Interpretable Aspect-Aware Capsule Network for Peer Review Based Citation Count Prediction

SIQING LI, Renmin University of China
YALIANG LI, Alibaba Group
WAYNE XIN ZHAO, Renmin University of China
BOLIN DING, Alibaba Group
JI-RONG WEN, Renmin University of China

11

Citation count prediction is an important task for estimating the future impact of research papers. Most of the existing works utilize the information extracted from the paper itself. In this article, we focus on how to utilize another kind of useful data signal (i.e., peer review text) to improve both the performance and interpretability of the prediction models.

Specially, we propose a novel aspect-aware capsule network for citation count prediction based on review text. It contains two major capsule layers, namely the feature capsule layer and the aspect capsule layer, with two different routing approaches, respectively. Feature capsules encode the local semantics from review sentences as the input of aspect capsule layer, whereas aspect capsules aim to capture high-level semantic features that will be served as final representations for prediction. Besides the predictive capacity, we also enhance the model interpretability with two strategies. First, we use the topic distribution of the review text to guide the learning of aspect capsules so that each aspect capsule can represent a specific aspect in the review. Then, we use the learned aspect capsules to generate readable text for explaining the predicted citation count. Extensive experiments on two real-world datasets have demonstrated the effectiveness of the proposed model in both performance and interpretability.

CCS Concepts: • **Information systems → Data mining;**

Additional Key Words and Phrases: Citation count prediction, peer review, capsule network

This work was partially supported by National Natural Science Foundation of China under Grant No. 61872369 and 61832017, Beijing Academy of Artificial Intelligence (BAAI) under Grant No. BAAI2020ZJ0301, Beijing Outstanding Young Scientist Program under Grant No. BJJWZYJH012019100020098, the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China under Grant No. 18XNLG22 and 19XNQ047. This work was also partially supported by Alibaba Group through Alibaba Innovative Research Program.

Authors' addresses: S. Li, Beijing Key Laboratory of Big Data Management and Analysis Methods, Renmin University of China, Beijing, 100872, China; email: lisiqing@ruc.edu.cn; Y. Li, Alibaba Group, 205 108th Ave NE, Suite 400, Bellevue, WA 98004, USA; email: yaliang.li@alibaba-inc.com; W. X. Zhao (corresponding author), Gaoling School of Artificial Intelligence, Beijing Key Laboratory of Big Data Management and Analysis Methods, Renmin University of China, Beijing, 100872, China; email: batmanfly@gmail.com; B. Ding, Alibaba Group, 205 108th Ave NE, Suite 400, Bellevue, WA 98004, USA; email: bolin.ding@alibaba-inc.com; J.-R. Wen, jrwen@ruc.edu.cn, Gaoling School of Artificial Intelligence, Beijing Key Laboratory of Big Data Management and Analysis Methods, Renmin University of China, Beijing, 100872, China; emails: jirong.wen@ruc.edu.cn.

1046-8188/2021/11-ART11 $15.00
https://doi.org/10.1145/3466640

ACM Reference format:
Siqing Li, Yaliang Li, Wayne Xin Zhao, Bolin Ding, and Ji-Rong Wen. 2021. Interpretable Aspect-Aware Capsule Network for Peer Review Based Citation Count Prediction. *ACM Trans. Inf. Syst.* 40, 1, Article 11 (November 2021), 29 pages.
https://doi.org/10.1145/3466640

1 INTRODUCTION

Today, the number of research papers increases at an unprecedented growth rate. How to quickly identify important and influential ones from a large corpus of research papers? One of the useful criteria is the *citation count*, which indicates how many other research papers have cited this one. However, except for a small number of pioneer work, for most papers, the citation count needs a long time to accumulate.

To estimate the potential impact of a paper at its early stage, the task of citation count prediction [7] emerges, which aims to predict a paper's citation count once the paper is published. Most of the existing work about citation count prediction utilizes the information extracted from the paper itself. In early studies, some researchers made the first attempt to predict citation count by utilizing authors' information such as the *h*-index, the information of coauthors and the past publishing history of authors [4, 7, 10, 76]. Later on, some researchers investigated the usefulness of content such as the title, the abstract, and the main body of the paper [10, 27, 30]. Recently, time series based methods have been adopted to predict the citation count by considering the temporal features derived from the past citation count information [20, 50, 70].

The aforementioned existing methods neglect another important piece of information for citation count prediction: the information generated during the peer review process. The authors of research papers tend to discuss their advantages and claim contributions while ignoring the shortcomings of their own work. Thus, it is necessary to adopt the review text as the complementary information for citation count prediction. For example, we show two NeurIPS papers in Table 1. From it, we can clearly see that with the help of review text, it becomes easier to predict that the first one will have a high citation count while the second one seems not, and these predictions can be difficult if we only use the information from papers. Today, more and more conferences, such as ICLR[1] and NeurIPS,[2] open their review process, and the review comments can be accessed by the public. This trend brings a great opportunity to the citation count prediction task, and it is valuable to study how to utilize such external information.

Recall the process in which we evaluate the quality of a research paper: several individuals who are familiar with the studied topic will be invited to write comments toward this paper, followed by a discussion led by a meta-reviewer. During such peer review process, each reviewer evaluates the quality of a paper from several *aspects* (e.g., originality, technical novelty, and evaluation) or evaluation dimensions, then discusses with other ones to align their rating scores, and finally achieves an agreement that can be summarized into a meta-review comment.

