Learning to Align Comments to News Topics

LEI HOU, Tsinghua University
JUANZI LI, Tsinghua University
XIAO-LI LI, Institute for Infocomm Research, A*STAR
JIE TANG, Tsinghua University
XIAOFEI GUO, Tsinghua University

With the rapid proliferation of social media, increasingly more people express their opinions and reviews (user generated content; UGC) on recent news articles through various online services, such as news portals, forums, discussion groups, and microblogs. Clearly, identifying hot topics that users greatly care about can improve readers' news browsing experience and facilitate research into interaction analysis between news and UGC. Furthermore, it is of great benefit to public opinion monitoring and management for both industry and government agencies. However, it is extremely time consuming, if not impossible, to manually examine the large amount of available social content. In this paper, we formally define the news comment alignment problem and propose a novel framework that: 1) automatically extracts topics from a given news article and its associated comments, 2) identifies and extends positive examples with different degrees of confidence using three methods (i.e., hypersphere, density and cluster chain), and 3) completes the alignment between news sentences and comments through a weighted-SVM classifier. Extensive experiments show that our proposed framework significantly outperforms state-of-the-art methods.

CCS Concepts: • **Information systems** → **Data mining**; • **Theory of computation** → *Unsupervised learning and clustering*; • **Computing methodologies** → *Classification and regression trees*; Natural language processing; • **Applied computing** → Document management and text processing;

Additional Key Words and Phrases: User-Generated Content, Alignment, Dependent Topic Model, PU Learning, Density, Cluster Chain

ACM Reference format:
Lei Hou, Juanzi Li, Xiao-Li Li, Jie Tang, and Xiaofei Guo. 2017. Learning to Align Comments to News Topics. *ACM Transactions on Information Systems* 0, 0, Article 0 (2017), 31 pages.
DOI: 0000001.0000001

1 INTRODUCTION

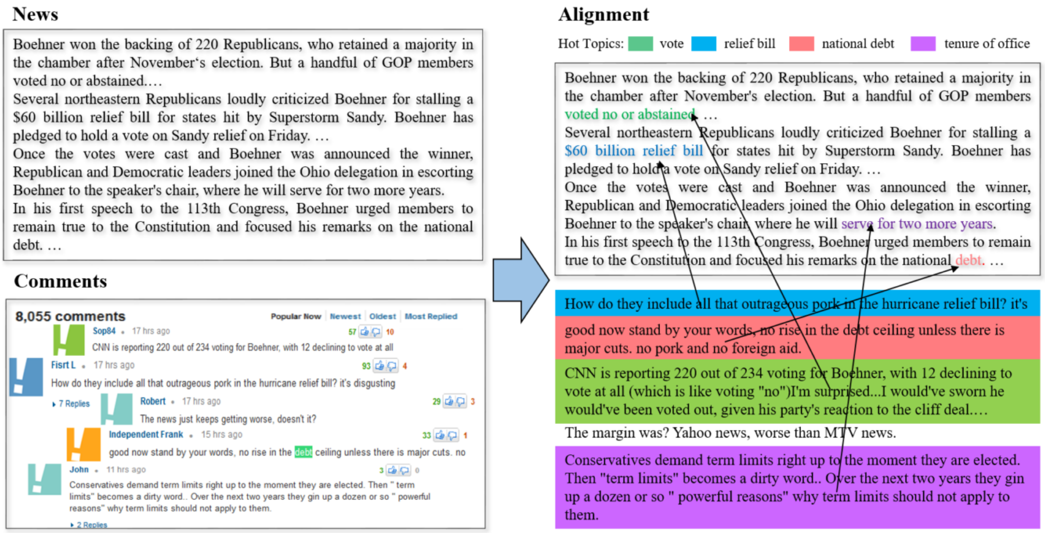
Social applications, such as Facebook and Twitter, have become indispensable tools for sharing information and personal comments on breaking news, political events, movies, products, etc. Report from *Pew Research Center* said that 63% of social users from Twitter and Facebook accessed

Lei Hou, Juanzi Li, and Jie Tang are supported by NSFC Key Program (No. 61533018), National Basic Research Program of China (973 Program, No. 2014CB340504), Fund of Online Education Research Center, Ministry of Education (No. 2016ZD102), NSFC-NRF (No. 61661146007) and Key Technologies Research and Development Program of China (No. 2014BAK04B03).

Author's addresses: Lei Hou, Juanzi Li (corresponding author, lijuanzi@tsinghua.edu.cn), Jie Tang and Xiaofei Guo, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, P.R.China; Xiao-Li Li, Institute for Infocomm Research, A*STAR, 1 Fusionopolis Way #21-01 Connexis Singapore 138632.

DOI: 0000001.0000001

news online, and roughly a quarter of them actively expressed their opinions on daily news through these social media applications [3]. Numerous news portals allow users to leave comments on various aspects/topics following the news reports. The resulting vast number of comments can clearly reflect the importance and popularity of different topics in the news. Understanding the correspondence between news and comments could benefit multiple media stakeholders, e.g., news readers could gain a quick overview of discussing topics and contribute their own opinions on those topics that they are particularly interested in; journalists could be aware of the news-triggered hot topics and further provide their responses/follow-up news reports to better serve the news consumers in a more focused and dedicated manner; media managers or PR departments of involved organizations could rapidly understand the topics that readers care about and further properly and intelligently respond the public opinions (e.g., let their advocates defend them, or use facts and figures, and cite third party sources) to achieve their objectives, such as effectively reduce the negative effect of bad news, controversy, and scandals.


News
Comments
Alignment
Hot Topics: 🟩 vote 🟦 relief bill 🟥 national debt 🟪 tenure of office
Popular Now | Newest | Oldest | Most Replied

Boehner won the backing of 220 Republicans, who retained a majority in the chamber after November's election. But a handful of GOP members voted no or abstained.…
Several northeastern Republicans loudly criticized Boehner for stalling a $60 billion relief bill for states hit by Superstorm Sandy. Boehner has pledged to hold a vote on Sandy relief on Friday. …
Once the votes were cast and Boehner was announced the winner, Republican and Democratic leaders joined the Ohio delegation in escorting Boehner to the speaker's chair, where he will serve for two more years.
In his first speech to the 113th Congress, Boehner urged members to remain true to the Constitution and focused his remarks on the national debt. …

8,055 comments

Sop84 • 17 hrs ago 57 10
CNN is reporting 220 out of 234 voting for Boehner, with 12 declining to vote at all

Fisrt L • 17 hrs ago 93 4
How do they include all that outrageous pork in the hurricane relief bill? it's disgusting

▸ 7 Replies

Robert • 17 hrs ago 29 3
The news just keeps getting worse, doesn't it?

Independent Frank • 15 hrs ago 33 1
good now stand by your words, no rise in the debt ceiling unless there is major cuts. no

John • 11 hrs ago 3 0
Conservatives demand term limits right up to the moment they are elected. Then " term limits" becomes a dirty word.. Over the next two years they gin up a dozen or so " powerful reasons" why term limits should not apply to them.

▸ 2 Replies

Boehner won the backing of 220 Republicans, who retained a majority in the chamber after November's election. But a handful of GOP members voted no or abstained. …
Several northeastern Republicans loudly criticized Boehner for stalling a $60 billion relief bill for states hit by Superstorm Sandy. Boehner has pledged to hold a vote on Sandy relief on Friday. …
Once the votes were cast and Boehner was announced the winner, Republican and Democratic leaders joined the Ohio delegation in escorting Boehner to the speaker's chair, where he will serve for two more years.
In his first speech to the 113th Congress, Boehner urged members to remain true to the Constitution and focused his remarks on the national debt. …

How do they include all that outrageous pork in the hurricane relief bill? it's

good now stand by your words, no rise in the debt ceiling unless there is major cuts. no pork and no foreign aid.

CNN is reporting 220 out of 234 voting for Boehner, with 12 declining to vote at all (which is like voting "no")I'm surprised...I would've sworn he would've been voted out, given his party's reaction to the cliff deal.…

The margin was? Yahoo news, worse than MTV news.

Conservatives demand term limits right up to the moment they are elected. Then "term limits" becomes a dirty word.. Over the next two years they gin up a dozen or so " powerful reasons" why term limits should not apply to them.

Fig. 1. An alignment example from Yahoo! News

Fig. 1 illustrates a news snippet about *Boehner*[1] from Yahoo! News, as well as several corresponding comments dedicated to five specific topics within the news such as *votes*, *relief bills*, *tenure of office*, and *national debt*. In this figure, we can see that the news attracted totally 8055 public comments and different users had different concerns about different topics. A popular news article often attracts large numbers of user discussions, e.g., [1] reported that both the Huffington Post and the Guardian received an estimated 25,000 – 50,000 comments per day and our preliminary statistics showed that, Sina[2] daily focused news received over 30,000 comments on average. In addition to the large volume of comments, most news websites present comments by threads – micro-conversations within the comments on an article. This form of comment organization could also waste readers' time because many threads could drift away from the initial news topics or

[1] http://news.yahoo.com/blogs/ticket/john-boehner-elected-speaker-house-190301689--politics.html
[2] The major Chinese news portal, website: http://news.sina.com.cn/hotnews/

even include totally irrelevant advertisements. Hence, it is a challenging task for users to manually read and digest the overwhelming volume of noisy comments.

In this article, we present a novel problem of *news comment alignment*, which attempts to address how to automatically align users' comments to news topics, thus alleviating the difficulties and challenges in large-volume comment browsing and understanding. Recently, numerous methods for processing user generated content have been proposed, including topic extraction in news and UGC [21, 52, 53, 55], news–UGC integration and summarization [34, 44, 45, 54]. However, existing topic extraction methods do not consider the fact that users post their comments based on news topics, and the integration and summarization research focuses on identifying common and private topics in news and UGC. Although we can adapt these methods to address the alignment problem, they are rather ineffective as they are not designed for our task. As such, to address the limitations, we propose a novel, two-phase framework, as shown in Fig. 2.

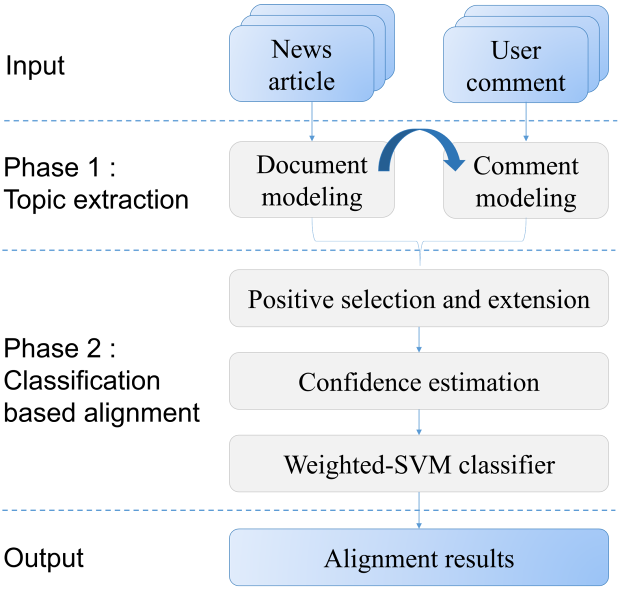


Fig. 2. Overview of the proposed approach.

In the first phase, we extract topics from both news and users' comments. A straightforward technique is to independently use a probabilistic topic model on news and comments and then align the extracted topics on both sides [45]. However, it is difficult to guarantee the consistency between two different topic sets from news and comments respectively. An alternative method is modeling news and comments together [21, 35, 55], but topic bias may occur due to the extremely unbalanced volumes (i.e., substantially more comments have been generated compared to news). In this study, we observe that comments heavily depend on the news content, but both lines of existing methods ignore this dependence. In this paper, we propose a novel document-comment topic model (DCTM) that can effectively leverage the dependence between news and comments using two correlated generative processes. In particular, it first models the news sentences using a standard topic model (i.e., Latent Dirichlet Allocation [6]). For the associated comments, it then employs a Bernoulli distribution to determine whether words in comments are generated from news-inherited or comment-specific topics.

In the second phase, intuitively, two types of machine learning methods [45], namely, unsupervised methods and supervised methods, can be employed for news comment alignment at the topic level. Unsupervised methods first generate weighted feature vectors for both news and comments (e.g., TF-IDF calculates the word statistics [25] and ESA leverages the document-related Wikipedia concepts to represent documents [16, 17]). Then, for each topic extracted in the first phase, the methods select the comments with the highest similarities as the aligned results. However, as comments are usually fragmented, informal, poorly structured and they can be written using quite different vocabularies compared with news, unsupervised methods thus cannot effectively match news sentences with associated comments. Supervised learning methods, on the other hand, require users to provide training data (i.e., the news sentences or comments are annotated with topics). Then, a classification model can be built to classify each comment into one of the topics. However, in many practical applications, labeling sufficient training examples to build accurate classifiers is time consuming. Thus, supervised learning methods are not suitable either.

