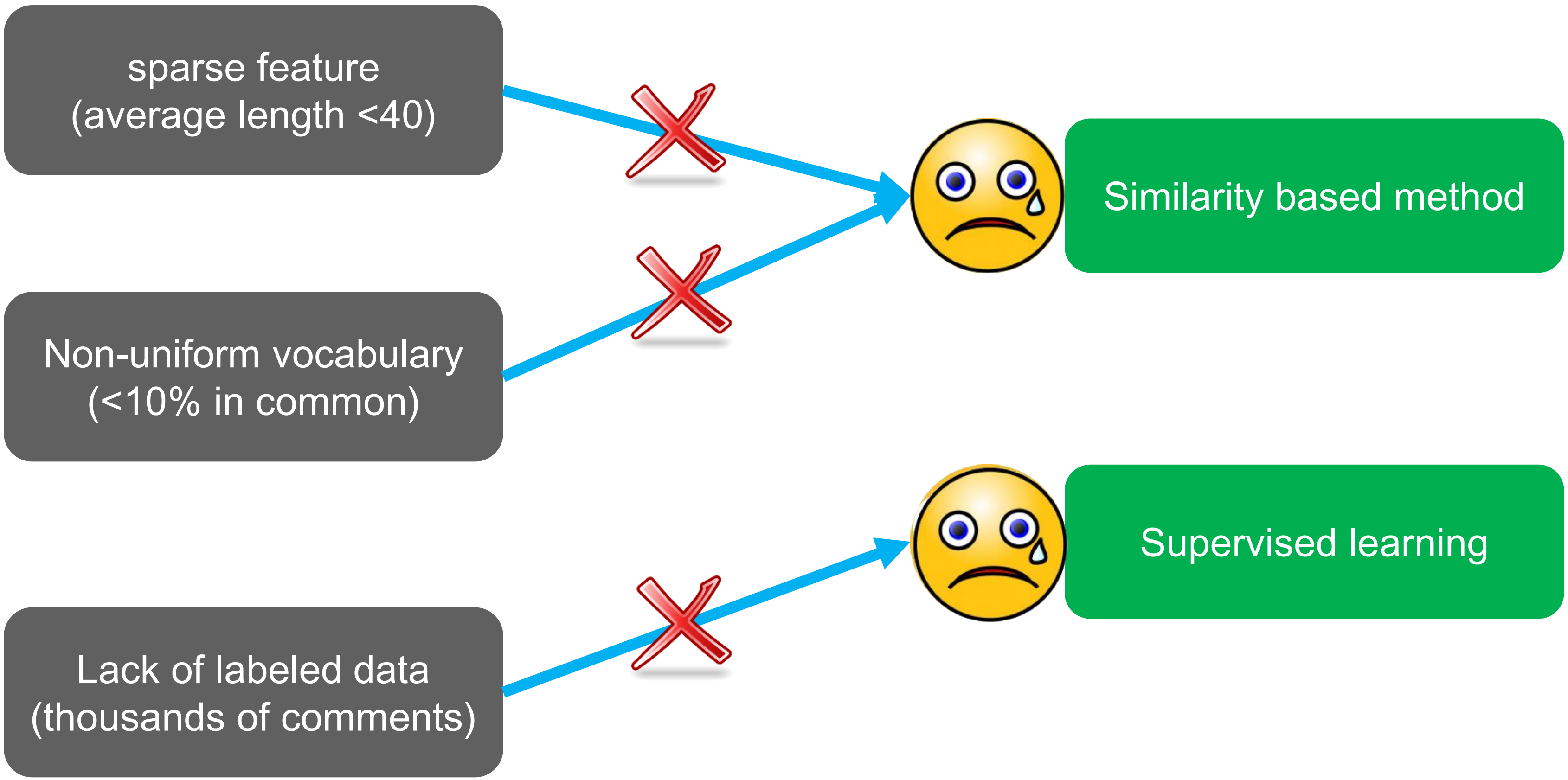
How to find what the users care about

Motivation

- How to achieve that?
  - Link sentences and comments ←→Social Content Alignment
- How to align?