As we can see, peer review text (in short as *review text*) is a mixture of content in multiple aspects, and is likely to contain minor or supplementary comments, such as spelling errors and suggested references. Intuitively, different aspects have varying importance levels on assessing the contributions of a paper. For example, experimental evaluation is the first priority factor for an empirical study paper, whereas technical novelty is more important to consider for a methodology-based paper. To fully utilize review text for our task, it is essential to effectively

[1] https://openreview.net/group?id=ICLR.cc.
[2] https://nips.cc/.

Table 1. Examples of Two NeurIPS Papers (with Citation Count 19,008 and 36, Respectively) Showing the Necessity of Considering Review Texts

ID	Paper Abstract	Review Comments
❶	...In this paper, we present several improvements that make the Skip-gram model more expressive and enable it to learn higher quality vectors more rapidly...	...This is a good paper. The part on speeding up learning justifies the paper. The work presented is significant enough to be accepted in NIPS and shared among other researchers in the area...
❷	...the experimental results demonstrate that the proposed learning approach outperforms a number of comparison cross language representation learning methods...	...while this seems like a good idea, I'm not sure that the results are robust. I would urge the authors to use an ML system that is insensitive to rescalings of individual feature...

extract adaptive aspect semantics from mixed content and reduce the influence of less useful information.

In this article, we propose a novel aspect-aware capsule network for citation count prediction based on review text. The core idea is to utilize the excellent learning capacity of capsule networks to extract useful aspect semantics, called *aspect capsules*, from review text for our task. Instead of directly learning aspect capsules, we start with a more fine-grained kind of semantic representations, called *feature capsules*, for capturing local semantics for sentence segments. Furthermore, we propose an abstract routing mechanism in the feature capsule layer to extract more important and useful semantic information from the review text by referring to abstract text, which summarizes the core contributions of a paper. After feature capsules are learned, we further adopt a dynamic routing mechanism to derive aspect capsules by extracting corresponding aspect semantics from feature capsules. By adopting such a hierarchical representation architecture, our model is able to gradually extract useful semantic characteristics from fine-grained units to high-level aspects.

Besides performance, interpretability is another important factor to consider in predictive models for explaining the learned models and prediction results. Although the proposed capsule network is capable of learning complex data characteristics, it is difficult to explain the underlying working mechanism, such as what has been encoded by an aspect capsule. To address this issue, our idea is to utilize review text information to enhance the model interpretability, since it is readable and contains important assessment comments on the contribution or weakness of a paper. First, we utilize unsupervised topic models to extract meaningful topics (considered as *aspects*) from review text. We use the topic distribution of the review text to guide the learning of aspect capsules so that each aspect capsule can represent a specific aspect in the review. Then, we also use the learned aspect capsules to generate readable text for explaining the predicted citation count.

To validate the effectiveness of the proposed method, we conduct a series of experiments on two real-world datasets collected from ICLR and NeurIPS conferences. Experimental results under different metrics confirm that the proposed method outperforms the state-of-the-art methods for citation count prediction. We further demonstrate that the proposed method has a better interpretability by quantifying the review aspect coverage, examining the correlation to citation count, and showing qualitative examples of generated review summarization.

To summarize, the main contributions of this article are the following:

- We propose a novel aspect-aware capsule network for citation count prediction based on review text. By learning both feature capsules and aspect capsules, we explore the aspect-level representation for extracting useful semantic characteristics from review text.
- We propose two strategies to enhance the interpretability of our proposed model. By aligning to the learned topics, the aspect capsules are easier to understand. We also use the learned aspect capsules to generate readable text for explaining the predicted citation count.
- We conduct extensive experiments on real-world datasets to validate the effectiveness of the proposed method, and demonstrate its advantages in terms of performance and interpretability.

2 RELATED WORK

In this section, we discuss related work from the following three perspectives that are close to our work: the citation count prediction task, review text modeling, and an aspect-based sentiment analysis and capsule network for aspect extraction.

2.1 Citation Count Prediction

For the citation count prediction task, various methods have been proposed by utilizing different kinds of features. In the early stage, researchers regard this task as a regression task or a classification task. They investigate hand-crafted features such as author information [7], the past citation count data [30], and temporal and topological features [20]. Among them, Davletov et al. [20] cluster the temporal changes in the number of citation counts and use a regression model to the predict citation count given the early citation performance of a paper. With the development of natural language processing technology, some researchers propose to learn the information from text, such as title, abstract, and main body text [10, 27, 30]. Among these works, Chen and Zhang [10] utilize six content features and 10 author features as input. They introduce Gradient Boosted Regression Trees to predict the citation count. Fu and Aliferis [27] use a mixture of content-based and bibliometric features with machine learning methods to predict the long-term impact of publication. Following the previous research, some researchers began to formally investigate the factors that are useful for citation count prediction [4, 10, 76]. Among these works, Yan et al. [76] utilize various features of fundamental characteristics for papers and consider several regression models to conduct experiments. Bhat et al. [4] model the effects of each author's influence in coauthorship graphs and words in the title of the paper. They build classifiers to predict the interval in which a paper's citation count will be. Furthermore, as some of the conferences have opened access to the peer review texts, recent work has begun to investigate how to utilize the external information from review to predict the citation count [40]. However, they only learn the document-level representation from sentences, not modeling the fine-grained aspect-level information and also lacking the interpretability. In addition, Zhao [87] directly uses the structural and temporal information of the citation network formed in the early stage to predict the citation count. Van Dongen et al. [64] use a pre-trained **Bidirectional Encoder Representations from Transformers (BERT)** model to extract features and a deep learning model to learn from the features and predict the citation count. But they do not model the interaction between abstract and review and the aspect information.