In our previous work [22], a Positive Unlabeled learning method (PU learning) was proposed for building accurate classifiers. Given a news topic, sentences associated with the topic will be treated as positive data P, and all of the other news sentences and comments are treated as unlabeled data U. Due to their extremely unbalanced volumes (the size of U is far greater than that of P), we designed a positive extension step before building the final classifier. In particular, we employed a hypersphere to identify more potential positive data from U and further assigned them confidence scores. However, a deeper analysis (see Section 3.2.4) shows that the positive distribution may not be in the shape of a hypersphere. Therefore, we propose two novel positive extension strategies (density-based and cluster-chain-based algorithms) to address this issue. Correspondingly, the confidence estimation step (i.e., the Rocchio classifier) changes into its weighted version. The main contributions in this paper are summarized as follows:

- We formally define the news comment alignment problem and present a novel, two-phase framework to address it.
- We propose an innovative document comment topic model that can exploit news, comments, as well as the content dependence to facilitate an accurate topic extraction. We also present a theoretical derivation for the parameter estimation.
- We conduct comprehensive analysis on the benchmark datasets and observe that the comments within a topic do not distribute according to a regular hypersphere shape. Based on the observation, we propose density-based and cluster chain based positive extension methods, which can identify more accurate positive examples than the existing strategy [22] for building the final classifier.
- The experimental results show that our proposed framework can effectively address the challenging issues in news comment alignment and can significantly outperform state-of-the-art methods.

The rest of this paper is organized as follows. We first formulate the news comment alignment problem in Section 2. Then, we introduce the data collection process and observations in Section 3 and the proposed methods in Section 4. Next, the experimental results and detailed analysis are reported in Section 5. We review the related literatures in Section 6. Finally, we conclude the paper in Section 7.

2 PROBLEM DEFINITION

Although user generated content can be expressed in various forms, in this paper, we focus on the most commonly used *textual* information, namely, the users' comments dedicated to news posted through social media applications or Web 2.0 technologies.

Preliminary. A news article d consists of a sentence set $S = \{s_1, s_2, \ldots, s_M\}$ and is associated with a comment set $C = \{c_1, c_2, \ldots, c_N\}$, both of which cover several topics $T = \{t_1, t_2, \ldots, t_K\}$. Each sentence $s \in S$ belongs to a topic $t \in T$, while each comment $c \in C$ can have single or multiple associated topic(s). Both sentence and comment can be represented as a vector $\mathbf{w}_s$ (or $\mathbf{w}_c$) of N_s (or N_c) words, where each word w_{si} (or w_{ci}) in $\mathbf{w}_s$ (or $\mathbf{w}_c$) is chosen from a vocabulary of size W.

Definition 2.1 (News Comment Alignment). Given a news article with the sentence set S and the associated comments C, the goal of news comment alignment is to identify the common topics T, where each topic $t_j \in T$ is associated with several news sentences in S, and generate a set of matching pairs $\{(c_i, t_j)| \; c_i \in C, t_j \in T \cup \emptyset\}$. For a pair (c_i, t_j), $t_j \in T$ means comment c_i is talking about topic t_j and thus can be aligned to the associated sentences, and $t_j = \emptyset$ indicates that c_i is irrelevant to any news topics or concerns topics beyond the current news article.

As shown in Fig. 1, the left panel includes a news article entitled *John Boehner re-elected as speaker of the House* from Yahoo! News and the comments posted by users. On the right, we present the discovered topics (e.g. *votes*, *relief bills*, *tenure of office* and *national debt*) and the alignment with arrows linking comments to the representative sentences in news. Note that the 4th comment, without links, indicates that we cannot automatically find the proper alignment in the news. In the alignment process, topics act as bridges between news sentences and users' comments. Clearly, a news article often covers several topics. We observe that each sentence in given news article is typically associated with one topic as its author or reporter usually wants to express a specific topic and semantic to facilitate readers' understanding. A user comment, on the other hand, might have multiple associated topics because users often express their opinions on multiple topics in their comments after they read through the whole news. They are very flexible to provide their views on various topics in the news, without considering good readability. To perform the alignment, we propose a two-phase framework, namely, multi-source probabilistic topic modeling and positive unlabeled classification learning, which will be discussed in Section 4.

3 DATA AND OBSERVATION

Before introducing our proposed method, we first present our datasets and various observations in this section.

3.1 Data Collection

Since no benchmark datasets for news comment alignment have been made available, we generated in-house datasets so that we could evaluate different alignment approaches. Note that each dataset consists of a news article and corresponding comments posted by different users. To increase the diversity, we crawled news articles and their associated user comments from two popular news websites, namely, Sina (China) and Yahoo! (U.S.). We selected the most commented news articles from *Dec. 1st* to *Dec. 11th*, 2012, and obtained 10 Chinese datasets from Sina and 12 English datasets from Yahoo!.

Then, we recruited 7 annotators from the widely-used crowd-sourcing platform named Zhubajie[3] to manually build gold-standard links between the news sentences and users' comments. These annotators are all undergraduates majored in *Journalism and Communication*, *English* or *Computer Science*. Particularly, we first assigned numerical ID for sentences and comments, and then let the annotators organize their results as $\langle cID, sID, confidence\rangle$ tripes (see Table 1 for details). We required the annotators to process at least 400 comments for each news article. Normally, one

[3] http://task.zbj.com/2338979/

needs 3 – 4 hours to complete the alignment for each news article, and to facilitate the annotation, we encouraged the annotators to use some clues, such as named entity.

Table 1. Annotation Results Organization Form

Column Name	**Instructions**
cID	comment ID
sID	news sentence ID, fill -1 if they can not find a proper alignment
confidence	a value between [0,1], indicating their confidence over this link

After annotation, we obtained 13,316 different links for Chinese datasets and 4517 links for English datasets, and the average Kappa coefficient among annotators is 0.611, indicating the alignment is not a trivial task. Finally, we only kept those links on which the majority (i.e., 5 out of 7) of people agreed (i.e., confidence value ≥ 0.7). Interested readers can access the datasets from our website[4].

To facilitate the subsequent experiments, we performed pre-processing for both datasets, including removing stop words, and filtering low-frequency words (frequency ≤ 3). The statistics about our constructed benchmark datasets after pre-processing are summarized in Table 2, including the numbers of the sentences/comments, number of the words, vocabulary size and number of the final links.

Table 2. Dataset Statistics of our 22 benchmark datasets

Source		**#Documents**	**#Words**	**Vocabulary size**	**Final links**
Sina	*Sentence*	516	8,932	2,772	7,260
	Comment	4,069	112,853	13,891	
Yahoo!	*Sentence*	434	5,767	2,679	2,423
	Comment	2,150	39,917	9,972	

3.2 Observations

In this section, we study the interplay between news and comments at both the sentence level and the topic level using our datasets. Specifically, we want to address the following three key questions:

- *Comment to sentence/topic linking distribution* - how many sentences or topics does a particular comment cover?
- *Sentence/topic to comment linking distribution* - how many comments does a news sentence or topic attract?
- *Comment distribution within topic* - how do the comments within a specific topic distribute?

3.2.1 Comment to Sentence Linking Distribution. Fig. 3(a) shows the distribution of comments with respect to the number of linked news sentences. We observe that 87% of comments (36.5%+ 32.2%+ 17.4%+ 0.9%) are linked to 1 – 3 news sentence(s), while the remaining 13% are irrelevant to any news sentence, indicating that we can leverage comments to enhance topic detection, especially in our scenario where news articles are usually short.

[4] https://github.com/THU-KEG/ijcai13data

3.2.2 Sentence to Comment Linking Distribution. We then study the distribution from the *opposite* direction, and the results are also presented in Fig. 3(b) and Fig. 3(c).

- Fig. 3(b) shows that 22% of news sentences attract more than 10 comments. The statistic shows that it is important to automatically mine relevant comments from the large amount of comments, and it can significantly facilitate users to quickly and effectively understand what the public care about.
- We further investigate these 22% popular news sentences towards numbers of aligned comments. From Fig. 3(c), we observe that sentences approximately follow a *Power-law* distribution with respect to the number of related comments. This tells us that a small number of hotspot sentences in the news attract substantially more comments than other sentences.
- We also investigate those sentences (27%) without any related comments and find that these sentences just provide the background information for the news and thus lead to the “no comments" phenomenon.

3.2.3 Topic-Level Distribution. As defined in Section 2, topics serve as the bridge between news and comments. Therefore, we investigate the relationship between news topics and comments, and the results are also presented in Fig. 3. Specifically, Fig. 3(d) demonstrates the distribution of comments with respect to the number of related topics. In Fig. 3(e), we randomly select *five* news articles to see how many comments each topic can attract (note that topics are extracted by a standard LDA with the topic number set as 5). We observe the following results:

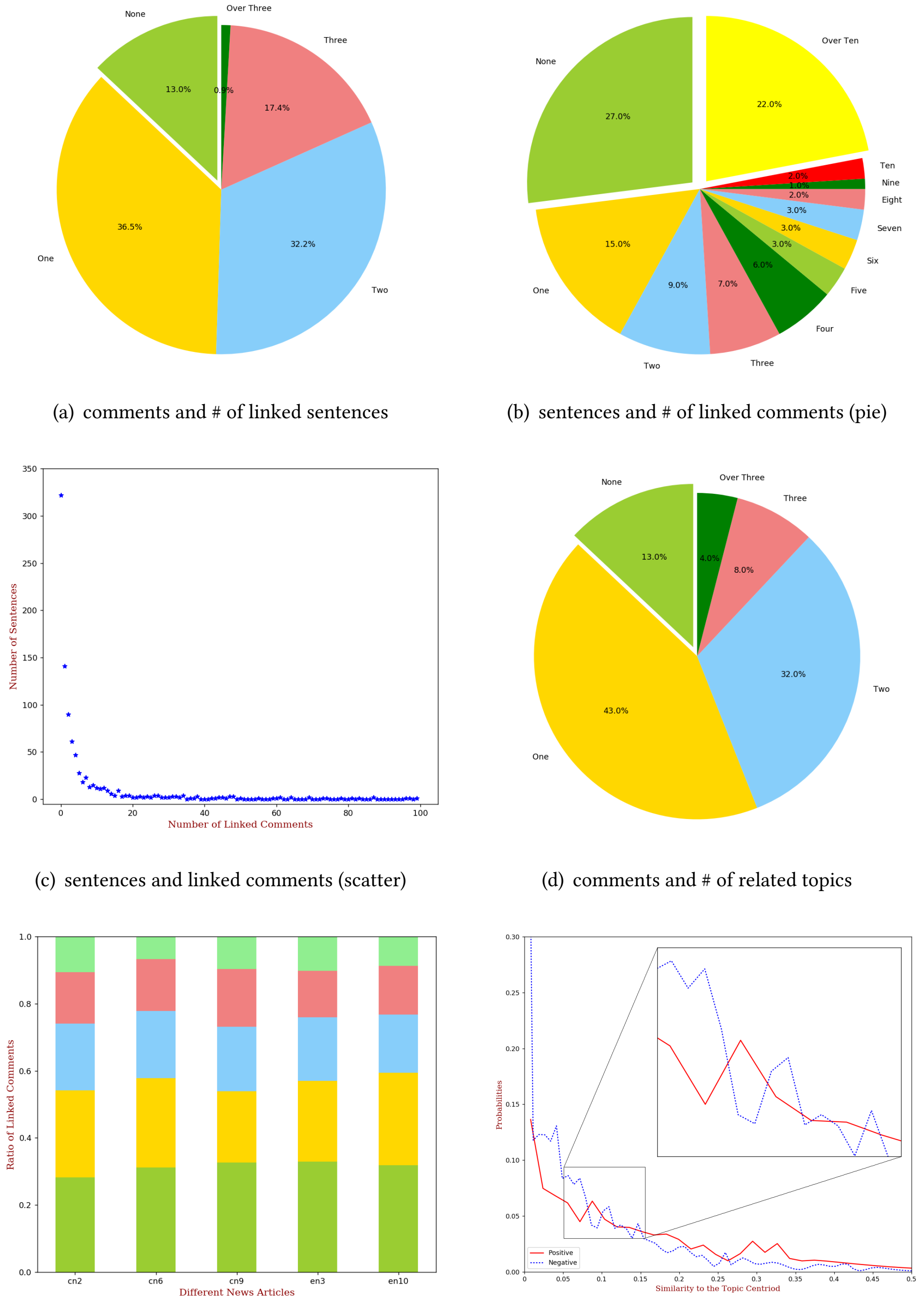
- Most comments, i.e., more than 75% (45% + 32%), belong to 1 or 2 topics. This observation is realistic because users' comments are typically very short and often focus on 1 or 2 aspects of the news. It can help us filter noisy results obtained by alignment algorithms (e.g., if a comment is classified into more than two topics).
- The top *4* topics for a given news article can cover more than 90% of the comments. This indicates that, generally speaking, few topics are sufficient for topic modeling and we can focus on those important topics for news comment alignment if the efficiency is a major concern.