Challenges


sparse feature
(average length <40)
Non-uniform vocabulary
(<10% in common)
Lack of labeled data
(thousands of comments)
Similarity based method
Supervised learning

Related Work-social content analysis

- Readalong: reading articles and comments together.
  - Dyut Kumar Sil, Srinivasan H. Sengamedu,and Chiranjib Bhattacharyya.
  - In WWW’11(poster)
- Supervised matching of comments with news article segments.
  - Dyut Kumar Sil, Srinivasan H. Sengamedu,and Chiranjib Bhattacharyya.
  - In CIKM’11(short papar)
- Opinion integration through semi-supervised topic modeling.
  - Yue Lu and Chengxiang Zhai.
  - In WWW’08

Related Work-topic modeling

- A time-dependent topic model for multiple text streams.
  - Liangjie Hong, Byron Dom, Siva Gurumurthy, and Kostas Tsioutsiouliklis.
  - In KDD’11
- Multi-topic based query-oriented summarization.
  - Jie Tang, Limin Yao, and Dewei Chen
  - In SDM’09
- Cross-domain collaboration recommendation.
  - Jie Tang, Sen Wu, Jimeng Sun, and Hang Su.
  - In KDD’12,

Related Work-positive unlabeled learning

- Building text classifiers using positive and unlabeled examples.
  - Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu.
  - In ICDM’03
- Learning with positive and unlabeled examples using weighted logistic regression.
  - Wee Sun Lee and Bing Liu.
  - In ICML’03.
- Learning to classify texts using positive and unlabeled data.
  - Xiaoli Li and Bing Liu.
  - In IJCAI’03.
- Learning to identify unexpected instances in the test set.
  - Xiaoli Li, Bing Liu, and See-Kiong Ng.
  - In IJCAI’07.

Approach Framework

PHASE 1

Document Comment Topic Model

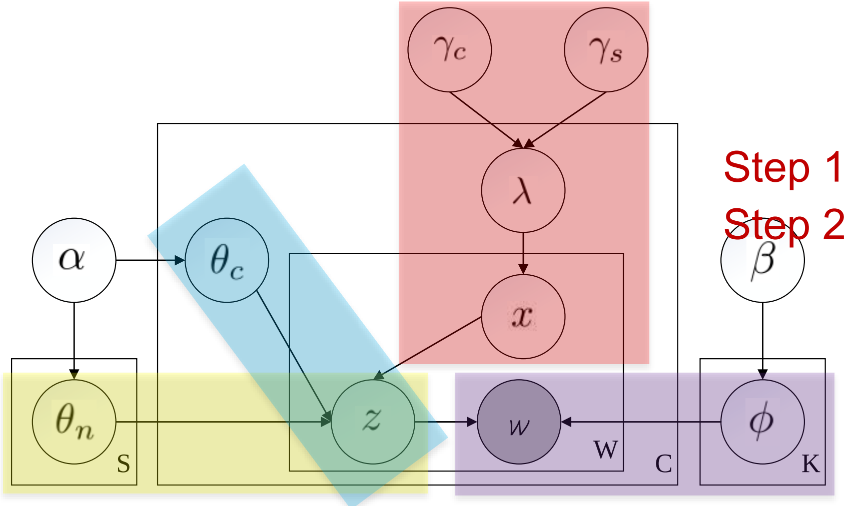
- Different vocabulary
- Sparse feature
- Dependency

PHASE 2

Learning from Positive and Unlabeled Data

- Unbalanced volume
- Lack of labeled data

Document-Comment Topic Model



Step 1:

Step 2:

Algorithm 1: Generative process for DCT model.

