Weighted Multi-label Classification Model for Sentiment Analysis of Online News

Li, X., Xie, H., Rao, Y., Chen, Y., Liu, X., Huang, H. and Wang, F.

Sun Yat-sen University

Caritas Institute of Higher Education

Sentiment analysis

- Given a piece of document, the system automatically detects its sentimental polarity
  - positive / negative (Amazon and Zhihu, a most famous question answering websites in China)
  - happiness, anger, empathy, warmness and touching (Sina news, the most popular news portal in China)

Sentiment analysis

- Target data
  - Movie review (IMDb)
  - Customer review (Amazon)
  - Short texts in social network (Twitter, Sina Weibo)
  - Articles on news portals (Sina news)
  - Texts on CQA websites (Quora, Stack Overflow, Zhihu)

Basic idea of SA

- Word level
  - Building emotion lexicon
- Sentence / Document level
  - Model the semantic representation of sentence / document based on word embedding

- Our work is to use word-emotion relatedness to infer the sentiment of the given text
- This work aims to detect the sentiment in the news or news headlines

Challenges

- It's hard to learn the discriminative features directly from the text
- Short texts suffer the data sparsity problem
- Noisy training documents will greatly affect the performance

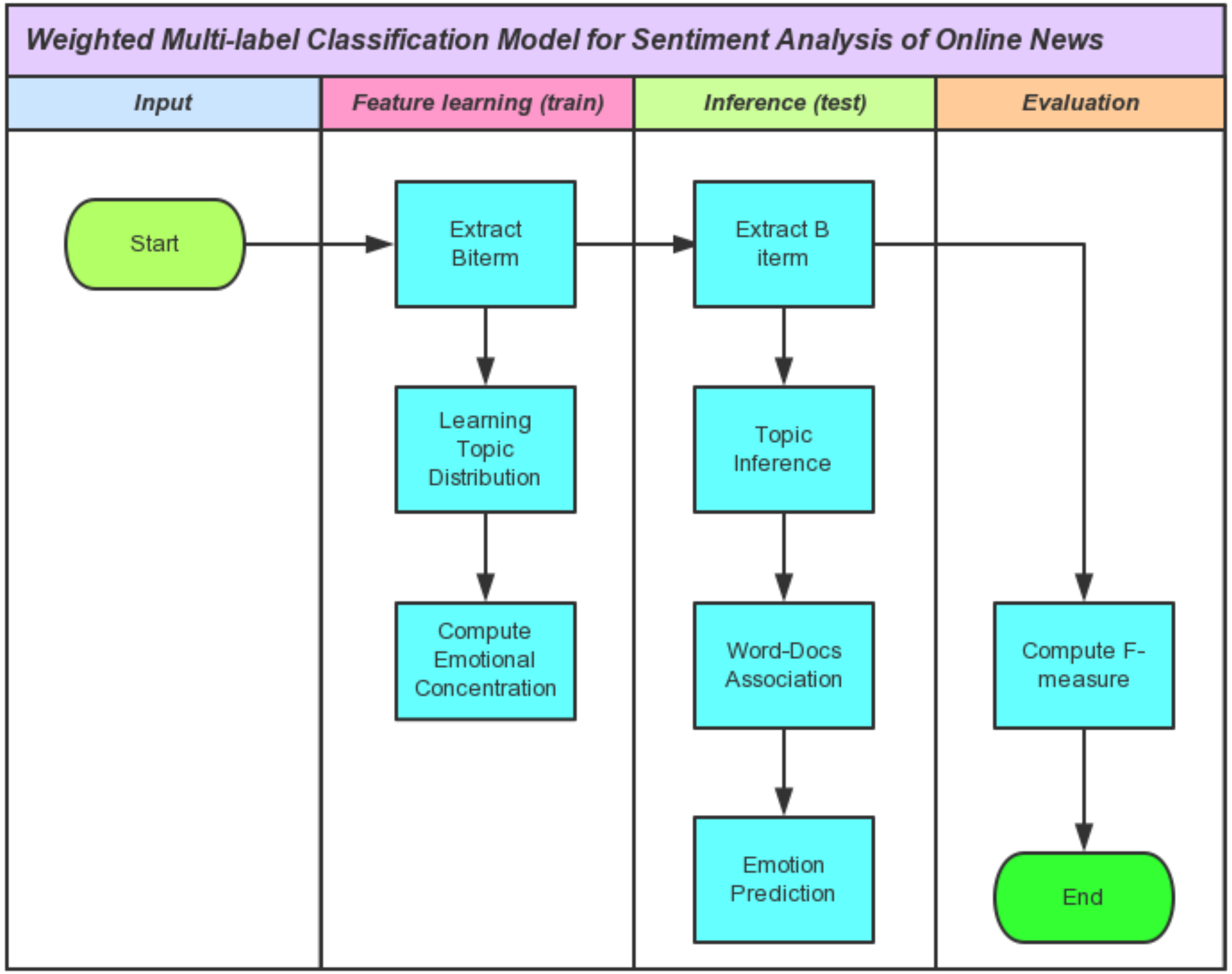
Our solution

- Deploy Biterm Topic Model (BTM) to generate more semantic features without external knowledge
- Propose “emotion concentration” to reduce the impact of “non-emotional” documents

Idea of our work

- Document with uniform / nearly uniform distribution of emotion votes is confused since words in it are not discriminative for the sentiment classifier
- Numerous generated word-word biterms can provide rich contextual information for feature extractor
- The composition pattern within a window can help us discover some hidden semantics

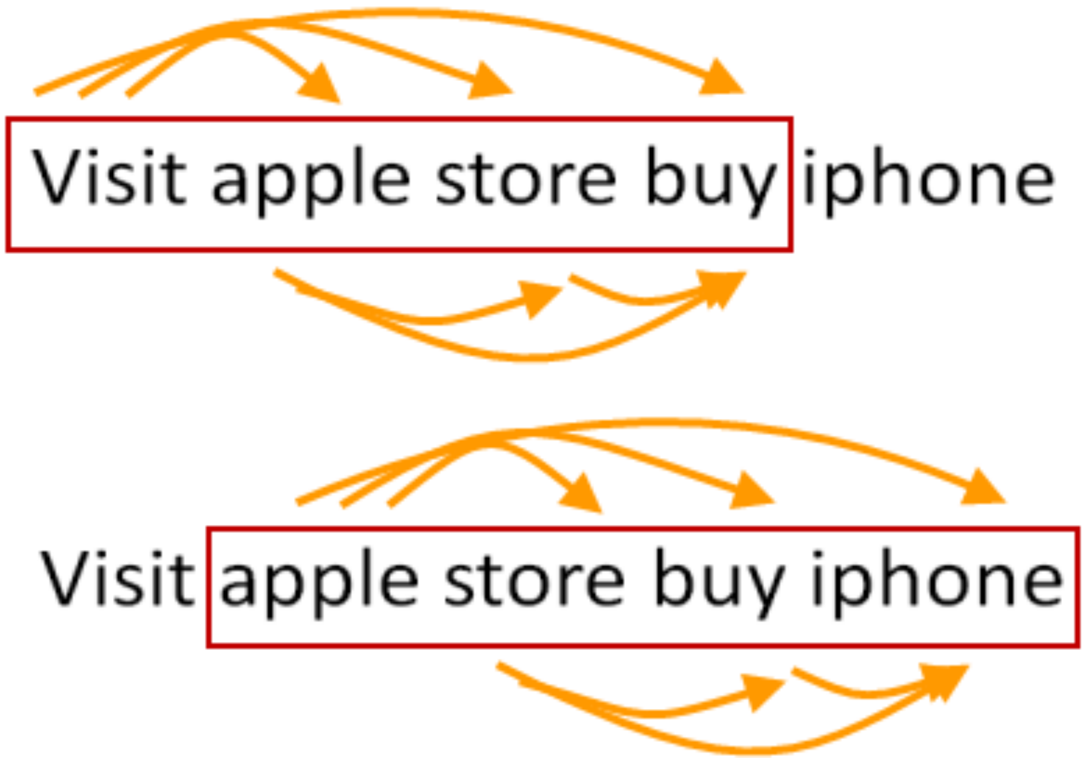
System pipeline



Biterm generation

- Biterm is an unordered word-pair within the text span
- The unified length of text span (i.e., window size) will be specified before the experiment
- Combine the words in the same window
  - For example, given the text “I will visit apple store and buy an iphone”
  - After filtering the stop words “visit apple store buy iphone”
  - Assume that *win* = 4, then we can get <visit, apple>, <visit, store>, <visit, buy>, <apple, store>, <apple, buy>, <apple, iphone>, <store, buy>, <store, iphone>, <buy, iphone>…