3.2.4 Comment Distribution within Topics. Finally, we select the representative news sentences for each topic in the headline *Beijing is studying the university entrance exam policy for migrant students*[5]. Because we are attempting to identify the comments that belong to each topic, we plot the distribution of the annotated positive comments (link to news sentences within target topics) and negative comments according to their cosine similarities to the topic centroids (representing all of the representative news sentences) in Fig. 3(f). To calculate the cosine similarities, we use the top 500 words for each topic based on their LDA probabilities, and remove common words occurring across different topics to obtain the distinguishing vocabulary.

We observe that in high-similarity regions, the positive examples play a dominant role, which validates the hypersphere extension in our previous work [22] (i.e., we can expand those positives near the topic centroid). However, only a small number of true positives can be reached. On the other hand, if we relax the similarity constrain to a small value to include more positives, some negatives will also be included in the extension set. Therefore, we need to design effective methods to find additional true positives without introducing too many false positives.

Summarizing the statistics above, we come to the following conclusions:

[5] http://news.sina.com.cn/c/2012-12-03/035425713856.shtml



(a) comments and # of linked sentences

(b) sentences and # of linked comments (pie)

(c) sentences and linked comments (scatter)

(d) comments and # of related topics

(e) topics and attracted comments

(f) positives and negatives within topics

Fig. 3. Data Observations.

- The generation of users' comments relies heavily on news content (i.e., a dependence exists between news and comments), and comments posted by users are normally based on topics within the news they browse. Therefore, in this paper, we propose document comment topic model in Section 4.1, leveraging the content dependence to improve topic extraction, and verify the observation in Section 5.1.
- Although it is possible for comments to be linked to multiple topics, they generally only cover one or two topics. This gives us a hint that a reasonable constraint on the alignment or an additional pruning step is required to filter out poor-quality links. Therefore, for a given comment, we limit the maximum number of its aligned topics to 3 at the last alignment step in Section 4.2.4 as it is sufficient to link a comment to its relevant topics.
- Since topics/sentences follow *power-law* like distributions with respect to the number of linked comments, we do not have to process all topics/sentences because the major topics/sentences cover almost all of the comments. Note that we report the results on all topics to demonstrate the effectiveness of the proposed method in this paper because efficiency is not the major concern currently.
- The comments surrounding a specific topic follow a *power-law* like distribution, indicating that our previous positive extension method [22] misses many positive examples. Therefore, we design two additional extension methods in Section 4.2.2 based on the observation, i.e., *density extension* and *cluster chain extension*, and verify that the combination of three strategies obtains more pure positives.

4 METHOD

In this section, we describe our proposed framework based on the data observation. First, we discuss the topic extraction method by incorporating the dependence between news and comments. Then, we present our effort on positive and unlabeled classification learning. In particular, we build classifiers for linking comments to the appropriate topic(s) in four steps: positive selection, positive extension, confidence estimation, and weighted support vector machine.

4.1 Document-Comment Topic Model

To address the sparse and non-uniform feature problem when modeling comments, large-scale data collections, such as Wikipedia, can be used to enrich features [40], but it may introduce noise and often fails on newly generated content. Consider the following scenario: When a user reads news online, he/she may have opinions on some interested topics and thus may subsequently post comments on such topics. Although different words may be used in news and comments, they could be unified in the topic space. Therefore, extracting topic features is a better choice for solving the problem.

Through the observation, it is clear that comments heavily leverage the news articles (refer to Section 3.2.1: 87% of comments are linked to one or more news sentence(s)), and topics build a bridge between news and comments. Therefore, news behaves as a kind of background knowledge that provides guidance for the comment generation. However, all of the existing works pay little attention to the cross-media dependence. Therefore, we propose the document-comment topic model (DCTM) to simultaneously model news and corresponding comments.

The basic idea is to use two correlated generative processes to model news and comments. In the first process, we employ standard LDA to model the news sentences, and in the second process we take the previous results as background knowledge to guide comment modeling. In particular, for each word in the comments, we use a Bernoulli distribution to determine whether this word is generated from a news-inherited or comment-specific topic. Fig. 4 shows the graphical structure of the DCTM (for simplicity, we omit the modeling part for news and focus on the modeling of comments).

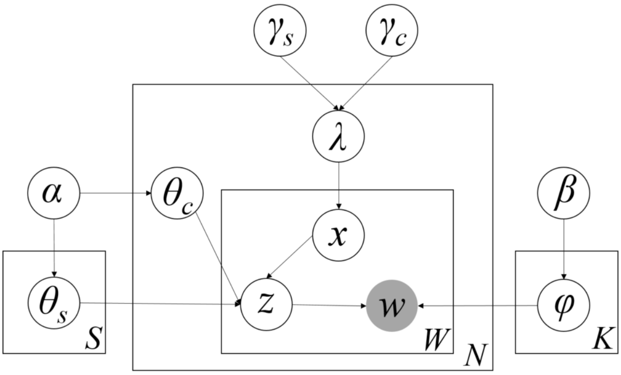


Fig. 4. Graphical representation of DCTM.

Let us briefly introduce our notations. θ_s and θ_c are topic models from documents and comments; x is a binary variable indicating whether the current word inherits the topic from document-related topics ($x = 1$) or comment-specific topics ($x = 0$); α and β are the Dirichlet hyper parameters; λ is a parameter for sampling the binary variable x; γ_c and γ_s are *beta* parameters used to generate λ. Table 3 summarizes the notations.

Table 3. Notations in Gibbs Sampling

Notations	**Description**
K, W, M, N	number of topics, vocabulary size, number of sentences and comments
w_{di}	the *i-th* word in document *d*
z_{di}	the topic assigned to word w_{di}
x_{di}	whether w_{di} is a word from a document-related topic ($x_{di} = 1$) or a comment-specific topic ($x_{di} = 0$)
θ_s, θ_c	multinomial distribution over topics
ϕ	multinomial distribution over words
α, β	the *Dirichlet* priors
λ	parameter for sampling the binary variable x
γ_c, γ_s	*Beta* parameters used to generate λ

Formally, the generative process is described in Algorithm 1. It extracts topics for document sentences according to the distribution $p(\theta_s|\alpha)$, and for word w_{ci} in comment c, a coin x is first tossed according to $p(x|c) \sim beta(\gamma_c, \gamma_s)$ to decide whether w_{ci} is sampled from a news-inherited or comment-specific topic. Then, the user would decide the topic z_{ci} to comment on according to $p(\theta_c|\alpha)$ and $p(\theta_s|\alpha)$. Finally, the word w_{ci} is sampled from $p(\phi_{z_{ci}}|\beta)$.

The model solution is to estimate the unknown parameters in the DCTM, which can be categorized into three sets: 1) the corresponding topic z_{si} for each word w_{si} in sentence s and the distribution

ALGORITHM 1: Generative process for DCTM.

```
Input: the priors $\alpha$, $\beta$, $\gamma_c$, $\gamma_s$; $S$ and $C$
Output: estimated parameters $\theta_s$, $\theta_c$, $\lambda$ and $\phi$
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 $\gamma_c$
          and $\gamma_s$ 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 news-inherited topic mixture;
        end
        Draw a word $w_{ci} \sim multi(\phi_{z_{ci}})$ from $z_{ci}$-specific word distribution;
    end
end
```

θ_s of M sentences; 2) the corresponding coin x_{ci} and z_{ci} for each word w_{ci} in comment c, the distribution θ_c of N comments, and the distribution λ of C comments; and 3) the distribution ϕ of K topics. Obviously, we can estimate parameters for news and comment separately or in a joint manner. Considering the subsequent classifiers are built on news topics to process the continuously produced comments, which requires the news topic model to be precise and stable, thus we model news and comments separately as shown in Algorithm 1.

Performing exact inference is typically an intractable problem, and a variety of algorithms have been proposed to conduct approximate inference, such as variational EM methods [6] and *Gibbs* sampling. In this paper, we take the widely-used *Gibbs* sampling technique for its ease of implementation. For the parameters in 1), we use the same sampling algorithm as for the LDA model (i.e., with the posterior probability):

$$p(z_{si} = k|\mathbf{z}_{\neg si}, \cdot) = \frac{n_{sk}^{\neg si} + \alpha}{\sum_z (n_{sz}^{\neg si} + \alpha)} \times \frac{n_{kw_{si}}^{\neg si} + \beta}{\sum_w (n_{kw}^{\neg si} + \beta)} \tag{1}$$

where n_{sk} is the number of times that topic $z = k$ has been sampled from the multinomial distribution specific to sentence s; n_{kw} is the number of times that a word w has been generated by topic $z = k$; and the superscript '$\neg$' indicates excluding the current instance from counting.

Then, we consider a two-step sampling for parameter estimation in 2). First, we sample the coin x according to the posterior probability as follows (detailed derivation can be found in Appendix [48]):

$$p(x_i = 0|\cdot) = \frac{n_{cx_0}^{\neg ci} + \gamma_c}{n_{cx_0}^{\neg ci} + n_{cx_1}^{\neg ci} + \gamma_c + \gamma_s} \times \frac{n_{z_{ci}}^{\neg ci} + \alpha}{\sum_z (n_z^{\neg ci} + \alpha)} \tag{2}$$

where n_{cx_0} is the number of times that $x = 0$ has been sampled in c, and $n_{z_{ci}}$ is the number of times that topic z has been sampled from c. $p(x_i = 1|\cdot)$ can be analogously defined as Equation 2.

Subsequently, the posterior probability of the assigned topic z_{ci} is defined as

$$p(z_{ci}|x_{ci} = 1, \mathbf{x}, \mathbf{z}_{\neg ci}, \cdot) = \frac{n_{z_{ci}w_{ci}}^{\neg ci} + m_{z_{ci}w_{ci}} + \beta}{\sum_w (n_{z_{ci}w}^{\neg ci} + m_{z_{ci}w} + \beta)} \times \frac{n_{cz_{ci}}^{\neg ci} + m_{cz_{ci}} + \alpha}{\sum_z (n_{cz}^{\neg ci} + m_{cz} + \alpha)} \tag{3}$$

where $n_{z_{ci}w_{ci}}^{\neg ci}$ and $m_{z_{ci}w_{ci}}$ denote the number of times that word w_{ci} has been generated by topic z_{ci} in comments and news, respectively. $n_{cz_{ci}}^{\neg ci}$ and $m_{cz_{ci}}$ are the number of times that topic z_{ci}

has been sampled from a comment-specific or news-inherited topic distribution. Note that the only difference between separate and joint parameter estimation is whether the $m_{z_{ci}w_{ci}}$ and $m_{cz_{ci}}$ change during comment modeling, i.e., these values are determined after sentence modeling in the separate estimation, while they might change during the iteration in the joint setting.

During the sampling process, the algorithm keeps track of $(M + N) \times K$ (sentence+comment by topic), $N \times 2$ (comment by coin), $N \times 2 \times K$ (comment by coin by topic), and $K \times W$ (topic by word) count matrix, based on which we can easily estimate the three above-mentioned sets of parameters (i.e., θ_s, θ_c, λ and ϕ).

As for the hyperparameters α, β, γ_s, and γ_c, we can estimate the optimal values using a *Gibbs* EM algorithm [2] or a variational EM method. For certain applications, topic models are sensitive to the hyperparameters and it is thus necessary to obtain the proper values for the hyperparameters. In the applications discussed in this work, we find that the estimated topic models are not very sensitive to the hyperparameters. Therefore, we take fixed values for simplicity, as explained later in the experimental section.

After topic modeling, both news sentences and comments are associated with probability distributions over extracted topics (i.e., θ_s and θ_c), according to which we can select representative sentences for each extracted topic.

4.2 Learning from Positive and Unlabeled Data

Once we complete the news topic extraction, a straightforward idea for comment alignment is, inferring the topic assignments for input comments and then linking them to corresponding news sentences with the same topic. However,

- generative models (e.g., our proposed DCTM) learn the joint distribution of the data and they picture how data is generated (e.g., DCTM tries to simulate the user's commenting behavior). Directly using the model result is not the best choice in some discriminative tasks, as is the case with our alignment task (see the experiment results in Section 5.1).
- On the other hand, discriminative models learn conditional distribution to create decision boundaries between different classes of data, but it often requires neatly selected features.