Input: the priors α, β, γ_c, γ_s; S and C

Output: estimated parameters θ_s, θ_c, λ and ϕ

Initialize a standard LDA model over S;

foreach *comment* $c \in C$ **do**

  foreach *word* $w_{ci} \in c$ **do**

    Toss a coin x_{ci} according to $bernoulli(x_{ci}) \sim beta(\gamma_s, \gamma_c)$, where $beta(.)$ is a beta distribution, and γ_c and γ_s are two parameters;

    if $x_{ci} = 0$ **then**

      Draw a topic $z_{ci} \sim multi(\theta_c)$ from a comment-specific topic mixture;

    else

      Draw a topic $z_{ci} \sim multi(\theta_s)$ from a document-related topic mixture;

    end

    Draw a word $w_{ci} \sim multi(\phi_{z_{ci}})$ from z_{ci}-specific word distribution;

  end

end

Top words for topic *launch cost*

Aid	Korea
Stomach	Money
America	Launch
Food	America
Korea	Food

Comment only

News only

Both

The left only uses comments, and the right takes news as background

PU Learning

Algorithm 2: PU learning

Input: news sentences S, social contents C, topic distribution θ, word distribution ϕ
Output: A set of classifiers
for *each topic* **do**

1. Extract the positive and unlabeled example set;
2. Build first classifier:
   - calculate centroid and radius to construct a hyper-sphere
   - extract potential positive examples and negative examples
   - build first classifier using *Ricchio*
3. Build final classifier:
   - classify unlabeled data using first classifier
   - build final classifier using WSVM

end

s & c \ topic	vote	relief	…	debt
S_1	**0.173**	0.039	…	0.094
S_2	0.082	**0.127**	…	0.077
…				
S_M	**0.184**	0.083	…	0.105
C_1	…	…	…	…
C_2	…	…	…	…
…				
C_N	…	…	…	…

Positive example for topic *vote*

1. But a handful of GOP members voted no or abstained.
2. Boehner's ... seemed tenuous going into the vote.
3. Once the votes were cast and ... .

…

PU Learning

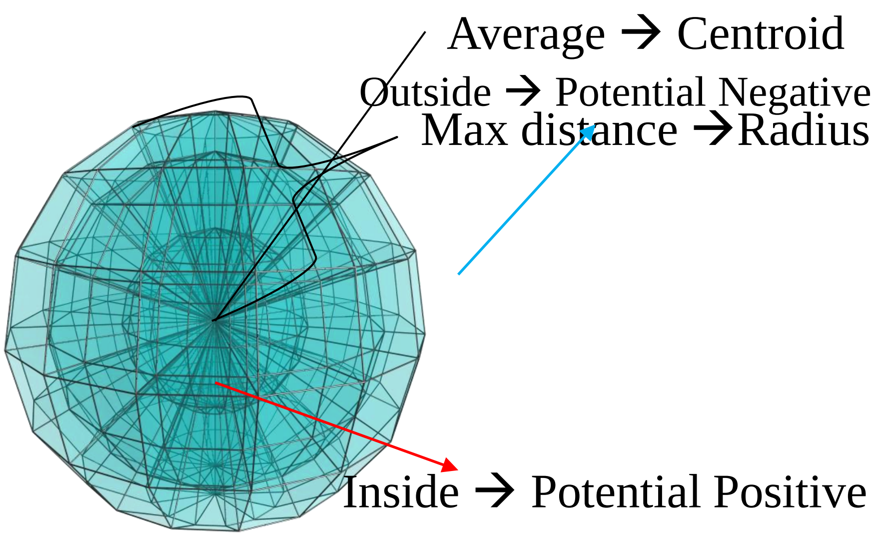
Algorithm 2: PU learning

Input: news sentences S, social contents C, topic distribution θ, word distribution ϕ
Output: A set of classifiers
for *each topic* **do**

1. Extract the positive and unlabeled example set;
2. Build first classifier:
   - calculate centroid and radius to construct a hyper-sphere
   - extract potential positive examples and negative examples
   - build first classifier using *Ricchio*
3. Build final classifier:
   - classify unlabeled data using first classifier
   - build final classifier using WSVM

end

	f_1	f_2	…	f_K
P_1	0.043	0.019	…	0.024
P_2	0.052	0.037	…	0.017
…				
$P_{\|P\|}$	0.054	0.033	…	0.015