2.2 Review Text Modeling

Peer review, widely adopted in various journals and conferences, is an important paper evaluation mechanism [26, 42, 52]. In the early stage, researchers have studied the usefulness of peer review

based on private review data in terms of issue localization [72], review utility [71], and quality/tone [51]. Due to the limitation of access to the review of scholarly papers, the related research works on the modeling of peer review are rare.

However, research works on a similar task of predicting the quality of products using review text are relatively rich [8, 17, 28, 35, 36, 39, 47, 60, 63, 68, 69, 79, 82–86]. The review text modeling in our task is related to these works. Among them, several works use reviews to improve the explainability of recommender systems [8, 17, 36, 60, 63]. Some other research works extract information from user reviews and integrate them with ratings to improve the recommendation performance [28, 35, 39, 68], whereas other researchers explore under which scenarios the review text can be useful for recommender systems [1, 9, 25, 48, 57, 72]. In addition, Zhang et al. [84] use product reviews to detect the causal relationship between technology evolution and social life. Recently, some researchers have explored review text mining at the aspect level [2, 28, 31, 36, 43, 45, 58, 62, 68, 79, 85]. Among them, Guan et al. [28], Yuan et al. [79], and Zhao et al. [85] adopt attentive models to extract aspects from user reviews. Jiang et al. [31] propose a graph representation learning method to transfer and detect a cross-domain aspect category. Mukherjee et al. [45] and Tian et al. [62] use a generation task to extract aspect from reviews. Shi et al. [58] explore the content of reviews by extracting latent topics related to different aspects with unsupervised topic modeling techniques. However, they mainly focus on product reviews in recommender systems and doctor reviews in healthcare systems, which have different patterns from peer reviews of scholarly papers.

Recently, a public dataset of peer reviews has been released to lower the barrier to studying peer reviews for the scientific community [33]. Based on this dataset, some research works on modeling peer review text have emerged [40, 65]. Wang and Wan [65] leverage peer review to predict the overall decision status for paper submissions. Li et al. [40] first use peer review text to predict the citation count of scholarly papers. Different from our work, they model the review text at the document and sentence level, ignoring the fine-grained aspect information.

2.3 Aspect-Based Sentiment Analysis

Topic modeling has been widely applied to aspect-based sentiment analysis [6, 32, 74, 75, 90], which can learn coherent semantic topics from online review text [18, 81]. These learned topics are usually called *aspects* [77, 90], which correspond to different attributes or dimensions about a product or service. Most of the studies developed their approach by extending **Latent Dirichlet Allocation (LDA)** [5]. Among these works, Brody and Elhadad [6] proposed an unsupervised local topic model technique. They treated each sentence as a document and applied standard LDA on each sentence to obtain the aspects. Zhao et al. [90] proposed a hybrid topic-based semi-supervised model, which incorporates maximum entropy along with topic modeling to identify aspects and opinions. Xu et al. [74] proposed a joint aspect/sentiment model to extract aspects from reviews and generated the aspect-dependent sentiment lexicons. They extended LDA to extract aspects and opinion words, and further utilized the opinion words to extract the implicit aspects. In the domain of product reviews, knowledge-based topic modeling has also been proposed to extract aspects by using shared knowledge [12–14].

Recent studies along this line mainly adopted the **Long Short-Term Memory (LSTM)** model [53, 61]. They utilized LSTM cells to encode sentence information and capture the relatedness of target words with context words. For example, Wang et al. [66] proposed an attention-based LSTM for aspect-level sentiment classification. The attention mechanism can concentrate on different parts of a sentence when different aspects are taken as input. Liu and Zhang [41] discriminated between left and right contexts given a certain target following previous work. More recently, BERT-based methods that utilized pretraining techniques have achieved significant performance improvement [49, 59, 73]. Xu et al. [73] constructed an auxiliary sentence from the aspect and

converted the aspect-based sentiment analysis task to a sentence-pair classification task. They fine-tuned the pre-trained model from BERT and achieved new state-of-the-art results.

Compared with these studies, our work has different task settings and technical challenges. Sentiment analysis is usually associated with a specific user-product pair, whereas citation count prediction is performed in terms of a single paper. In addition, we mainly focus on modeling the semantic interaction between review text and paper content, where such interaction has seldom been modeled in sentiment analysis studies.

2.4 Capsule Network for Aspect Extraction

The capsule network is usually used for aspect extraction, and it has been proposed to reform the convolutional neural network by using vectors to represent a cell so that it can extract rich information [29, 55, 56]. Hinton et al. [29] propose the capsule networks that can be used to learn features in the form of a set of vectors, which is a much more promising way of dealing with various information than the methods currently employed in the neural networks community. It is first used in the computer vision area by utilizing a special routing approach instead of pooling operations so that the capsules can capture high-level features. Recently, researchers have investigated that the capsule network is also useful in natural language processing [11, 23, 78, 80, 88]. Among these works, Zhao et al. [88] discuss the development of capsule networks for challenging NLP applications. Chen et al. [11] use capsule networks to capture argument interaction in semantic role labeling. Yang et al. [78] propose a query-guided capsule network to cluster context information into different perspectives from which the target translation may concern. Du et al. [23] adopt capsule networks to construct the vectorized representation of semantics and propose a novel dynamic routing mechanism to select the proper sense for the downstream classification task. More importantly, the capsule network can deal with the problem of aspect extraction [15, 24, 36, 67]. Du et al. [24] propose to use the capsule network to construct vector-based feature representation and model the relationship between aspect terms and context by the capsule routing procedure. Chen and Qian [15] propose a Transfer Capsule Network model for transferring document-level knowledge to aspect-level sentiment classification. Li et al. [36] use capsules to extract the viewpoints and aspects from the user and item, and review documents to improve rating prediction with explanation. Different from these works, we propose a novel abstract routing approach to extracting the aspect-aware information from review text and abstract text.