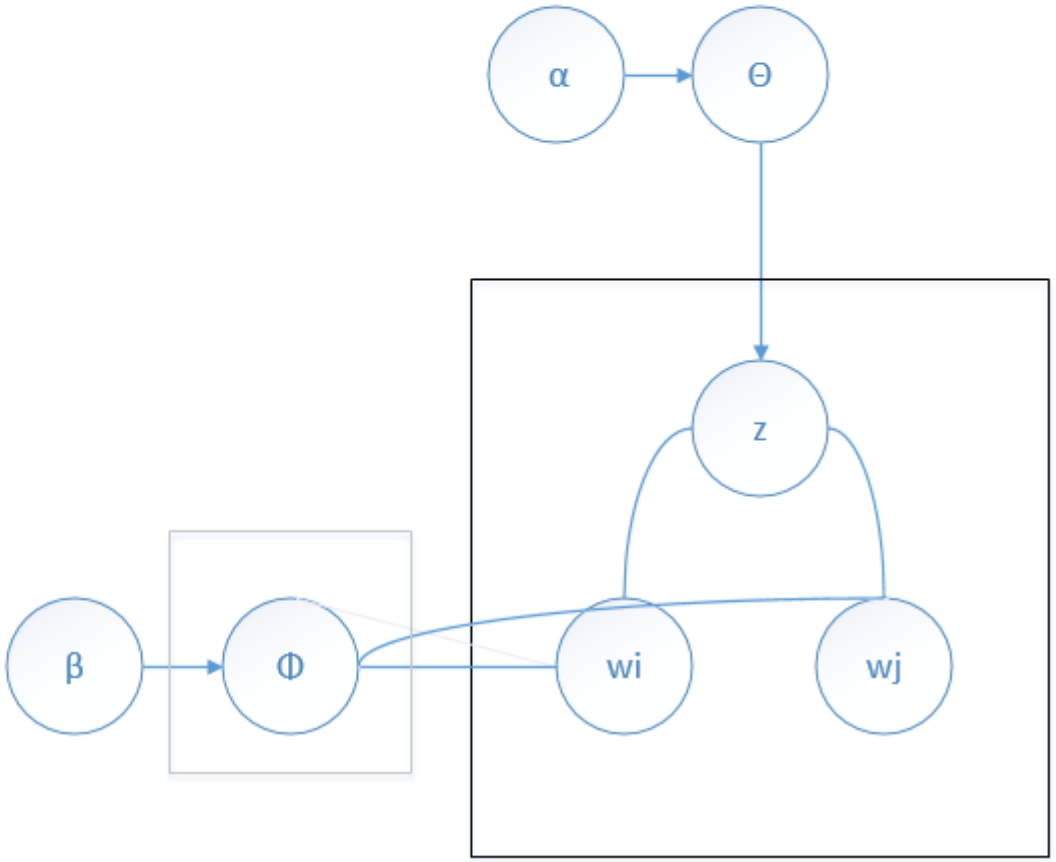
Biterm generation



Topic distillation

- Draw a topic distribution $\theta \sim Dir(\alpha)$ for the whole collection
- For each topic $k \in [1, K]$
  - Sample a word distribution $\varphi_k \sim Dir(\beta)$
- For each biterm
  - (1) draw a topic assignment $z \sim Mult(\theta)$
  - (2) draw two words $w_i, w_j \sim Mult(z)$

Topic distillation



Estimate document weight

- Tuition 1: a news document with uniform or nearly uniform rating distribution will decrease the overall performance
  - E.g.: a news article's vote is {"happy": 5, "anger": 5, "sympathy": 5, "fear": 5}
  - It is confused for user to "detect" the sentimental information, let alone the machine
- Tuition 2: news articles are the product of writer's subjective feeling, bias may exist. User votes of articles will reflect sentiment tendency more objectively.