Previous studies [47] show, leveraging the advantages of both models, i.e., employing the features learned by generative models into discriminative models, could achieve a better performance. Therefore, we design a positive and unlabeled learning method to align comments with the news topics identified in previous section.

Particularly, we present the detailed process in four steps, namely, positive selection, positive extension, confidence estimation and use of a weighted support vector machine classifier.

4.2.1 Positive Selection. Without loss of generality, we take a particular topic t_j as an example to introduce our proposed method. Once topic t_j is specified, the sentences in S belonging to t_j (i.e., with the maximal probability to topic t_j instead of other topics) constitute the positive set P_j (in the following sections, we omit the subscript, i.e., P, for simplicity). All comments in C will be treated as an unlabeled set U, as shown in Equation 4. Our PU learning model attempts to build an accurate classifier for topic t_j to classify all of the comments in U into either a positive or negative class – comments that are classified into the positive class will thus be aligned to the topic t_j.

$$\begin{aligned} P &= \{s_i | \arg\max_{1 \le k \le K} \theta_{s_i k} = j,\ i = 1, 2, \ldots, M\} \\ U &= C \end{aligned} \tag{4}$$

4.2.2 *Positive Extension.*

As a news article typically contains a small number of sentences in S, given a topic t_j, its positive set P ($P \subseteq S$ only covers those news sentences related to t_j) is usually very small (the number of related sentences ranges from 3 to 11 in our data statistics). Directly building a classifier on extremely unbalanced examples (i.e., few positives and hundreds of negatives) performs badly. So our initial PU learning step is to expand P by including those potential positives from unlabeled set U because U contains *hidden* positive comments about t_j.

In particular, we propose a method that integrates three different strategies to expand P, namely, *hypersphere* extension, *density* extension, and *cluster chain* extension. Moreover, we will assign those expanded positive examples higher (or lower) confidence scores if they are extended by multiple (or individual) extension methods.

Hypersphere Extension. This strategy was used in our previous work [22], where we partition the unlabeled set U into a potential positive set PP and a potential negative set PN by constructing a hypersphere classifier:

$$\begin{aligned} &\mathbf{o} = \frac{\sum_{d \in P} \mathbf{d}}{|P|} \quad r = \frac{\sum_{d \in P} dist(\mathbf{d}, \mathbf{o})}{|P|} \\ &PP = \{u | u \in U \ and \ dist(\mathbf{u}, \mathbf{o}) \leq r\} \\ &PN = U - PP \end{aligned} \tag{5}$$

where $\mathbf{d}$ is the feature vector of document d (either sentence or comment), and the features in $\mathbf{d}$ consist of two parts, namely the topic distributions from previous step and the standard TF-IDF representations (the TF-IDF values are normalized before splicing); $\mathbf{o}$ and r denote the centroid and radius of the hypersphere. Comments that fall into the hypersphere are treated as potential positives, and the others are treated as potential negatives.

However, as observed in Section 3.2.4, the comments surrounding particular topic approximately follow a *power-law* distribution, i.e., there exist negative examples that are much closer to the centroid $\mathbf{o}$ than positive examples, which means that the potential positive set PP contain negatives (noise or false positives). Since hypersphere is a simple regular shape, it cannot better balance the extension quantity and quality through parameter adjustment.

Density Extension. Our density-based extension algorithm expands P by iteratively including P's *densely connected examples*. This is because, although examples from positive class P could be heterogeneously distributed in the space (forming different shapes/clusters), these densely connected samples are very similar to each other intra-class and thus tend to share the same topic label. Actually, each dense space is a small hypersphere, and it follows the observation in Section 3.2.4 (i.e., the positive examples play a dominant role in the high-similarity regions). We now review some useful concepts in density-based clustering [15].

Definition 4.1 (ε Neighborhood). Given an input dataset D and an example $e \in D$, the ε Neighborhood of e, denoted by $N_\varepsilon(e)$, is defined by $N_\varepsilon(e) = |\{q | dist(e,\ q) \leq \varepsilon, q \in D\}|$, where ε can be viewed as the neighborhood radius.

Definition 4.2 (Directly Density Reachable). Given a radius ε and an integer J_ε, an example e' is directly density reachable from e if $e' \in N_\varepsilon(e)$ and $|N_\varepsilon(e)| \geq J_\varepsilon$. J_ε is usually called a *minimum neighborhood support.*

Definition 4.3 (Density Reachable). An example e' is *density reachable* from e if there is a chain of examples $e_1 \longrightarrow \ldots \longrightarrow e_n$ that satisfies $e_1 = e, e_n = e'$, and e_{i+1} is directly density reachable from e_i ($1 \leq i \leq n-1$). We define the length of the shortest chain as *density level* of e' (e.g., if e' is directly density reachable from e, its density level is 1).

In our problem, D denotes all of the sentences and comments (i.e., $D = S \cup C$). P and U are the positive and unlabeled set, respectively, as defined in Equation 4. For a positive instance $e_0 \in P$, if $|N_\varepsilon(e_0)| \geq J_\varepsilon$, then we consider e_0 to be located in a reasonably dense space and that its neighbors are directly density reachable to e_0. Otherwise, we consider e_0 to be located in a sparse space and that the neighbors are not density reachable. J_r is typically set to 5, as recommended in [15].

Note that we perform the density extension in an iterative manner. After obtaining e_0 $'s$ ε-neighborhood $\{e_{1,1}, e_{1,2}, \ldots, e_{1,n_1}\}$, for each newly added neighbor $e_{1,i}$ $(1 \leq i \leq n_1)$, we continue to compute its ε-neighborhood. If $e_{1,i}$ is located in a dense region, we add its neighbors that were not previously included to a new set (i.e., $\{e_{2,1}, e_{2,2}, \ldots, e_{2,n_2}\}$). Correspondingly, the density level of the neighbors newly added to e_0 is 2. The extension is iteratively performed to identify the instances with higher density levels $= 3, 4, 5, \ldots$ until no further extension is possible so that we obtain a multi-level instance set for each labeled instance $e_0 \in P$. Finally, all of the unlabeled instances in the obtained expanded set constitute the potential positive set PP, and the unreachable instances constitute the potential negative set PN.

To avoid introducing false positive instances, as well as to punish the examples far from e_0, we multiply the radius ε by a damping factor ρ $(\rho \leq 1)$ (i.e., $\varepsilon * \rho$), to replace ε in the next iteration. ρ is defined as

$$\rho = \frac{P_l}{P_l + Ext} \tag{6}$$

where P_l and Ext stand for the number of obtained positives before this extension and the number of expanded positives in this iteration, respectively. Intuitively, as more examples are expanded in the current iteration, the smaller examples will be expanded in the next iteration because the next radius $\varepsilon * \rho$ will be smaller. The detailed extension procedure is summarized in Algorithm 2.

ALGORITHM 2: Density-based Extension Algorithm.

```
Input: the positive set $P$, unlabeled set $U$, the radius $\varepsilon$, and the minimum neighbor support $J_\varepsilon$
Output: potential positive set $PP$ and potential negative set $PN$
Initialize $D = P \cup U,\ PP = PN = \emptyset, ext = P$
while $ext \neq \emptyset$ do
    $tmp = \emptyset$
    foreach $instance\ x \in ext$ do
        $N_\varepsilon(x) = \{q| \parallel x - q \parallel \leq \varepsilon, q \in D\}$
        if $|N_\varepsilon(x)| \geq J_\varepsilon$ then
            $tmp = tmp \cup N_\varepsilon(x)$;
        end
    end
    $tmp = tmp - P - PP$
    $PP = PP \cup tmp$
    $ext = tmp$
    calculate the damping factor $\rho = \frac{|tmp|}{|tmp|+|ext|}$
    $\varepsilon = \varepsilon \cdot \rho$
end
```

Cluster Chain Extension. We observe that both the hypersphere and density extension can only expand limited positives because they follow strict requirements, such as expanded positives near the positive centroid, and are located in *densely* distributed regions. This can cause the subsequent classifier fail to identify an accurate boundary between the positive and negative data. Thus, we adapt the idea of local cluster chain [38] for the positive extension. We first employ *K-Means* algorithm to partition the unlabeled data U into n small, unlabeled clusters:

$$U \underset{clustering}{\longrightarrow} \bigcup_{i=1}^{n} UC_i$$

We assume that the examples in each cluster UC_i are likely to share the same label.

Then, we sort the obtained unlabeled clusters UC_i by their distances from P, select those with a distance from P greater than the median distance as the reliable negative set RN, and leave the rest clusters as ambiguous set AM. These clusters are used to build cluster chains as defined below.

Definition 4.4 (Cluster Chain). is a series of clusters $UC_s \to UC_1 \to \ldots \to UC_l \to UC_e$, which starts from the positive set and ends at one of the reliable negative sets, namely, $UC_s = P$ and $UC_e \in RN$.

Finally, we break each cluster chain at its longest edge (or maximal margin). All of the AM clusters within a sub-chain including P are treated as potential positive clusters, while those within the other sub-chain including a cluster in RN are treated as potential negative clusters. Fig. 5 illustrates the scenario where we have 5 reliable negative clusters and 5 ambiguous clusters. The arrows present the cluster chains and the ambiguous clusters that surrounded by the broken lines are treated as the potential positive clusters. The details of the strategy are presented in Algorithm 3.

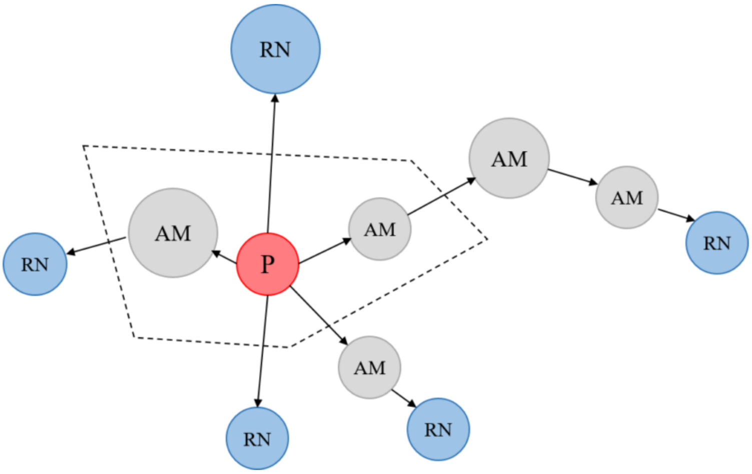


Fig. 5. Cluster Chain Based Positive Extension (P, AM and RN represent the initial positive set, an ambiguous cluster and a reliable negative cluster respectively)

.

Note that each extension method has its advantages and drawbacks, e.g., *hypersphere* is simple and *density* introduces less noise, but both expand limited positives; *cluster-chain* identifies a more accurate boundary but its robustness relies on the sub-clusters generation. Therefore, in this paper we take the union of the three extended positive sets from the above three methods as the *final* potential positive sets FPP (correspondingly, we obtain the *final* potential negative set FPN by taking the intersection of three potential negative sets). For each example $e \in FPP$, we assign its weight $w(e) \in \{1, 2, 3\}$ as the number of methods that contain it (i.e., we favor those potential

ALGORITHM 3: Cluster Chain Extension Algorithm

```
Input: the positive set $P$, unlabeled set $U$
Output: potential positive set $PP$ and potential negative set $PN$
$U \rightarrow \{UC_i\}_{i=1}^{|UC|}$ ; $md \leftarrow Median(dist(UC_i, P))$
$RN \leftarrow \bigcup_{dist(UC_i,P))>md} UC_i$
$AM \leftarrow \bigcup_{dist(UC_i,P))<md} UC_i$
$i \leftarrow 0$
while $AM \neq \emptyset$ do
    build $CH_i$ start with $P$
    while $Tail(CH_i) \neq \emptyset$ do
        find a nearest cluster for $Tail(CH_i)$ and add it to $CH_i$
    end
    $i \leftarrow i + 1$
end
$PP \leftarrow \emptyset, PN \leftarrow \emptyset$
foreach $CH_i$ do
    break $CH_i$ at the longest edge $(CH_i^k, CH_i^{k+1})$
    $PP \leftarrow PP \cup \bigcup_{m \leq k} CH_i^m$
    $PN \leftarrow PN \cup \bigcup_{m \geq k+1} CH_i^m$
end
```

positive examples supported by multiple (2 or 3) extension methods over an individual method as they are more reliable). Through positive extension, the example unbalance problem is effectively alleviated (the initial positives normally cover 3 − 11 sentences, and the number of examples in *FPP* could reach 150 − 200).