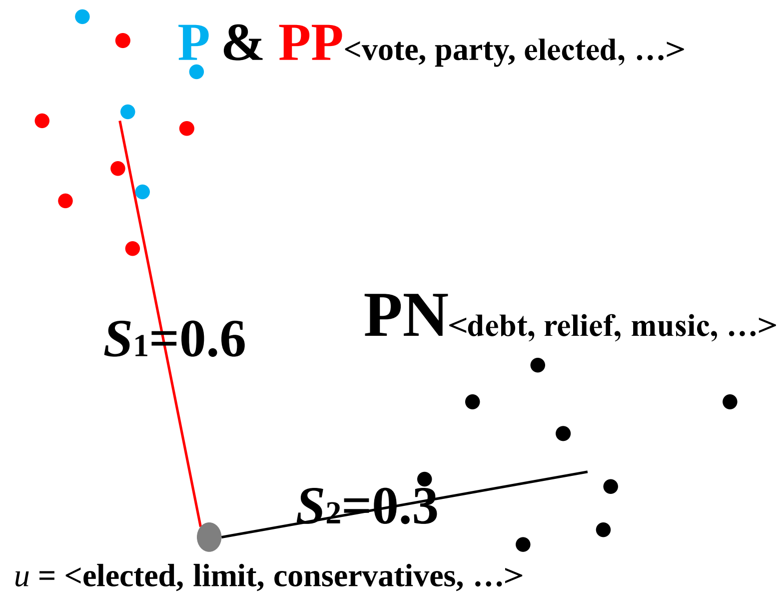
PU Learning

Algorithm 2: PU learning

Input: news sentences S, social contents C, topic distribution θ, word distribution ϕ
Output: A set of classifiers
for *each topic* **do**

1. Extract the positive and unlabeled example set;
2. Build first classifier:
    - calculate centroid and radius to construct a hyper-sphere
    - extract potential positive examples and negative examples
    - build first classifier using *Ricchio*
3. Build final classifier:
    - classify unlabeled data using first classifier
    - build final classifier using WSVM

end



Adjust the label according to **S**$_1$ and **S**$_2$, as well as assign a confidence score

$$L = \frac{\max(s_1, s_2)}{s_1 + s_2}$$

PU Learning

Algorithm 2: PU learning

Input: news sentences S, social contents C, topic distribution θ, word distribution ϕ
Output: A set of classifiers
for *each topic* **do**

1. Extract the positive and unlabeled example set;
2. Build first classifier:
   - calculate centroid and radius to construct a hyper-sphere
   - extract potential positive examples and negative examples
   - build first classifier using *Ricchio*
3. Build final classifier:
   - classify unlabeled data using first classifier
   - build final classifier using WSVM

end

	L	*f₁*	*f₂*	…	*f_K*
P_1	1	0.043	0.019	…	0.024
P_2	1	0.052	0.037	…	0.017
…					
LP_1	0.7	0.054	0.033	…	0.015
…					
LN_1	0.83	0.003	0.061	…	0.055
…					

$$Minimize: \frac{1}{2}\mathbf{w}^T\mathbf{w} + C_P\sum_{i\in P}\xi_i + C_{LP}\sum_{j\in LP}\xi_j + C_{LN}\sum_{k\in LN}\xi_k$$

$$subject\ to: y_i(\mathbf{w}^T\vec{x}_i + b) \geq 1 - \xi_i,\ i = 1, 2, ..., n$$

Data Set

- Sources (Chinese: Sina, English: Yahoo!)
- 22 news articles (10 Chinese, 12 English)
- 950 news sentences (516 in Chinese, 434 in English)
- 6,219 comments (4,069 in Chinese, 2,150 in English)

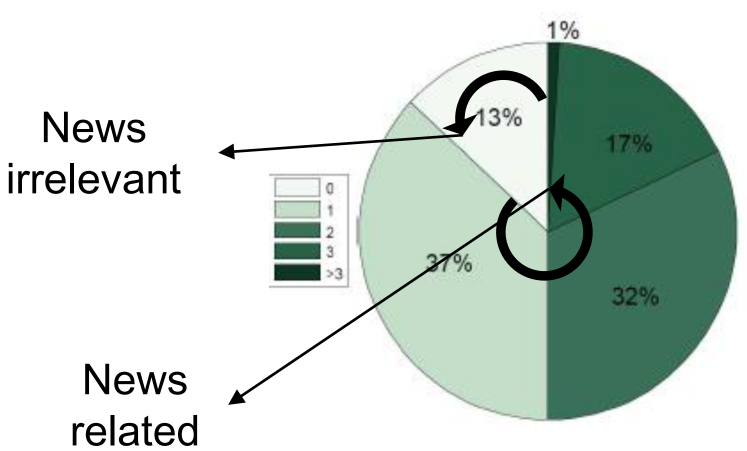
Table 1: Statistics on datasets

Source		#Sen/Com	Words	Vocabulary
Sina	*Sen*	516	8,932	2,772
	Com	4,069	112,853	13,891
Yahoo!	*Sen*	434	5,767	2,679
	Com	2,150	39,917	9,972