Estimate document weight

- Based on these tuitions, we propose "emotional concentration" to weight the document
  - Given the ordered non-decreasing vote list $\{v_1, v_2, \ldots, v_E\}$
  - v is the arithmetic mean of the above values
  - Compute emotional concentration

$$F_i = i/E, \quad Q_i = \sum_{j=1}^{i} v_j / E * v$$

$$E\_con = \sum_{i=1}^{E-1} (F_i - Q_i) / \sum_{i=1}^{E-1} F_i$$

Analysis

- Weight of document *d* equals 0 for minimum concentration when $F\downarrow i - Q\downarrow i = 0 (i=1,2,...,E)$
  - In this case, $v_1 = v_2 = ... = v_E = \bar{v}$,the emotion perplexity reaches the peak according to the user vote, which means it is useless in the process of inference
- Weight of document *d* equals 1 for maximum concentration when $F\downarrow i - Q\downarrow i = F\downarrow i \ (i=1,2,...,E)$
  - In this case, $v_1 = v_2 = ... = v_{E-1} = 0$, all users vote for the *E* label, that is, this is a discriminative document for sentiment classifier
  - Many informative words can be found in the document

Sentiment classification

- Run BTM again to infer the document-topic distribution and topic-word distribution

$$P(z=k|d)=N(d,k)+\alpha/N(d)+K*\alpha\ ,P(w=t|z)=N(z,t)+\beta/N(z)$$
$$+W*\beta$$

  - *N(d, k)* is the number of word in *d* that assigned to topic *k*
  - *N(z, t)* is the number of word instances that assigned to topic *k*
  - Hyperparameters α ,β are used to avoid zero probability
- Associate word with document based on emotional concentration
- Predict the emotional label

Sentiment classification

- Topic-based emotion association between word and document
  - Given a document with 100 words and a word with 100 instances in the whole corpus
  - Assume that all of the documents are modeled by 4 topics 1, 2, 3, 4. There are 10, 20, 30, 40 instances of w assigned to topics 1, 2, 3 and 4 respectively and 10, 20, 30, 40 words in document are assigned to topics 1, 2, 3 and 4 respectively. If we instantiate topic 4 with "marriage", then
    - The document is mainly talking about "marriage", which will no-doubt invoke positive response
    - The word occurs most frequently in those documents make reader feel happy (marriage)

Sentiment classification

- Topic-based emotion association between word and document
  - Topic is a "bridge" to link a word in testing doc and a training doc
  - From the BTM, we can derive the doc-topic distribution P(z|d) and word-topic distribution P(z|w):

$$P(z|d)=N(d,z)+\alpha/N(d)+K*\alpha\ ,P(z|w)=N(w,z)+\alpha/N(w)+K*\alpha$$

  - *N(w)* is the number of instances of *w* and *N(w, z)* is the number of *w* instances assigned to topic *z*

Sentiment classification

- Topic-based emotion association between word and document
  - Then we compute their cosine-similarity to measure the relatedness between word and document

$$P(w,d) \approx \cos ine < P(z|d), P(z|w) >$$

Sentiment classification

- Computation of word-emotion relatedness
  - Given a word *w* and an emotion *E* followed with a collection of documents $\{d_1, d_2, \ldots, d_n\}$
  - Prior probability of a word w in training doc given the emotion label *E*

    $P(w|e) = \sum_{i=1}^{n} P(w, d_i) * E\_con_i$

  - *n* is the number of training document in emotion class *E*

Sentiment classification

- Bayesian-like inference is used to compute the score
  - The document will be classified into emotion class with maximum posterior probability
  - Posterior probability:

$$P(e|d)=P(e)*\prod_{i=1}^{n_d} P(e|w)$$

  - *P(e)* is the prior emotion distribution derived from the training data

Experiment

- Dataset
- Settings of BTM
- Baselines
- Evaluation
- Performance

Experiment

- Dataset
  - *SemEval (*short documents*)*. An English dataset in 2007 SemEval-14 task, which includes 1246 pieces of news headlines. First 1000 document samples are used to train the classifier and the rest is testing dataset
  - *SinaNews (*long documents*)*. A Chinese corpora contains 4570 news articles. Particularly, the first 2228 documents are regarded as training documents and the rest is used to evaluate the performance