4.2.3 Confidence Estimation. We introduce a weighted *Rocchio* classification model to measure the confidence of each example in final potential positive *FPP* and negative *FPN*, which takes $P \cup FPP$ and *FPN* as positive and negative training instances, respectively. Since the potential positive examples have different weights, we adapt the prototype vector construction formulas [8] to its weighted version as follows:

$$\mathbf{p} = \frac{\mu \cdot \sum_{e \in P \cup FPP} \frac{w(e)\cdot \mathbf{e}}{\|\mathbf{e}\|}}{|P \cup FPP|} - \frac{\nu \cdot \sum_{e \in FPN} \frac{\mathbf{e}}{\|\mathbf{e}\|}}{|FPN|} \tag{7}$$

$$\mathbf{n} = \frac{\mu \cdot \sum_{e \in FPN} \frac{\mathbf{e}}{\|\mathbf{e}\|}}{|FPN|} - \frac{\nu \cdot \sum_{e \in P \cup FPP} \frac{w(e)\cdot \mathbf{e}}{\|\mathbf{e}\|}}{|P \cup FPP|} \tag{8}$$

where $\mathbf{e}$ denotes the feature vector of e (which can be a sentence or comment) with weight $w(e)$, $|\cdot|$ stands for the set cardinality, and $\|\mathbf{e}\|$ returns the norm of vector $\mathbf{e}$. $\mathbf{p}$ and $\mathbf{n}$ are our constructed positive and negative prototype vectors, and the parameters μ and ν are set as 16 and 4, as recommended in [8].

For each unlabeled instance $u \in U$, we compute its cosine similarities with $\mathbf{p}$ and $\mathbf{n}$: if $sim(\mathbf{u}, \mathbf{p}) > sim(\mathbf{u}, \mathbf{n})$, then u will be added to the likely positive set *LP*, otherwise the likely negative set *LN*. Finally, the confidence score l_u of each instance $u \in U$ can be calculated as follows:

$$l_u = \frac{\max(cosine(\mathbf{u}, \mathbf{p}), cosine(\mathbf{u}, \mathbf{n}))}{cosine(\mathbf{u}, \mathbf{p}) + cosine(\mathbf{u}, \mathbf{n})} \tag{9}$$

Note that the range of l_u is $(0.5, 1]$, and a larger value indicates a higher quality example. The confidence score is set to 1 for all the initial positive instances in P.

4.2.4 Weighted Support Vector Machine Classifier. Now, we have obtained a set of training examples $\{(\mathbf{e}_1, y_1), (\mathbf{e}_2, y_2), \ldots, (\mathbf{e}_\mathbf{n}, y_n)\}$, where $\mathbf{e}_\mathbf{i}(1 \leq i \leq n)$ denotes the feature vector for each sentence or comment and y_i is its label $y_i \in \{+1, -1\}$, which indicates whether it relates to the target topic t_j. Then, we build the final classifier using the Weighted Support Vector Machine (WSVM), whose optimizing goal is

$$\begin{gathered} Minimize : \frac{1}{2}\mathbf{w}^T\mathbf{w} + C_P \sum_{i \in P} \xi_P + \\ C_{LP} \sum_{j \in LP} \xi_{LP} + C_{LN} \sum_{k \in LN} \xi_{LN} \\ subject\ to : y_i(\mathbf{w}^T\mathbf{e}_i + b) \geq 1 - \xi_i,\ i = 1, 2, \ldots, n \end{gathered}$$

where C_P, C_{LP} and C_{LN} represent the penalty factors of misclassification for three types of training examples: the original positive set P, the likely positive set LP and the likely negative set LN. We directly apply the average confidence score to each example in P, LP and LN as C_P, C_{LP} and C_{LN} because we are more confident about the positive set P than the likely positive set LP or likely negative set LN. Correspondingly, we apply a larger penalty if examples from P are misclassified as negative compared to if examples from LP are classified as negative and if examples from LN are classified as positive.

Finally, we employ our constructed weighted SVM model to classify the comments in C. Those comments classified as the positive class are aligned to the target topic t_j. Apparently, we can perform the alignment task by repeating the entire process, as summarized in Algorithm 4, on all of the extracted topics from news. Specifically, for the comments having multiple alignments with news topics, we only keep at most three aligned results as suggested by the observation in Section 3.2.3.

ALGORITHM 4: Positive and Unlabeled Learning

```
Input: sentences set $S$, comments set $C$, topic distribution $\theta_s$ $\theta_c$
Output: A set of aligned topic-comment pairs $R$
Initialization: $R = \emptyset$
foreach topic $t_j$ do
    1) Identity positive examples $P$ and unlabeled example set $U$ using Equation 4
    2) Employ combined positive extension method to obtain final potential positive set $FPP$ and final
       potential negative set $FPN$
    3) Build weighted Rocchio classifier by computing prototype vectors $\mathbf{p}$ and $\mathbf{n}$
    4) Initialize $LP = LN = \emptyset$
    foreach $u \in U_1 \cup U_2$ do
        if $cosine(\mathbf{u}, \mathbf{p}) > cosine(\mathbf{u}, \mathbf{n})$ then
            $LP = LP \cup \{u\}$
        else
            $LN = LN \cup \{u\}$
        end
    end
    5) Build WSVM classifier and let $R_{t_j} \subseteq C$ denotes the final positive examples in comments
    6) $R = R \cup \{< c_i, t_j > | c_i \in R_{t_j}\}$
end
```

5 EMPIRICAL EVALUATION

In this section, we evaluate our proposed alignment method using the benchmark datasets introduced in Section 3.1. First, we report the alignment performance, including baseline methods, the evaluation metrics, and the experimental results for different approaches with various settings. Then, we investigate the factors that could influence the alignment, including the comparison of our three positive extension methods introduced in Section 4.2, whether similar news help improve the performance and the hyperparameter settings in the proposed method. Finally, we conduct an error analysis for the wrongly aligned and null aligned comments and discuss how to further improve the current method.

5.1 Alignment Result Evaluation

In this section, we first introduce the experiment settings including baseline methods and evaluation metrics, then report the overall results and detailed comparison between our proposed approach and its previous version in [22].

5.1.1 Baseline Methods. Generally, we use T-PU to denote the two-phase alignment technique which combines topic extraction and positive unlabeled learning. According to different strategies used in positive extension, T-PU^h, T-PU^d, T-PU^c, T-PU$^+$, T-PU^u respectively denote the alignment methods with hypersphere, density, cluster chain, combined strategy and unweighted classifier. To test the effectiveness of T-PU$^+$ proposed in this paper, we compare it with the following 6 existing methods:

- ***VSM*** is a simple similarity based method that uses word-level features. Specifically, it employs *TF-IDF* for both sentence and comment representation and applies cosine similarity for selecting the most related sentence in the news to align with each comment.
- ***BSVM*** is a classification-based method that also uses word-level features. It trains a binary classification model for *each sentence* in the news article and subsequently uses it to predict whether a comment is related to the sentence. In our experiment, we use Libsvm[6] for building classification models. Note that the training examples are extremely unbalanced, i.e., for a specific sentence, the number of related comments ranges from 0 to 78 (3.93 in average) while irrelevant comments can be over 100.
- ***DCTM*** is a straightforward method that uses topic-level features. It classifies the comments according to the distribution obtained by our proposed topic model, in the same manner as in the positive example identification in Equation 4.
- ***MSTM*** is the topic model introduced in [35]. The only difference from DCTM is, MSTM imposes a constraint that each comment corresponds to exactly one topic.
- ***SCTM*** is the topic model introduced in [10]. It allows a news-comment pair to have more than one topic vector. In our experiment, we use the released implementation[7]. To obtain the topic distribution of sentences, we add them to the comments set when applying this model.
- ***T-SVM*** is a supervised method that requires users to provide manually labeled training examples. The only difference from ***BSVM***, is that classifiers here are built on the *topics* extracted by DCTM instead of individual sentences.

Among the above 6 methods, *BSVM* and *T-SVM* are supervised methods, and the others are unsupervised. Clearly, labeling training examples in supervised learning is a time-consuming process. Additionally, for different news articles and corresponding comments, supervised learning

[6] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[7] http://thetb.github.io/sctm/

methods must perform repeatedly labeling and thus they are not suitable for this task. We include them here for comparison purposes to the benchmark with other unsupervised methods, including our T-PU$^+$. Note that out of the 6 existing methods, *VSM* and *BSVM* only use the word-level features whereas the other methods rely on different topic models.

5.1.2 *Evaluation Metrics.*

Aligning the comments with the news sentences at the topic level can be regarded as a classification problem. Thus, we adapt the widely used evaluation metrics for text classification for our evaluation and comparison, namely, *precision*, *recall* and *F1-Value.*

In particular, let the sentence-comment set pair be (S, C). For each comment $c_i \in C$, let $r_i \subseteq S$ be the set of aligned news sentences, labeled by annotators. If $|r_i| > 1$, then c_i has multiple related news sentences, while $|r_i| = 0$ indicates c_i has no related news sentences. Let $\tilde{r}_i$ $(\tilde{r}_i \subseteq S)$ be a set of sentences that are predicted to align with comment c_i by a prediction method. Then, we consider that the prediction for c_i is correct if $r_i \cap \tilde{r}_i \neq \emptyset$. Finally, we can define the three evaluation metrics as follows:

$$Precision = \frac{|\{c_i | c_i \in C, r_i \cap \tilde{r}_i \neq \emptyset\}|}{|\{c_i | c_i \in C, \tilde{r}_i \neq \emptyset\}|}$$

$$Recall = \frac{|\{c_i | c_i \in C, r_i \cap \tilde{r}_i \neq \emptyset\}|}{|\{c_i | c_i \in C, r_i \neq \emptyset\}|}$$

$$F1 - Value = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Note that in our scenario, *precision* is more important than *recall* because users prefer to read fewer relevant comments on a certain topic rather than many comments with irrelevant or noisy information. To make the comparison fair, we use the topic results from our developed DCTM for those methods that require topic information.

5.1.3 *Overall Results.*

Table 4 demonstrates the overall performance of our proposed T-PU$^+$, and the comparisons based on average results with other six methods on 10 and 12 datasets from Sina and Yahoo!, respectively. We make the following observations:

- Generally, all the two-phase methods outperform the one-step methods. These results show that two-phase methods can combine the advantages of generative and discriminative models as mentioned at the beginning of Section 4.2, and thus improve the alignment performance.
- All the T-PUs outperform the baseline methods, VSM and DCTM, because our methods utilize topic-level features and word-level features, whereas the other two methods only use one of them.
- As for the three different topic extraction methods, MSTM always gets lower recall and SCTM is not so good at precision, while our proposed DCTM achieves the best overall performance. The reasons for such results are the problem setting and different assumptions of three topic models: MSTM requires all the slave (comment) topics rely on only one master (news) topic but from Section 3.2 we find that not only can comments relate to multiple topics but it can have its own topics; SCTM assumes comments are associated with multiple topic distributions, which increases their topic diversity but has side effect on the alignment precision.
- Compared with BSVM, our methods achieve significantly better results. While it is difficult to build an accurate classifier with very few positive examples and noisy negative examples, our positive extension process can split the latter into likely positive and likely negative examples, which enhances the limited positive set as well as purifies the negative set.

- As the intermediate result, the performance of Rocchio classifier is not such satisfactory. It suffers similar problem with the generative models (i.e., DCTM), namely, the achieved decision boundary is not accurate enough, and thus it tends to retrieve a small part of alignments with relatively high precision but low recall.
- When comparing the four simplified versions of T-PU$^+$ that only using one positive extension strategy or unweighted data, we can find T-PU^h in our previous work [22] outperforms the T-PU^d and T-PU^c because the density-based method always obtains high-quality but less extension results than hypersphere while cluster-chain achieves more noisy extension on the contrary, which are reflected in the final results. The fully armed method (i.e. T-PU$^+$) consistently outperforms T-PU^h in terms of all three metrics. Its *precision* obtains the highest relative improvement (+1.2% on Sina and +1.6% on Yahoo!). The unweighted T-PU^u can achieve comparable recall, but its precision is less than T-PU$^+$ possibly because it put excessive attention to those noise examples.
- T-SVM, which performs the best among all of the methods, obtains an approximately 1.5-3.7% improvement over T-PU$^+$. However, T-SVM is a supervised method that relies on the quality and quantity of labeled data, while our positive unlabeled learning based methods, T-PU$^+$, can achieve comparable results without using labeled examples; thus, our method is more appropriate and preferable for the news comment alignment task.