3 TASK DEFINITION

We first formally define the citation count prediction task. Let d denote a research paper from a literature corpus $\mathcal{D}$ and $\boldsymbol{x}_d^a$ denote its abstract text. Usually, a paper will be assigned to multiple reviewers, and we combine all the reviewers' comments into a single document, denoted by $\boldsymbol{x}_d^r$. Other useful information of the paper d, such as authors' h-index, is denoted as $\boldsymbol{x}_d^o$. For the task of citation count prediction, we aim to learn an effective predictive function to predict the citation count of a research paper.

To be more specific, we take the review text $\boldsymbol{x}_d^r$, abstract text $\boldsymbol{x}_d^a$, and other available information $\boldsymbol{x}_d^o$ as the input, learn the aspect-level representations of the review texts, and then predict the future citation count after a given time period t as follows:

$$f\left(\boldsymbol{x}_d^r, \boldsymbol{x}_d^a, \boldsymbol{x}_d^o\right) \rightarrow \hat{c}_d, \tag{1}$$

where $\hat{c}_d$ is the predicted citation count. The loss function of the learning procedure can be defined through the mean squared error as

$$L_{pred} = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} (\hat{c}_d - c_d)^2, \tag{2}$$

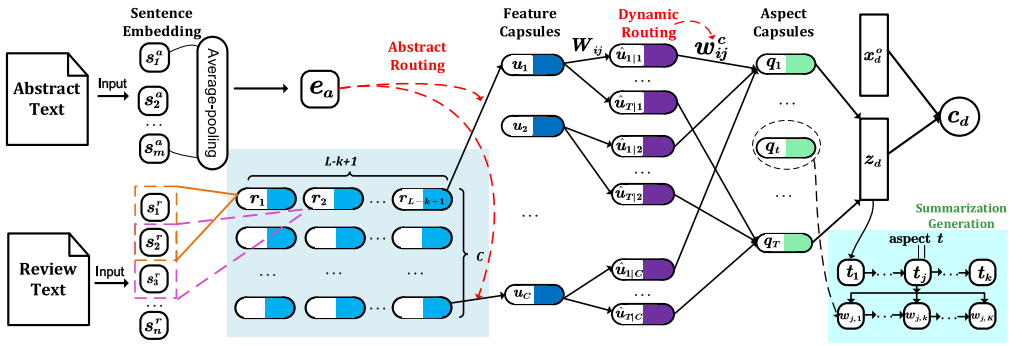

Fig. 1. Overview of the proposed method. The blue capsules denote feature capsules. The green capsules denote aspect capsules. The red lines correspond to two routing approaches. The summarization generation module is on the bottom right corner.

where c_d is the ground-truth citation count of the paper d. Here, following previous studies [4, 10, 27, 40, 64, 76], the ground-truth citation count c_d is accumulated after a fixed-length time period (e.g., 3 to 5 years). In this way, the increasing trend of the citation count would become relatively stable so that we can make relatively reliable predictions about future citation counts received in the given period. Another interesting direction is to make dynamic prediction for the citation count. However, in most scenarios, we are more concerned about the accumulative citation count in the future. We will leave this problem as future work.

4 METHODOLOGY

In this section, we present the proposed aspect-aware capsule neural network for citation count prediction based on peer review text. The overview of the proposed method is illustrated in Figure 1.

The main components of the proposed method are designed to learn useful characteristics from review text for our prediction task. It contains two capsule layers, namely the feature capsule layer and the aspect capsule layer, with two different routing approaches, respectively. Feature capsules are able to encode the local semantics from review sentences and further form aspect capsules for capturing high-level semantic representations. Besides the predictive capacity, we also enhance the interpretability of the proposed model with two strategies. First, we use the topic distribution of the review summarization text to guide the learning of aspect capsules so that each aspect capsule can represent a specific aspect in the review. We also use the learned aspect capsules to generate an explainable text for the predicted citation count, which can improve the interpretability of the proposed method. The notations and the descriptions are shown in Table 2.

4.1 Learning Feature Capsules with Abstract Routing

In this section, we study how to learn feature capsules, which will be served as basic representations for deriving aspect capsules.