Experiment

- Dataset

Dataset	Emotion label	# of articles	# of ratings
SemEval	anger	87	12042
	disgust	42	7,634
	fear	194	20,306
	joy	441	23,613
	sad	265	24,039
	surprise	217	21,495
SinaNews	touching	749	41,798
	empathy	225	23,230
	boredom	273	21,995
	anger	2,048	138,167
	amusement	715	43,712
	sadness	355	37,162
	surprise	167	11,386
	warmness	38	7,986

Experiment

- Settings of Topic model
  - For the BTM
    - SemEval: α=50/K, β=0.01, winsize=2
    - SinaNews: α=50/K, β=0.01, winsize=15
  - For the other topic-based methods
    - SemEval: α=0.05, β=0.01
    - SinaNews: α=50/K, β=0.01
  - The parameters of topic model are set empirically

Experiment

- Baselines
  - *SWAT*. One of the top-performing systems on the 2007 SemEval-07 task, which uses the unigram model to annotate the emotional content of news
  - *ET.* ET (Emotion term) ET followsthe naive Bayes (NB) method by assuming words are

    independently generated from emotional labels in two sampling steps
  - *ETM*. ETM (Emotion topic model) introduces an additional layer and utilizes the emotional distribution to reasonably guide the topic generation
  - *MSTM & SLTM*. Both MSTM (Multi-labeled supervised topic mode) and SLTM (Sen-timent latent topic model) will conduct two Gibbs sampling. MSTM begins by generating topics from words and then sampled emotions from each topic. SLTM, on the other hand, generated topics directly from user emotions
  - *RPWM* (Reader perspective weighted model) Entropy and LDA were used to estimate the weight of documents and then associate word with emotions

Experiment

- Evaluation

$$Pred_d@n = \begin{cases} 1, & e_p \in Top_d@n \\ 0, & otherwise \end{cases}, \quad MicroF1 = \frac{\sum_{d \in D_{test}} Pred_d@1}{|D_{test}|}$$

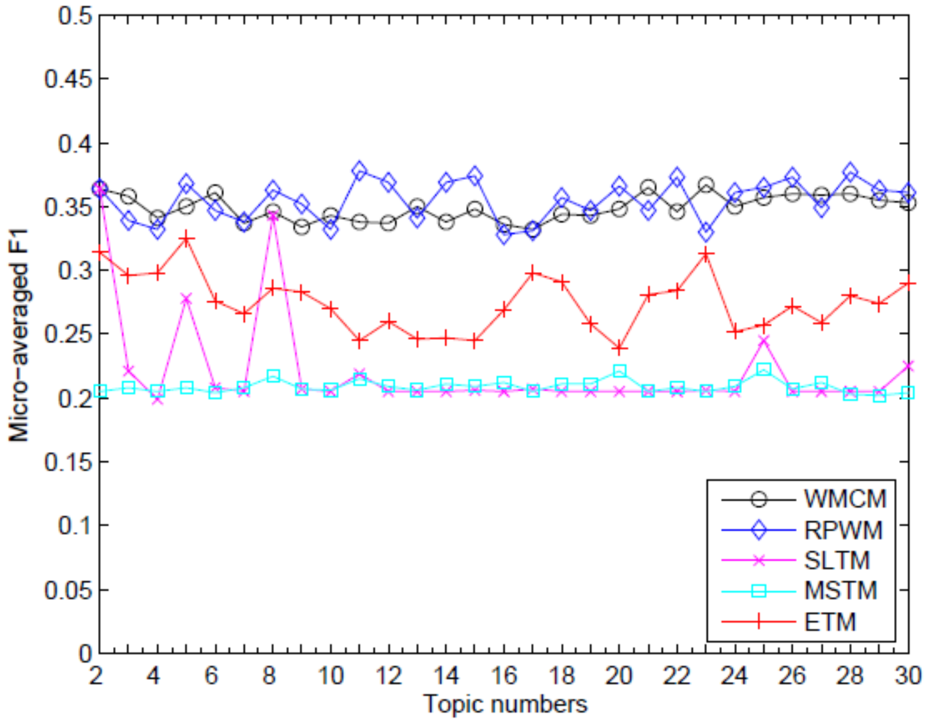
  - Particularly, e_p is the output label of classifier and $Top_d@n$ is the set of top-ranked emotions on the count of user votes
  - The left equation means that if the predicted label occurs in $Top_d@n$, then the prediction is accepted
  - The right equation means that in our experiment, only “best-match” is a right prediction