Table 4. Overall Results on Two Datasets

Methods	**Sina**			**Yahoo!**		
	Precision(%)	Recall(%)	F1-Value(%)	Precision(%)	Recall(%)	F1-Value(%)
VSM	70.1	37.7	49.0	68.8	38.1	49.0
BVSM	59.4	52.1	55.5	57.7	51.9	54.6
DCTM	65.9	40.4	50.1	63.4	43.6	51.7
MSTM	63.3	39.1	48.3	62.1	41.5	49.8
SCTM	60.6	41.8	49.5	60.3	44.7	51.3
T-SVM	77.4	58.2	66.4	77.8	64.1	70.3
Ricchio	68.3	42.5	52.4	65.7	48.9	56.1
T-PU^u	70.1	56.9	62.8	70.8	63.0	66.7
T-PU^h	75.3	56.7	64.7	74.9	63.4	68.7
T-PU^d	73.7	51.4	60.6	72.5	59.3	65.2
T-PU^c	72.1	57.1	63.7	71.4	63.2	67.1
T-PU$^+$	76.2	56.6	65.0	76.1	63.6	69.3

Note: Ricchio represents our method omitting the last weighted SVM classifier, and T-PU^u means replacing the last two steps with standard SVM without example weights.

5.1.4 T-PU^h and T-PU$^+$. Next, we present ten datasets with the most significant improvements between T-PU^h and T-PU$^+$ in Table 5, and the results show that T-PU$^+$ can achieve a slight improvement when applied to the English datasets compared with the Chinese datasets.

We also verify that the improvement of T-PU$^+$ over T-PU^h is not random. As suggested in [11], we conduct a paired t-test, which checks whether the average difference in their performance over the data sets is significantly different from zero. Let v_i and v_i^+ be the performance scores (e.g., *precision*) of T-PU^h and T-PU$^+$ on the *i-th* dataset, and δ_i be their difference $v_i^+ - v_i$. Then, the

Table 5. Precision Comparison of T-PU^h and T-PU$^+$

Data		Precision(%)		Improvement
		T-PU^h	*T-PU$^+$*	
Sina	cn-9	72.1	74.4	3.19%
	cn-6	75.6	78.0	3.17%
	cn-8	80.7	82.7	2.48%
	cn-4	68.2	69.4	1.76%
	cn-2	78.7	79.8	1.40%
Yahoo!	en-10	75.2	77.7	3.32%
	en-2	74.4	76.4	2.69%
	en-6	70.6	72.3	2.41%
	en-3	75.6	77.3	2.25%
	en-9	80.6	82.4	2.23%

t-statistic is computed as

$$t - value = \frac{\bar{\delta}}{\sigma_d / \sqrt{Q}} \tag{10}$$

where $\bar{\delta}$ and σ_d are the average and standard deviation of all differences and Q is the number of news. We calculate that $t = 3.41$ over all 22 news articles. It is greater than $t_{0.05/2,21} = 2.080$. Thus, we can conclude that the improvement is significant.

5.2 Positive Extension Evaluation

In this section, we compare three positive extension methods introduced in Section 4.2. For simplicity, we report their performance on a selected news article with 37 sentences, 531 comments and 810 labeled alignments in total – the results on other news articles follow the samilar trend. The labeled dataset is relatively small for a typical classification model, but actually the extension methods use all the news comments (both labeled and unlabeled) and the evaluation is conducted only on the labeled part (*a hot news item normally attracts thousands of comments and we managed to label part of them for evaluation purpose as described in Section 3.1*). Table 6 shows the performance (in terms of precision and recall) of all extracted topics, and it conforms with the characteristics of different methods, namely,

- The density extension achieves the best *precision* because it follows stricter requirements than the hypersphere, whereas the cluster chain obtains more extensions and performs well in term of *recall* since its aim is to find the accurate boundary, as mentioned in Section 4.2.2.
- The results generated using the three extension methods overlap with each other and *recall* is significantly improved if we combine their extension results. The positive extension is to alleviate the example unbalance problem by identifying some potential positives from the unlabeled data, so the *recall* is more meaningful under the premise of reasonable *precision.* That's why we propose an integrated approach to combine the three extension results in the following steps, with a view to reach higher recall, without greatly sacrificing the precision.
- Through an investigation of the topics that still have lower recall using our combined method, we find that the missing comments are related to multiple topics and thus failed to be reached by certain topics, or use rhetorical expression and lack background knowledge.

- In terms of efficiency, the sphere extension is much faster (5-7 times faster) than the cluster chain extension, which is slightly more efficient than the density extension.

Besides the above comparison, we also generate the distribution of the extended positive comments as we did in the data observation (Section 3.2.4), and compute their KL-divergence to the true distribution from labeled data. The last line in Table 6 presents the results, from which we can see that, only focusing on high-similarity region (i.e., hypersphere and density) does not achieve the ideal distribution, while the combined method which balances the precision and recall gets the best result that is much closer to the true distribution.

In summary, the combined method addresses the imbalanced data problem to a certain extent (i.e., the *final* potential positive set *FPP* is 20-30 times larger than the initial limited positive set *P* as described in Section 4.2.2).

Table 6. Results on Positive Extension

Topic	Sen./Com.	Sphere		Density		Chain		Combined	
		Prec.	Recall	Prec.	Recall	Prec.	Recall	Prec.	Recall
0	10/361	66.2	27.7	65.4	14.7	48.0	32.4	53.3	40.7
1	6/87	19.1	31.0	28.1	31.0	27.0	50.6	29.1	67.8
2	10/180	37.6	21.1	43.0	22.2	35.3	39.4	38.2	49.4
3	8/98	20.9	38.8	37.7	29.6	25.0	33.7	29.7	78.6
4	11/128	32.3	49.2	45.9	39.8	41.8	57.8	33.2	73.4
KL-divergence		0.4027		0.4134		0.3968		0.3755	

Note: in all the 5 topics extracted through DCTM, *Sen./Com.* are the numbers of initial positive examples and the labeled linked comments, the following columns present the results obtained through single or combined method.

5.3 Does Similar News Help?

Besides the unlabeled comments, similar news can also be used to enrich the initial positive set. Similar news articles often form events to track a theme during a specific time period, namely, there exists strong content correlation among them. Therefore, we make a preliminary attempt to use similar news for positive extension. Particularly, for a given news article, we search similar news that meets the following requirements:

- **similar news content**: we need similar, not exactly the same news article, so we first extracted the named entities (namely, persons, locations and organizations), based on which we retrieved similar news articles that contain all the identified named entities. In other words, we want to find the similar news that are talking about the same persons and organizations in some specific locations.
- **similar publish time**: news is defined as *packaged information about current events happening somewhere else* in Wikipedia [8], which states it is a timely media. Thus, we limit the similar news with publishing time within a range of [-3, 3] days.
- **same website**: different news portals have their own views and preference, we restrict the same news website to keep the news articles be consistent with each other in event reporting, the users' habits and preference.

In our experiments, five Chinese news and eight English news could find suitable similar news. Currently, the similar news articles are only used to enrich the topic modeling and positive extension,

[8] https://en.wikipedia.org/wiki/News

and we could further analyze the commenting patterns over similar news or users' tastes if enough datasets are collected in the future.

Table 7. Precision Comparison without/with Similar News

Dataset	Precision(%)		Improvement
	without similar news	*with similar news*	
Sina	75.3*	73.9	-1.86%
Yahoo!	74.7*	72.2	-3.35%

Note: The precisions are different from those in Table 4 because we only include datasets that find suitable similar news, i.e., 5 Chinese and 8 English news.

Preliminary results in Table 7 show that including similar news did not lead to direct performance improvement. Based on the result analysis, we find that the effect of introducing similar news for positive extension is twofold: on the one hand, similar news articles alleviate the data sparseness in statistical topic model and further increase the topic diversity, which could help enlarge the initial positive set, as well as improve the extension recall; on the other hand, similar news articles inevitably bring more or less unrelated information (or *noise*), diluting the quality of initial sentences that focus on some specific topics, which might have side effects on all the three extension methods (especially hypersphere). Despite all of these, we still believe that the alignment could be more meaningful with the addition of much more elaborate data (e.g., similar news and associated comments or even comments from the same user). Nevertheless, much more focused research need to be done to carefully select those relevant news and corresponding comments that are useful to enhance the current limited news sentences without introducing noise.

5.4 Hyperparameters

Here, we discuss the hyperparameter settings in our proposed topic model and positive extension methods.

5.4.1 Parameters in Topic Model. We take fixed values for the hyperparameters α and β (i.e., $\alpha = 0.5, \beta = 0.01$) according to model perplexity [6], which measures how well topic models predict unseen test data. The *Beta* parameters γ_c and γ_s are defined to represent our preference for news-inherited topics. We apply two methods for setting their values, calculating the common-word ratio and referring the comment-sentence distribution in Section 3.1. The results show that the DCTM is not very sensitive to the prior, and thus, we set fixed values with stable performance, i.e. $\gamma_c = 5$ and $\gamma_s = 0.2$.

Besides the above priors, another important parameter is the number of topics K. Generally, the proper K selection is also based on perplexity. However, topic is employed as bridge between news and comments in our alignment task, so we use alignment precision as the metric to set the proper K. Fig. 6 shows how the alignment precision on two datasets changes when K is varying from 2 to 10, from which we can see that: the alignment precision fluctuates sharply when the number of topics is small and reaches the best value at 5. Then the performance decreases with K grows. The result makes sense since a typical news article will not cover too many topics.

5.4.2 Parameters in Positive Extension. In the density extension, we should specify the initial radius ε. Table 8 shows how the radius ε changes with the extension levels in for the news article. Note that all of the radius values are divided by the hypersphere radius so that they fall into (0,1]. From the table, we observe the following: 1) If we set ε to the same radius with hypersphere extension, the density search becomes the hypersphere extension, whereas too small of a value

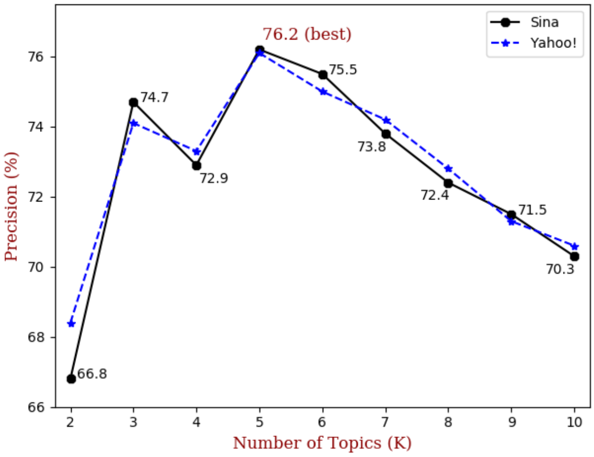


Fig. 6. Precision v.s. number of topics K (the values denote the precision on Sina dataset)

often leads to more computations. In addition, 2) the algorithm can converge quickly (4 levels at most), even when starting with a very small ε. Throughout the evaluations, we experimentally select $\varepsilon = 0.4 * r$ by considering both extension quality and efficiency.

Table 8. Damping Radius in Density Extension

Initial Radius	Radius after Extension			
	1	2	3	4
1.0	0.075	-	-	-
0.8	0.113	0.079	-	-
0.5	0.232	0.139	0.078	-
0.2	0.154	0.107	0.083	0.078

5.5 Error Analysis

Finally, we investigate comments for which our method fails to find a suitable mapping news topic. We find that the main reason is the topic drift. A drift example is shown in Fig. 7. It is a report about a *rocket launch* in *North Korea*. We find that there are no comments on Topic 0 (background topic; therefore, it is typical that users do not comment on it), and Topic 2 discusses the *launch cost* in the news. However, through topic drift, it changes to a different topic, *food aid*, leading to a failed mapping/detection. Similar cases occur in other news articles, which contribute most of the failed cases.

Another typical failure is caused by a lack of background or domain knowledge. For example, when discussing Mo Yan[9], the winner of the 2012 Nobel Prize in Literature, commentators do not discuss topics mentioned in the news. Instead, they focus on discussing other similar/famous people or events (such as Yang Zhenning). It is challenging, if not impossible, to align these comments with the news if we do not know that both of them are Chinese Nobel Prize winners, indicating

[9] http://news.sina.com.cn/c/2012-12-07/032125750538.shtml

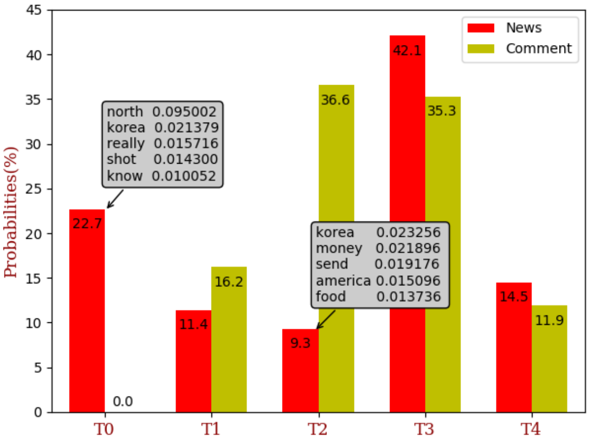


Fig. 7. Topic distribution in news and comments

that we need to consider some knowledge bases and related news articles (instead of just current news) to further enhance the news comment alignment.