4.1.1 Sentence Embedding. The inputs of our model are the word sequences of review text $\boldsymbol{x}_d^r$ and abstract text $\boldsymbol{x}_d^a$. We first use the word2vec model [44] to pretrain a word embedding lookup table, with which we map the word sequences of review text and abstract text to a list of word vectors, respectively. Then we use the convolutional neural network [34] to extract high-level representations of the sentences in review text and abstract text. We use multiple convolutional filters to produce a feature map, and adopt a max-pooling operation to obtain the representations of sentences in review text and abstract text, denoted as $\{\boldsymbol{s}_1^r, \ldots, \boldsymbol{s}_i^r, \ldots, \boldsymbol{s}_n^r\} \in \mathbb{R}^{d_s \times n}$ and $\{\boldsymbol{s}_1^a, \ldots, \boldsymbol{s}_i^a, \ldots, \boldsymbol{s}_m^a\} \in$

Table 2. Notations Used in This Article

Notation	Description
$\boldsymbol{x}_d^o$	Concatenation of the wide features of a paper d
$\boldsymbol{w}_{deep}^\top$	Parameter vectors for the deep component
$\boldsymbol{w}_{wide}^\top$	Parameter vectors for the wide component
b_o	Bias term
$\boldsymbol{z}_d$	Final review representation of a paper d
$\hat{\boldsymbol{c}}_d$	Predicted citation count of a paper d
$\boldsymbol{s}_i^r$	Learned representations of the i-th sentence in the review text
$\boldsymbol{s}_i^a$	Learned representations of the i-th sentence in the abstract text
d_s	Dimension of the sentence vectors
n	Number of sentences in each review text
m	Number of sentences in each abstract text
$\boldsymbol{r}_i$	Feature vector of the i-th n-gram in review sentences
$\boldsymbol{R}_i$	Feature capsules in the i-th feature space
$\boldsymbol{W}_r^{(j)}$	The j-th kernel matrix in the kernel group
v_i^a	Abstract routing weight
$\widetilde{\boldsymbol{s}}_a$	Average pooling of the sentence embeddings of abstract text
$\boldsymbol{P}$	Representation of routed feature capsules
$\boldsymbol{u}_i$	Final representation of feature capsules
k	Size of the sliding window
d_p	Dimension of feature capsules
d_h	Number of neurons for each row in the kernel
$\hat{\boldsymbol{u}}_{j\|i}$	Prediction vector generated by the i-th feature capsule toward the j-th aspect capsule
w_{ij}^c	Coupling coefficient defined by routing softmax
$\widetilde{\boldsymbol{q}}_j$	Vector representation of aspect capsules
$\boldsymbol{q}_j$	Final representation of aspect capsules
d_c	Dimension of aspect capsules
$\mathcal{T}$	Set of topics
$\boldsymbol{w}_j$	Aspect-specific parameter vector
$\pi_{d,j}$	Importance of the j-th capsule
$\boldsymbol{\theta}_d$	Multinomial distribution over the topic
$\boldsymbol{\phi}_t$	Multinomial distribution over the vocabulary learned with the topic model
t_j	Generated j-th aspect label
$w_{j,k}$	Generated k-th word in the j-th sentence

$\mathbb{R}^{d_s \times m}$, respectively, where d_s denotes the dimension of the sentence vector, n denotes the number of sentences in the review text, and m denotes the number of sentences in the abstract text.

4.1.2 Initial Feature Capsules. Based on sentence representations, we construct the feature capsule layer to learn important semantics for the local text window consisting of several sentences.

Similar to the operation applied to sentence vectors, we also apply multiple convolution operations to the k-sentence window in review text and derive its feature vector $\boldsymbol{r}_i = \{r_{i,j}\}_{j=1}^{d_p}$. Each element $r_{i,j}$ in $\boldsymbol{r}_i$ is computed as follows:

$$r_{i,j} = \tanh\left(\boldsymbol{s}^r_{i:i+k} \otimes \boldsymbol{W}_r^{(j)} + b_r\right), \tag{3}$$

where $\boldsymbol{W}_r^{(j)} \in \mathbb{R}^{d_h \times k}$ is the j-th kernel matrix in the kernel group, $d_h \times k$ is the kernel size, k is the size of the sliding window, d_h is the number of neurons for each row in the kernel, d_p is the dimensionality of feature capsules, b_r is the bias, and "$\otimes$" is the matrix operator that sums the results of the element-wise product between two equal-sized matrices as $\boldsymbol{A} \otimes \boldsymbol{B} = \sum_{i,j} a_{i,j} \cdot b_{i,j}$.

In our model, one kernel group corresponds to a specific kind of semantic mapping. We repeat the preceding procedure C times with different kernel groups so that we can derive multiple channels of feature capsules corresponding to C kinds of semantic meanings. Finally, the initial feature capsules learned from review text are denoted as $\mathbf{R}^* = [\boldsymbol{R}_1, \boldsymbol{R}_2, \ldots, \boldsymbol{R}_C] \in \mathbb{R}^{C \times d_p \times (L-k+1)}$.

4.1.3 Abstract Routing. Intuitively, some content in review texts such as comments on spelling and minor points might not be useful for the citation count prediction task. Here, our idea is to utilize the abstract text of the paper to learn the importance of different review features, as the abstract content of a paper is a summary of its core contributions. To this end, we propose an abstract routing mechanism to computing the importance of content from review text. Formally, we apply a fusing convolution operation to the sentence embedding of review text with a new kernel $\boldsymbol{W}_r \in \mathbb{R}^{d_h \times k}$, and we can derive the abstract routing weight:

$$v_i^a = \text{sigmoid}\left(\boldsymbol{s}^r_{i:i+k} \otimes \boldsymbol{W}_r + \widetilde{\boldsymbol{s}}_a^\top \cdot \boldsymbol{w}_a + b_a\right), \tag{4}$$

where $\widetilde{\boldsymbol{s}}_a$ is the average-pooling vector of the sentence embeddings of abstract text (i.e., $\widetilde{\boldsymbol{s}}_a$ is obtained by average-pooling over $\{\boldsymbol{s}_1^a, \ldots, \boldsymbol{s}_i^a, \ldots, \boldsymbol{s}_m^a\}$), $\boldsymbol{w}_a$ is a vector to map $\widetilde{\boldsymbol{s}}_a$ to a scalar value, and b_a is the bias.