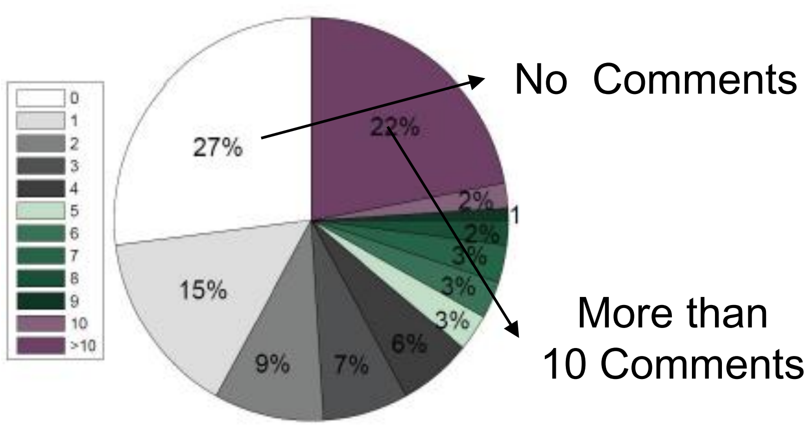
Annotation

- Manually Annotation
  - 7 annotators (publish task online)
  - Confidence: 5 out of 7 agree
  - Results: 7,520 (cn) + 2,327 (en) links
- Annotated Data Observation

Comment-News Sentences



News Sentences-Comment



Baseline Methods & Metric

- Methods
  - unsupervised
    - **VSM:** tf-idf + cosine similarity
    - **DCT:** topic directly
  - supervised
    - **BSVM:** classifier on sentence
    - **T-SVM:** classifier on topic
  - Ours(T-PU): unsupervised classifier on topic
- Metric

$$Precision = \frac{|\bigcup_{i=1}^{N}\{c_i | r_i \cap \tilde{r}_i \neq \emptyset\}|}{|C|}$$

where r_i and $\tilde{r}_i$ stands for the annotated alignments and the alignments that found by our method