6 RELATED WORK

There are three branches of research that are related to the news comment alignment task in this paper, namely, news and comment analysis, multi-source topic modeling and positive unlabeled learning. In this section, we review and discuss the recent work and literatures in these research.

6.1 News and Comment Analysis

Lu and Zhai studied how to automatically integrate opinions by well-written experts, with many opinions scattered in various sources such as blogs, spaces and forums, and proposed a semi-supervised solution model [34], which inspires us to design a document and comment topic model. Many researchers leveraged social information (comments, social relationship or social behavior like forwarding tweet) for news summarization [23, 46, 54]. The objectives of these studies are different from ours because they target data integration and document summarization by exploiting additional comment, whereas we attempt to find relationships between news and comments.

Recently, Sil *et al.* proposed to allow users to read news along with relevant comments and presented a supervised learning method for linking comments to news segments [44, 45]. Guo *et al.* proposed a graph-based latent variable model that modeled the inter short text correlations between tweet and news, and further employed weighted textual matrix factorization to link tweets to related news, thus enriched the event context of tweets [18]. Das *et al.* proposed the problem of specific comment location, which tried to identify comments that focus on specific parts of news/blog rather than the whole article [10]. Note that our method is topic-based, and is more flexible than segment-based methods [45] because all of the sentences within a segment must be continuous in the document, and the sentences within a topic can be distributed across the entire news article (see Fig. 1). Phan *et al.* presented a general framework for building classifiers that addressed short and sparse text and Web segments by fully utilizing hidden topics discovered from large-scale data collections [40]. As mentioned before, hand-labeling training sets in supervised learning methods is typically a time-consuming task, and the training set labeled for a given document and the corresponding comments can only be used once. Our proposed unsupervised framework can address the problem of lacking training data.

6.2 Topic Modeling

Hong *et al.* studied and discussed the effectiveness of existing topic models in microblogging environments [20]. Ramage *et al.* presented a scalable implementation of a partially supervised learning model (Labeled LDA [42]) that mapped the content of a Twitter feed into predefined dimensions (substance, style, status, and social characteristics of posts) [41]. Zhao *et al.* empirically compared the content of Twitter with a traditional news medium using unsupervised topic modeling *Twitter-LDA* [55]. However, none of them considers the dependence between different types of media.

For modeling multi-source documents, Blei *et al.* modeled pictures and textual annotations [4] and developed supervised topic models [5], where each document was paired with a response, to infer latent topics predictive of the response. Wang *et al.* further applied category information to picture modeling and classification [51]. Tang *et al.* proposed qLDA to extract an informative summary from a document collection for a given query [49], and developed cross-domain topic learning to learn and differentiate collaboration topics from other topics [48]. Hong *et al.* extended standard topic models by allowing each text stream to have both local and shared topics [21].

Another line of research models document relationships as links, wherein links are treated as attributes of documents just like words or utilized as degrees in graph theory. PHITS [9], as well as Link-LDA [14], used citations among papers as a special attribute to model the paper corpus with PLSI and LDA. However, the document dependence revealed by citations is ignored in these models. Pairwise link-LDA, Link-PLSI-LDA [36] and the CT model [19] were proposed to address the dependence problem, in which word generation was in part or all controlled by links. Liu *et al.* developed a Bayesian hierarchical approach that performs topic model and author community discovery in a unified framework [33]. Tang *et al.* proposed a two-layer Restricted Boltzmann Machine to simultaneously model the links and words, where they are linked together by a layer in the undirected graphical model [50].

Our proposed document comment topic model (DCTM) is partially inspired by qLDA [49] and pairwise link-LDA [36]. Ma *et al.* also proposed a similar model named Master-Slave Topic Model (MSTM) [35] based on CorrLDA [37], and the difference is that our DCTM allows comments containing multiple topics and could be generated without depending on news content, which is a natural reflection of users' commenting behavior.

6.3 Positive Unlabeled Learning

Learning only from positive data was conducted in the one-class SVM [43], which tried to learn the support of the positive distribution. Meanwhile, there has been considerable interest in learning from a small number of labeled positive and negative examples with a large number of unlabeled examples, e.g., Nigam *et al.* used naive Bayesian and EM algorithm [39], Joachims *et al.* proposed transductive SVM [24], and Blum *et al.* exploited the conditional independence of multiple data views to allow for co-training [7].

The theoretical study of Probably Approximately Correct learning from positive and unlabeled examples was conducted in [12], and the research problem was first proposed in [32], which showed that maximizing the number of examples classified as negative while constraining the function to correctly classify positive examples provided good performance with a large sufficient sample size. The core process could be described as the following two steps: identify a set of reliable negative examples from the unlabeled set U and then iteratively build a classifier using EM or SVM. The naive Bayesian algorithm was modified to learn from positive and unlabeled examples in [13]. Further study by [26–31, 38] proposed various negative example extraction methods and adapted many existing learning methods (e.g., SVM) into the positive unlabeled version.

The PU learning method in this paper typically follows the above two steps, but we propose three positive extension algorithms based on in-depth data analysis and assign the pseudo-positive examples different levels of confidence.

7 CONCLUSIONS

In this paper, we address the most fundamental problem in news and comments analysis: finding the corresponding news sentences (or topics) for each user comment. We propose a novel, two-phase framework to automatically accomplish this task. In particular, we present a document comment topic model to extract topics from news and comments by leveraging their content dependency. Then we introduce a positive and unlabeled learning method to build an accurate classifier that does not require users to provide labeled examples. Specifically, we design three positive extension methods (i.e., hypersphere, density and cluster chain extension) based on data observations. To evaluate the proposed methods and framework, we collect heatedly-discussed news in Chinese and English from influential news portals (i.e., Sina and Yahoo!) and manually build gold standard via crowd-sourcing. Experimental results demonstrate that our proposed framework can achieve comparable results with supervised learning methods that needs time-consuming and labor-intensive manually labeling process.

News comment alignment is a challenging yet interesting problem, and there are many potential future directions for this work. For example, users have their own tastes and often follow and comment similar news or event, and there should be potential associations between the comments over similar news from the same user. Thus, we can study the alignment over similar news (i.e., news reports on the same event) and explore the common comment patterns as well as the users' personal profiling and focus. Moreover, users are not isolated, they are actually linked by some offline relationships or form an implicit social network when browsing or commenting news, and there might be some correlation between the users' relationships and commenting behavior. As such, we can further investigate whether users' relationships can help the alignment, and study the topic drifts in their comment interactions.

APPENDIX

According to the generative process, we could integrate out the multinominal and Bernoulli distributions $\theta_n, \theta_c, \phi, \lambda$, because the model only use conjugate priors. For derivation, we write the joint probability:

$$\begin{aligned} p(\mathbf{w}, \mathbf{z}, \mathbf{x}|\alpha, \beta, \gamma_n, \gamma_c) \propto & \int p(\mathbf{x}|\lambda)p(\lambda|\gamma_n, \gamma_c)d\lambda \int p(\mathbf{z}|\theta_c, \mathbf{x})p(\theta_c|\alpha)d\theta_c \\ & \int p(\mathbf{z}|\theta_n, \mathbf{x})p(\theta_n|\alpha)d\theta_n \int p(\mathbf{w}|\phi, \mathbf{z})p(\phi|\beta)d\phi \end{aligned} \tag{11}$$

The conditional of x_i is obtained by dividing the joint distribution of all variables by the joint with all variables but x_i (denoted by $\mathbf{x}_{\neg i}$ where $\neg$ means exclusion) and canceling factors that do not depend on $\mathbf{x}_{\neg i}$:

$$\begin{aligned} p(x_i = 0|\mathbf{x}_{\neg i}, \mathbf{z}, \cdot) &= \frac{p(\mathbf{w}, \mathbf{z}, \mathbf{x}|\alpha, \beta, \gamma_n, \gamma_c)}{p(\mathbf{w}, \mathbf{z}, \mathbf{x}_{\neg i}|\alpha, \beta, \gamma_n, \gamma_c)} \\ &= \frac{\int p(\mathbf{x}|\lambda)p(\lambda|\gamma_n, \gamma_c)d\lambda}{\int p(\mathbf{x}_{\neg i}|\lambda)p(\lambda|\gamma_n, \gamma_c)d\lambda} \cdot \frac{\int p(\mathbf{z}|\theta_c, \mathbf{x})p(\theta_c|\alpha)d\theta_c}{\int p(\mathbf{z}|\theta_c, \mathbf{x}_{\neg i})p(\theta_c|\alpha)d\theta_c} \end{aligned} \tag{12}$$

Now we derive the first fraction. As we assume that x_i is generated from a Bernoulli distribution λ whose Beta parameters are γ_n, γ_c, then we have $p(\mathbf{x}|\lambda) = \prod_d \lambda_d^{n_{dx_0}} \cdot (1-\lambda_d)^{n_{dx_1}}$, where n_{dx_0} is the number of times that $x = 0$ has been sampled in document d and n_{dx_1} represents the number of times that $x = 1$ has been sampled in d. Because Beta is the conjugate prior of Bernoulli, we could solve the Bernoulli-Beta integral using Gibbs sampling. Specifically,

$$\begin{aligned}\int p(\mathbf{x}|\lambda)p(\lambda|\gamma_n,\gamma_c)d\lambda &= \prod_d \frac{1}{B(\gamma_n,\gamma_c)} \int_0^1 \lambda_d^{n_{dx_0}+\gamma_c-1} \cdot (1-\lambda_d)^{n_{dx_1}+\gamma_n-1} d\lambda_d \\ &= \prod_d \frac{B(n_{dx_1}+\gamma_n, n_{dx_0}+\gamma_c)}{B(\gamma_n,\gamma_c)} \\ &= \prod_d \frac{\Gamma(n_{dx_1}+\gamma_n)\Gamma(n_{dx_0}+\gamma_c)\Gamma(\gamma_c+\gamma_n)}{\Gamma(n_{dx_0}+n_{dx_1}+\gamma_c+\gamma_n)}\end{aligned} \tag{13}$$

Substitute the equation above into the first fraction and use $\Gamma(x+1) = x\Gamma(x)$ for simplification, we get:

$$\begin{aligned}\frac{\int p(\mathbf{x}|\lambda)p(\lambda|\gamma_n,\gamma_c)d\lambda}{\int p(\mathbf{x}_{\neg i}|\lambda)p(\lambda|\gamma_n,\gamma_c)d\lambda} &= \frac{\prod_d \frac{\Gamma(n_{dx_1}+\gamma_n)\Gamma(n_{dx_0}+\gamma_c)\Gamma(\gamma_c+\gamma_n)}{\Gamma(n_{dx_0}+n_{dx_1}+\gamma_c+\gamma_n)}}{\prod_d \frac{\Gamma(n_{dx_1}^{\neg di}+\gamma_n)\Gamma(n_{dx_0}^{\neg di}+\gamma_c)\Gamma(\gamma_c+\gamma_n)}{\Gamma(n_{dx_0}^{\neg di}+n_{dx_1}^{\neg di}+\gamma_c+\gamma_n)}} \\ &= \frac{n_{dx_0}^{\neg di}+\gamma_c}{n_{dx_0}^{\neg di}+n_{dx_1}^{\neg di}+\gamma_c+\gamma_n}\end{aligned} \tag{14}$$

The second fraction can be derived analogously. Specifically, as $p(\mathbf{z}|\theta_c, \mathbf{x})$ and $p(\theta_c|\alpha)$ are conjugate pair of Multinomial-Dirichlet, we can obtain:

$$\begin{aligned}\int p(\mathbf{z}|\theta_c,\mathbf{x})p(\theta_c|\alpha)d\theta_c &= \prod_d \frac{1}{\Delta(\alpha)} \prod_z \theta_{cz}^{n_{dz}+\alpha-1} \\ &= \prod_d \frac{\Delta(\mathbf{n}_d+\alpha)}{\Delta(\alpha)}\end{aligned} \tag{15}$$

where

$$\Delta(\alpha) = \frac{\Gamma(\alpha)^K}{\Gamma(K\alpha)} \tag{16}$$

and n_{dz} denotes the number of times that topic z has been sampled in document d. So the second faction can be written as:

$$\begin{aligned}\frac{\int p(\mathbf{z}|\theta_c,\mathbf{x})p(\theta_c|\alpha)d\theta_c}{\int p(\mathbf{z}|\theta_c,\mathbf{x}_{\neg i})p(\theta_c|\alpha)d\theta_c} &= \frac{\prod_d \frac{\Delta(\mathbf{n}_d+\alpha)}{\Delta(\alpha)}}{\prod_d \frac{\Delta(\mathbf{n}_{d\neg i}+\alpha)}{\Delta(\alpha)}} \\ &= \frac{\frac{\Gamma(n_{z_{di}}+\alpha)\cdot[\prod_d \prod_z \Gamma(n_{dz}+\alpha)]_{\neg di}}{\Gamma(\sum_{z'}(n_{dz'}+\alpha))\cdot[\prod_d \Gamma(\sum_{z'}(n_{dz'}+\alpha))]_{\neg di}}}{\frac{\Gamma(n_{z_{di}}+\alpha-1)\cdot[\prod_d \prod_z \Gamma(n_{dz}+\alpha)]_{\neg di}}{\Gamma([\sum_{z'}(n_{dz'}^{d\neg i}+\alpha)]-1)\cdot[\prod_d \Gamma(\sum_{z'}(n_{dz'}+\alpha))]_{\neg di}}} \\ &= \frac{n_{z_{di}}^{\neg di}+\alpha-1}{\sum_z(n_z^{\neg di}+\alpha)-1}\end{aligned} \tag{17}$$

Finally, we reach Equation 2 by combing these two fractions.