In this way, the computed routing weight $v_i^a \in [0, 1]$ reflects the importance of features from the review text by referring to the core contributions (i.e., the abstract text) of the paper. Thus, the initial feature capsules can be further routed according to the weights:

$$\mathbf{P} = \mathbf{R}^* \odot \mathbf{V}^a, \tag{5}$$

where "$\odot$" is the element-wise multiplication, $\mathbf{V}^a \in \mathbb{R}^{C \times d_p \times (L-k+1)}$ is formed by replicating $[w_1^a, w_2^a, \ldots, w_{L-k+1}^a]$ for $C \times d_p$ times, and $\mathbf{P} \in \mathbb{R}^{C \times d_p \times (L-k+1)}$ is the abstract-routed review feature capsules.

To avoid the over-fitting caused by the large number of feature capsules, we adopt the element-wise maximum function to aggregate all feature capsules in the same channel horizontally:

$$\boldsymbol{U}_{[j,i]} = \max_{k'=1}^{L-k+1} \mathbf{P}_{[i,j,k']}, \tag{6}$$

where the subscripts denote the index placeholders in the matrix, $\boldsymbol{U} \in \mathbb{R}^{C \times d_p}$ is the feature capsule matrix, and each row corresponds to a feature capsule vector. Finally, it is our hope that the length of each feature capsule $\boldsymbol{u}_i$ can represent the probability that $\boldsymbol{u}_i$'s semantic meaning is useful for the citation count prediction task. So we use the nonlinear "squash" function to limit its length into the interval of $[0, 1]$ as follows:

$$\boldsymbol{u}_i \leftarrow \text{squash}(\boldsymbol{u}_i) = \frac{||\boldsymbol{u}_i||^2}{1 + ||\boldsymbol{u}_i||^2} \cdot \frac{\boldsymbol{u}_i}{||\boldsymbol{u}_i||}. \tag{7}$$

4.2 Learning Aspect Capsules with Dynamic Routing

Based on the preceding extracted feature capsules $\{\boldsymbol{u}_i\}_{i=1}^{C}$, we propose to use the dynamic routing approach to obtaining the *aspect capsules*. Feature capsules mainly capture local semantics from a short text window, whereas aspect capsules aim to characterize the overall aspect semantics of the review text. Here, *aspects* are expected to capture the characteristics in terms of different evaluation dimensions for a paper, such as task, methodology, and experiments.

To learn aspect capsules, we first construct the associations between feature capsules and aspect capsules. A feature capsule $\boldsymbol{u}_i$ generates a "prediction vector" $\hat{\boldsymbol{u}}_{j|i}$ toward the j-th aspect capsule through the following way:

$$\hat{\boldsymbol{u}}_{j|i} = \boldsymbol{W}_{ij}\boldsymbol{u}_i, \tag{8}$$

where $\boldsymbol{W}_{ij} \in \mathbb{R}^{d_c \times d_p}$ is a weight matrix, and d_p and d_c are the dimensions of feature capsules and aspect capsules, respectively. All of the "prediction vectors" generated by the feature capsules are then summed up with weight w_{ij}^c to obtain the initial vector representation of the aspect capsule $\widetilde{\boldsymbol{q}}_j$:

$$\widetilde{\boldsymbol{q}}_j = \sum_i w_{ij}^c \hat{\boldsymbol{u}}_{j|i}, \tag{9}$$

where w_{ij}^c is a coupling coefficient defined by a "routing softmax":

$$w_{ij}^c = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}, \tag{10}$$

where each b_{ij} is the log prior probability that the i-th feature capsule should pass to the j-th aspect capsule. It is computed by using the dynamic routing approach, and we will discuss this later.

Given the initial aspect capsules $\{\widetilde{\boldsymbol{q}}_j\}$, we further apply the nonlinear "squash" function to them and derive the final representation of aspect capsule $\boldsymbol{q}_j$ as

$$\boldsymbol{q}_j = \text{squash}(\widetilde{\boldsymbol{q}}_j). \tag{11}$$

By doing so, the length of $\boldsymbol{q}_j$ is limited to the interval of $[0, 1]$, and it can represent the score of the paper under the corresponding aspect for the citation count prediction task.

4.2.1 Dynamic Routing. The logit b_{ij} in Equation (10) determines the correlation degree between the i-th feature capsule and the j-th aspect capsule. It is initialized with zero and is updated with an agreement coefficient w_{ij}^a:

$$w_{ij}^a = \hat{\boldsymbol{u}}_{j|i} \cdot \boldsymbol{q}_j. \tag{12}$$

Then, following the work of Sabour et al. [56], the agreement coefficient is added to the current value of logit b_{ij}:

$$b_{ij} \leftarrow b_{ij} + w_{ij}^a, \tag{13}$$

which will lead to the update of all coupling coefficients in Equation (10) and the aspect capsule vectors in Equations (9) and (11). Actually, the whole dynamic routing procedure is an iterative circulation of Equation (10) → Equation (9) → Equation (11) → Equation (12) → Equation (13). The procedure is repeated for r iterations.