Experiment

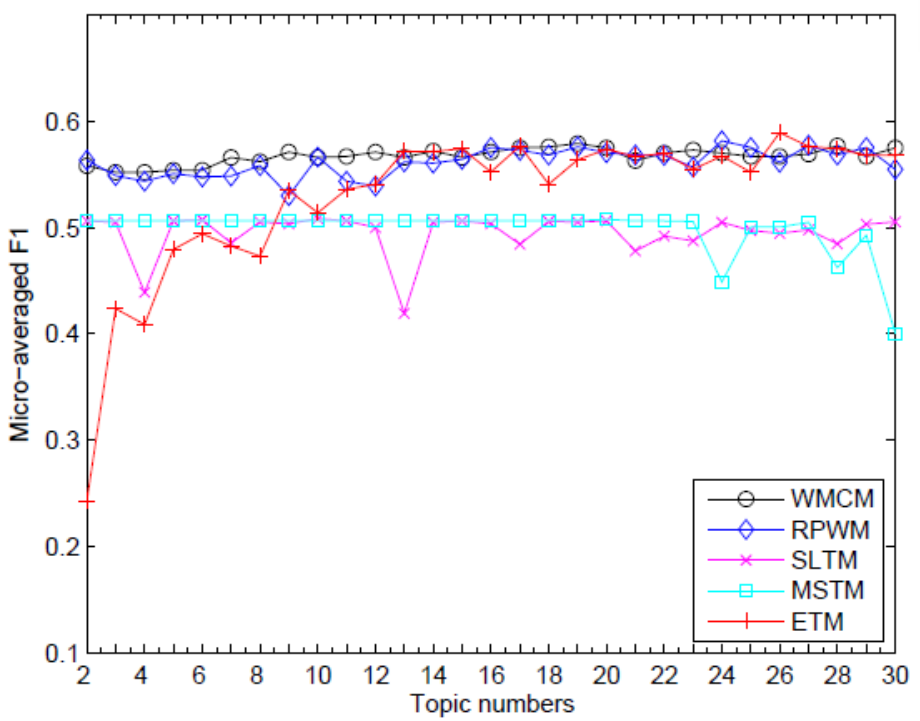
- Evaluation
  - Conduct F-test on WMCM and the baselines
  - Conduct t-test on WMCM and the baselines

Experiment

- Performance
  - Micro-averaged F1



(a) SemEval



(b) SinaNews

Experiment

- Performance
  - p-values of F-test on WMCM and baselines

Models	$SemEval$	$SinaNews$
RPWM	0.0105	0.0023
SLTM	8.4E-11	5.2E-7
MSTM	0.0001	3.0E-8
ETM	3.2E-5	0.0000

  - The result of F-test shows that performance of our WMCM is significantly more stable than those of baseline method since all of the p values are less than 0.05

Experiment

- Performance
  - p values of t-test on WMCM and baselines

Models	*SemEval*	*SinaNews*
RPWM	0.0928	0.0363
SLTM	2.3E-17	2.7E-19
MSTM	1.8E-43	7.5E-17
ETM	8.0E-19	0.0080
ET	1.7E-18	2.8E-37
SWAT	2.9E-17	2.1E-27

  - The results indicate the proposed WMCM outperformed the baselines of SLTM, MSTM, ETM, ET, SWAT significantly since the corresponding p values are less than 0.05
  - We also observed that WMCM's performance on SemEval was worse than RPWM's but the difference is not significant statistically (p-value=0.0928<0.05). Also, the former is better than the latter on SinaNews significantly (p-value=0.0363<0.05)

Thank you!