REFERENCES

[1] Ahmet Aker, Emina Kurtic, AR Balamurali, Monica Paramita, Emma Barker, Mark Hepple, and Rob Gaizauskas. 2016. A graph-based approach to topic clustering for online comments to news. In *European Conference on Information Retrieval*. Springer, Heidelberg, Berlin, Germany, 15–29.

[2] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I. Jordan. 2003. An introduction to MCMC for machine learning. *Journal of Machine learning* 50, 1-2 (2003), 5–43.

[3] Michael Barthel, Elisa Shearer, Jeffrey Gottfried, and Amy Mitchell. 2015. *The Evolving Role of News on Twitter and Facebook*. Technical Report. Pew Research Center.

[4] David M. Blei and Michael I. Jordan. 2003. Modeling annotated data. In *Proceedings of the 26th International Conference on Research and Development in Information Retrieval*. ACM, New York, NY, USA, 127–134.

[5] David M. Blei and Jon D. McAuliffe. 2007. Supervised Topic Models. In *Proceedings of 21st International Conference on Neural Information Processing Systems*. Curran Associates, Red Hook, NY, USA, 121–128.

[6] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.

[7] Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*. ACM, New York, NY, USA, 92–100.

[8] Chris Buckley, Gerard Salton, and James Allan. 1994. The Effect of Adding Relevance Information in a Relevance Feedback Environment. In *Proceedings of the 17th International Conference on Research and Development in Information Retrieval*. Springer-Verlag New York, Inc., New York, NY, USA, 292–300.

[9] David A. Cohn and Thomas Hofmann. 2000. The Missing Link - A Probabilistic Model of Document Content and Hypertext Connectivity. In *Proceedings of 14th International Conference on Neural Information Processing Systems*. MIT Press, Cambridge, MA, USA, 430–436.

[10] Mrinal Kanti Das, Trapit Bansal, and Chiranjib Bhattacharyya. 2014. Going beyond corr-lda for detecting specific comments on news & blogs. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, New York, NY, USA, 483–492.

[11] Janez Demšar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research* 7 (Dec. 2006), 1–30.

[12] Francois Denis. 1998. PAC Learning from Positive Statistical Queries. In *Proceedings of the 9th International Conference on Algorithmic Learning Theory*. Springer, Heidelberg, Berlin, Germany, 112–126.

[13] Francois Denis, Remi Gilleron, and Marc Tommasi. 2002. Text classification from positive and unlabeled examples. In *Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Esia, Annecy, France, 1927–1934.

[14] Elena Erosheva, Stephen Fienberg, and John Lafferty. 2004. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences of the United States of America* 101, Suppl 1 (2004), 5220–5227.

[15] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*. AAAI Press, Washington, DC, USA, 226–231.

[16] Evgeniy Gabrilovich and Shaul Markovitch. 2006. Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In *Proceedings of the 21st National Conference on Artificial Intelligence*. AAAI Press, Washington, DC, USA, 1301–1306.

[17] Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. AAAI Press, Washington, DC, USA, 1606–1611.

[18] Weiwei Guo, Hao Li, Heng Ji, and Mona T Diab. 2013. Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media.. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, 239–249.

[19] Zhen Guo, Shenghuo Zhu, Yun Chi, Zhongfei Zhang, and Yihong Gong. 2009. A latent topic model for linked documents. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval*. ACM, New York, NY, USA, 720–721.

[20] Liangjie Hong and Brian D. Davison. 2010. Empirical study of topic modeling in Twitter. In *Proceedings of the 1st Workshop on Social Media Analytics*. ACM, New York, NY, USA, 80–88.

[21] Liangjie Hong, Byron Dom, Siva Gurumurthy, and Kostas Tsioutsiouliklis. 2011. A time-dependent topic model for multiple text streams. In *Proceedings of the 17th ACM International Conference on Knowledge Discovery in Data Mining*. ACM, New York, NY, USA, 832–840.

[22] Lei Hou, Juanzi Li, Xiaoli Li, Jiangfeng Qu, Xiaofei Guo, Ou Hui, and Jie Tang. 2013. What Users Care about: a Framework for Social Content Alignment. In *Proceedings of the 23rd International Joint Conference on Artificial*

Intelligence. AAAI Press, Washington, DC, USA, 1401–1407.

[23] Meishan Hu, Aixin Sun, and Ee-Peng Lim. 2008. Comments-oriented document summarization: understanding documents with readers' feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, New York, NY, USA, 291–298.

[24] Thorsten Joachims. 1999. Transductive Inference for Text Classification using Support Vector Machines. In *Proceedings of the 16th International Conference on Machine Learning.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 200–209.

[25] Daniel Jurafsky and James H. Martin. 2009. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition* (2 ed.). Pearson Prentice Hall, Englewood Cliffs, New Jersey, USA.

[26] Wee Sun Lee and Bing Liu. 2003. Learning with Positive and Unlabeled Examples Using Weighted Logistic Regression. In *Proceedings of the 20th International Conference on Machine Learning.* AAAI Press, Washington, DC, USA, 448–455.

[27] Xiaoli Li and Bing Liu. 2003. Learning to Classify Texts Using Positive and Unlabeled Data. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 587–594.

[28] Xiaoli Li, Bing Liu, and See-Kiong Ng. 2007. Learning to Identify Unexpected Instances in the Test Set. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence.* AAAI Press, Washington, DC, USA, 2802–2807.

[29] Xiaoli Li, Bing Liu, and See-Kiong Ng. 2010. Negative Training Data Can be Harmful to Text Classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics, Stroudsburg, PA, USA, 218–228.

[30] Xiaoli Li, Philip S. Yu, Bing Liu, and See-Kiong Ng. 2009. Positive Unlabeled Learning for Data Stream Classification. In *Proceedings of the SIAM International Conference on Data Mining.* SIAM, Philadelphia, PA, USA, 257–268.

[31] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. 2003. Building Text Classifiers Using Positive and Unlabeled Examples. In *Proceedings of the 3rd IEEE International Conference on Data Mining.* IEEE Computer Society, Washington, DC, USA, 179–188.

[32] Bing Liu, Wee Sun Lee, Philip S. Yu, and Xiaoli Li. 2002. Partially Supervised Classification of Text Documents. In *Proceedings of the 19th International Conference on Machine Learning.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 387–394.

[33] Yan Liu, Alexandru Niculescu-Mizil, and Wojciech Gryc. 2009. Topic-link LDA: joint models of topic and author community. In *Proceedings of the 26th International Conference on Machine Learning.* ACM, New York, NY, USA, 665–672.

[34] Yue Lu and Chengxiang Zhai. 2008. Opinion integration through semi-supervised topic modeling. In *Proceedings of the 17th International World Wide Web Conference.* ACM, New York, NY, USA, 121–130.

[35] Zongyang Ma, Aixin Sun, Quan Yuan, and Gao Cong. 2012. Topic-driven reader comments summarization. In *Proceedings of the 21st ACM international conference on Information and knowledge management.* ACM, New York, NY, USA, 265–274.

[36] Ramesh Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. 2008. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM International Conference on Knowledge Discovery in Data Mining.* ACM, New York, NY, USA, 542–550.

[37] David Newman, Chaitanya Chemudugunta, and Padhraic Smyth. 2006. Statistical entity-topic models. In *Proceedings of the 12th ACM International Conference on Knowledge Discovery in Data Mining.* ACM, New York, NY, USA, 680–686.

[38] Minh Nhut Nguyen, Xiaoli Li, and See-Kiong Ng. 2011. Positive Unlabeled Leaning for Time Series Classification. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence.* AAAI Press, Washington, DC, USA, 1421–1426.

[39] Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom M. Mitchell. 1998. Learning to Classify Text from Labeled and Unlabeled Documents. In *Proceedings of the 15th National Conference on Artificial Intelligence.* AAAI Press, Washington, DC, USA, 792–799.

[40] Xuan Hieu Phan, Minh Le Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th International World Wide Web Conference.* ACM, New York, NY, USA, 91–100.

[41] Daniel Ramage, Susan T. Dumais, and Daniel J. Liebling. 2010. Characterizing Microblogs with Topic Models. In *Proceedings of the 4th International Conference on Weblogs and Social Media.* AAAI Press, Washington, DC, USA, 130–137.

[42] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics, Stroudsburg, PA, USA, 248–256.

[43] Bernhard Schölkopf, John Platt, John Shawe-Taylor, Alex Smola, and Robert Williamson. 2001. Estimating the Support of a High-Dimensional Distribution. *Neural Computation* 13, 7 (2001), 1443–1471.

[44] Dyut Kumar Sil, Srinivasan H. Sengamedu, and Chiranjib Bhattacharyya. 2011. ReadAlong: reading articles and comments together. In *Proceedings of the 20th International World Wide Web Conference (Companion Volume)*. ACM, New York, NY, USA, 125–126.

[45] Dyut Kumar Sil, Srinivasan H. Sengamedu, and Chiranjib Bhattacharyya. 2011. Supervised matching of comments with news article segments. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management*. ACM, New York, NY, USA, 2125–2128.

[46] Ruben Sipos and Thorsten Joachims. 2013. Generating Comparative Summaries from Reviews. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*. ACM, New York, NY, USA, 1853–1856.

[47] Jie Tang, Zhanpeng Fang, and Jimeng Sun. 2015. Incorporating Social Context and Domain Knowledge for Entity Recognition. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, New York, NY, USA, 517–526.

[48] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. 2012. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM International Conference on Knowledge Discovery in Data Mining*. ACM, New York, NY, USA, 1285–1293.

[49] Jie Tang, Limin Yao, and Dewei Chen. 2009. Multi-topic Based Query-Oriented Summarization. In *Proceedings of the SIAM International Conference on Data Mining*. SIAM, Philadelphia, PA, USA, 1147–1158.

[50] Jie Tang and Jing Zhang. 2009. A Discriminative Approach to Topic-Based Citation Recommendation. In *Proceedings of the 13th Pacific-Asia Knowledge Discovery and Data Mining*. Springer, Heidelberg, Berlin, Germany, 572–579.

[51] Chong Wang, David M. Blei, and Fei-Fei Li. 2009. Simultaneous image classification and annotation. In *Proceedings of 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Washington, DC, USA, 1903–1910.

[52] Xuanhui Wang, ChengXiang Zhai, Xiao Hu, and Richard Sproat. 2007. Mining correlated bursty topic patterns from coordinated text streams. In *Proceedings of the 13th ACM International Conference on Knowledge Discovery in Data Mining*. ACM, New York, NY, USA, 784–793.

[53] Xiang Wang, Kai Zhang, Xiaoming Jin, and Dou Shen. 2009. Mining common topics from multiple asynchronous text streams. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*. ACM, New York, NY, USA, 192–201.

[54] Zi Yang, Keke Cai, Jie Tang, Li Zhang, Zhong Su, and Juanzi Li. 2011. Social context summarization. In *Proceedings of the 34th International Conference on Research and Development in Information Retrieval*. ACM, New York, NY, USA, 255–264.

[55] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing Twitter and Traditional Media Using Topic Models. In *Proceedings of the 33rd European Conference on Information Retrieval*. Springer, Heidelberg, Berlin, Germany, 338–349.

Received March 2016; revised October 2016, March 2017; accepted March 2017