4.2.2 Citation Count Prediction with Aspect Capsules. We utilize the Wide & Deep framework to jointly consider the learned vector representation of review text and other features (i.e., $\boldsymbol{x}_d^o$ in Section 3) to predict the citation count, which shares some similarities with a previous study [40]. But different from it, we use the aspect capsules to derive the review representation, which can capture fine-grained semantic features for the task of citation count prediction.

For the deep component, we use the vector of aspect capsules to derive a context vector z_d as the final features of the review text:

$$z_d = \tanh(W_c Q + b_c), \tag{14}$$

where $Q = [q_1, \ldots, q_j, \ldots, q_T]$ (q_j is defined in Equation (11), and T is the number of aspect capsules), W_c is the weight matrix, and b_c is the bias vector. Here, we expect that z_d can encode necessary information of review text for paper d for citation count prediction.

Besides the learned hidden features, we construct the vector x_d^o as the wide component with several hand-crafted features including topic distribution, diversity, recency, and author influence:

- *Topic distribution*: We utilize LDA [5] to learn the probability distribution over topics from the main text of the research paper as the topic distribution features.
- *Diversity*: We calculate the entropy of the paper's topic distribution to measure the topical diversity of a paper.
- *Recency*: We use the year of publication as the temporal feature to predict the citation count.
- *Author influence*: We use the number of authors and the average h-index of these authors as author influence features.

More information can be found in implementation details (Section 5.1.2). Finally, we integrate the wide component and the deep component into a unified model and predict the citation count of paper d as

$$\hat{c}_d = \tanh\left(w_{deep}^\top \cdot z_d + w_{wide}^\top \cdot x_d^o + b_o\right), \tag{15}$$

where x_d^o is the concatenation of the wide features; w_{deep} and w_{wide} are the parameter vectors for the deep and wide components, respectively; and b_o is the bias term. With the predicted numerical value $\hat{c}_d$, we can compute the mean squared error based prediction loss in Equation (2).

4.3 Interpreting Capsule Network

Besides performance, interpretability is another important factor to consider in predictive models for explaining the learned models and predicted results. Although we have obtained multiple aspect capsules, the encoded information or the reflected semantics are difficult to explain. In this section, we study how to enhance the interpretability of the proposed capsule networks.

4.3.1 Aspect Learning with Topic Models. Review text contains reviewers' comments with different focus points on a paper, and it can be considered as a mixture of underlying aspect semantics. Here, we consider applying topic models [5] to extract underlying aspects semantics from review text. A major merit of topic models is that it provides an unsupervised way for aspect learning. Especially, it is usually able to produce meaningful and coherent topics. Here, we adopt a variant of topic models that are able to capture sentence-level coherence (i.e., Twitter-LDA [89]).

Let $\mathcal{T}$ denote a set of topics. Twitter-LDA models a document d as a multinomial distribution θ_d over the topic set $\mathcal{T}$, and a topic $t \in \mathcal{T}$ is modeled by a multinomial distribution ϕ_t over the vocabulary. Specifically, a sentence in a document would be assigned a topic label t that is sampled from θ_d. Then, the words of the sentence will be generated according to the corresponding topic-word distribution ϕ_t. To reduce the influence of high-frequency common words, Twitter-LDA also sets up background language models to adsorb the probability of background words. Given a paper d, we combine the review text from all of its reviewers into a single document. Then, we run the Twitter-LDA model over the entire review corpus. In this way, we can obtain a set of topics (i.e., topic-word distributions), which are considered as *aspects*. We can also obtain the topic distribution of the review text for a paper. These topics are expected to cover important aspects on paper assessment: originality, technical novelty, experiment, and so forth.

Indeed, various topic models can be applied to extract the underlying topics. The reason we adopt Twitter-LDA is that it can assign topic labels at the sentence level. As will be shown next, this merit is particularly useful in our task. Here, we use the terms *aspects* and *topics* in an interchanged way. We mainly follow the usage of *aspects* in aspect-based sentiment analysis [6, 32, 74, 90], where they also learn meaning topics as aspects from product-oriented review text.

4.3.2 Explanation with Aspect Alignment. In this section, we would like to align aspect capsules with the learned topics to explain the underlying semantics that they encode. We set the topic number to be the same as the number of aspect capsules. In this way, we can form a one-to-one mapping between an aspect capsule and a topic.

However, the two parts are learned in isolated models, and we have to incorporate specific objective loss to force the semantic alignment. Recall that we have learned a set of aspect capsules $\{\boldsymbol{q}_j\}$. The length of an aspect capsule reflects its importance or relatedness for a paper. With these aspect capsules, we can indeed compute an importance distribution $\boldsymbol{\pi}_d$ for a paper d, where each entry $\pi_{d,j}$ indicates the importance of the j-th capsule:

$$\pi_{d,j} = \text{softmax}\left(\boldsymbol{q}_j^\top \cdot \boldsymbol{w}_j\right), \tag{16}$$

where $\boldsymbol{w}_j$ is the aspect-specific parameter vector reflecting the overall preference of the j-th aspect capsule. Our idea is that if aspects capsules are aligned to topics, the two distributions $\boldsymbol{\pi}_d$ (learned with capsule network) and $\boldsymbol{\phi}_d$ (learned with topic models) should be similar. To formalize this idea, we incorporate a second loss for our model by minimizing the Kullback-Leibler divergence between the two distributions:

$$L_{aspect} = \frac{1}{|\mathcal{D}|}\text{KL}(\boldsymbol{\pi}_d, \boldsymbol{\phi}_d). \tag{17}$$

This loss drives aspect capsules to be aligned to topics learned with topic models. With aspect alignment, we can associate aspect capsules with more interpretable topics (i.e., topic-word distributions).