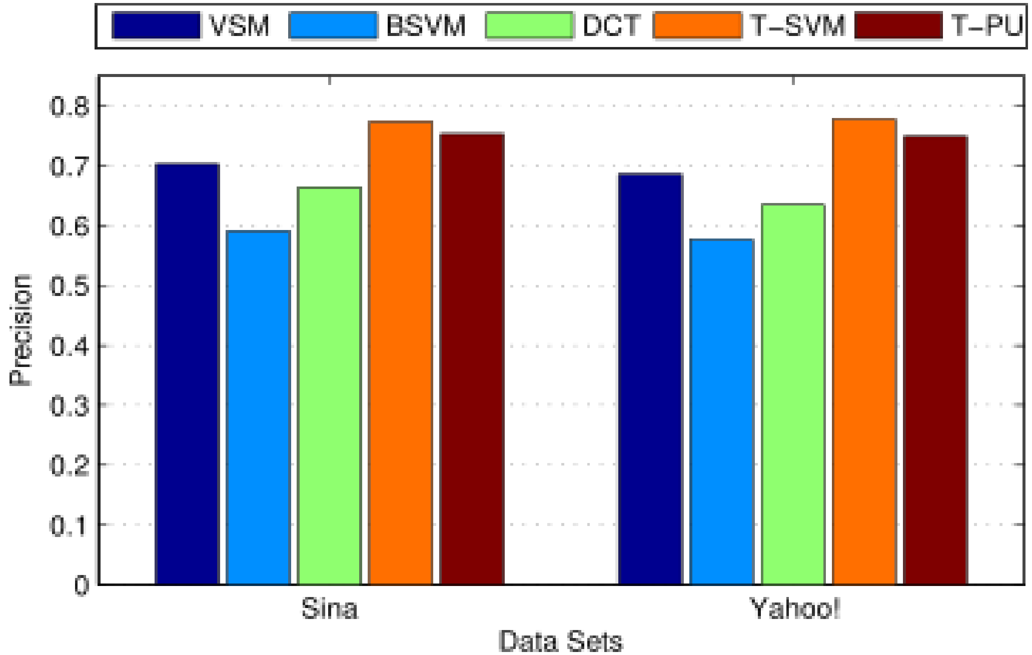
Results

- Overall

Table 2: Overall results on two datasets

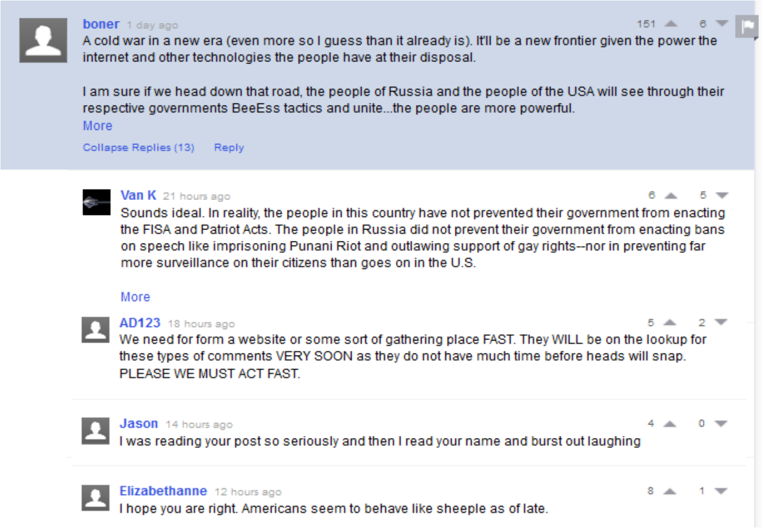
	Precision	Recall	F1-Measure
Sina	75.3%	56.7%	64.7%
Yahoo!	74.9%	63.4%	68.7%

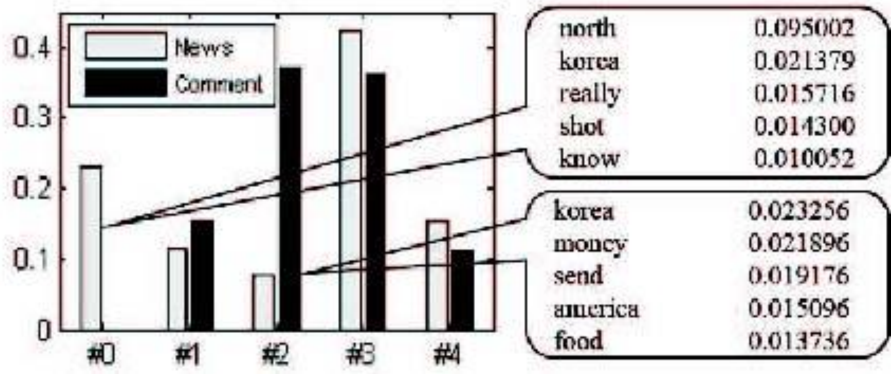
- Comparison
  - best among unsupervised methods (VSM **+7.9%**)
  - BSVM (**+25.9%**), significant improvement
  - T-SVM, comparable results (**-2.1%** in Sina and **-2.9%** in Yahoo!)



Results

- What leads to failed alignment
  - comment chain (a series of comments issued by two or more users while discussion)
  - topic drift
- Example:





Conclusion

- Study the social content alignment problem and present a two-phase framework to address it
- Propose DCT model which exploits Web document, social content and their dependency
- Employ PU learning algorithm for alignment
- Experimental results show the effectiveness of the proposed approach

Future Work

- Alignment over similar web documents
- Whether the social relationships influence the alignment
- Topic drift in the social content

Thanks!