4.3.3 Explanation with Aspect-Based Review Summarization. Besides the interpretability of each individual aspect capsule, we also would like to understand how the overall model architecture works—that is to say, why are we giving a high or low numerical prediction for citation count. The basic idea is to utilize the learned capsule network to generate text for explaining the predicted citation count. If it is able to produce high-quality explainable text, it should indicate that the model itself has good interpretability in the prediction.

However, in practice, it will be difficult to obtain a ground-truth explanation for the citation count of a paper. Here, we notice a kind of rich surrogate text (i.e., meta-review) available for most conference or journal papers. Note that the opinions reflected by meta-reviews may not be consistent with citation count trends. However, meta-reviews are able to summarize the overall contributions or weakness of a paper, which are potentially useful for explaining the impact of a paper.

Recall that the major model representations learned through capsule networks include the set of aspect capsules $\{\boldsymbol{q}_j\}$ and the overall representation $\boldsymbol{z}_d$. We utilize these representations to construct a text generation model. Especially, we adopt an aspect-aware two-stage generation approach [38]. First, an aspect sequence is generated, and each aspect label controls the generation of a sentence. Here, we adopt the Twitter-LDA [89] to sample an aspect sequence for the meta-review text. Second, a sentence is further generated according to the corresponding aspect label, and the process is repeated until all the sentences are generated.

Formally, for the aspect sequence $t^{1:k} = \langle t_1, \ldots, t_i, \ldots, t_k \rangle$, we generate the j-th aspect label by using a GRU-based decoder:

$$t_j \leftarrow \text{GRU}_{dec}(t_{1:j-1}; \boldsymbol{z}_d), \tag{18}$$

where $\text{GRU}_{dec}(\cdot)$ is implemented with a standard GRU network [19] with a softmax function over the aspect set. The key point is that we set the initial state to be $\boldsymbol{z}_d$. Since $\boldsymbol{z}_d$ denotes the overall representation for a paper from a capsule network, it should be useful to infer the aspect sequence. Then, we construct a similar decoder for generating the word sentence. Given aspect label for the j-th sentence, we generate its k-th word $w_{j,k}$ as follows:

$$w_{j,k} \leftarrow \text{GRU}_{dec}(w_{j,1:k-1}; \boldsymbol{q}_{t_j}, \boldsymbol{z}_d), \tag{19}$$

where $w_{j,<k}$ denotes the previous tokens in the j-th sentence, and $\boldsymbol{q}_{t_j}$ is the learned t_j-th aspect capsule. Since we have aligned aspect capsules with aspects, we can utilize the aspect label to index aspect capsules. Here, we incorporate $\boldsymbol{q}_{t_j}$ for the generation of each token as the context vector, and $\boldsymbol{z}_d$ will be used to set the initial state.

To train the generation network, we further define a loss to reconstruct both aspect labels and words in the review summary:

$$L_{gen} = \frac{1}{|\mathcal{D}|}\text{CE}(\{t_{d,j}\}, \{w_{d,j,k}\}, \{\hat{t}_{d,j}\}, \{\hat{w}_{d,j,k}\}), \tag{20}$$

where CE is the cross-entropy loss, and $\{t_{d,j}\}$, $\{w_{d,j,k}\}$, $\{\hat{t}_{d,j}\}$, and $\{\hat{w}_{d,j,k}\}$ are the actual and generated aspect/word sequences for the meta-review text of paper d, respectively.

4.4 Training

To learn the model parameters for citation count prediction, we integrate the two loss terms in Equation (2) and Equation (17). The overall loss function is

$$L = \lambda \cdot L_{pred} + (1 - \lambda) \cdot L_{aspect}, \tag{21}$$

where λ is a coefficient to balance these two terms. The whole method is trained with back-propagation in an end-to-end manner. The training algorithm is presented in Algorithm 1. To validate the interpretability of the proposed method, we then fix the parameters in the citation count prediction module and optimize the parameters in the generation module with the loss function L_{gen} in Equation (20). We will describe more implementation details in Section 5.

4.5 Discussion

In this section, we analyze the effectiveness and interpretability of the proposed aspect-aware capsule network in our task.

To learn the important semantics from the review text, we design the feature capsule layer with a novel abstract routing mechanism. By utilizing the abstract text, the feature capsules can learn important semantics reflecting a paper's core contributions while ignoring minor points. Compared with previous peer review learning methods [40, 65], the proposed model takes a novel perspective to learn the fine-grained aspect-level representations. In our model, aspect information is gathered to aspect capsules with dynamic routing.

Note that we use the topic distribution learned by LDA as supervision signals to guide the learning of aspect capsules. If the aspect capsules are learned without supervision, the learning process may not be well controlled. In our experiments, it has been shown that this kind of guidance is effective in terms of both prediction and interpretability. With the topic supervision, the aspect capsules can possess the interpretable meaning and the aspect information that is necessary for prediction. And further, with the help of the learned aspect capsules, we can know the reviewer's

ALGORITHM 1: The Training Algorithm for Our Proposed Model.

Input: A literature corpus $\mathcal{D}$ containing abstract text $\boldsymbol{x}_d^a$, review text $\boldsymbol{x}_d^r$, other available information $\boldsymbol{x}_d^o$, citation count c_d, and topic distribution $\boldsymbol{\phi}_d$

Output: Aspect capsules $\boldsymbol{Q}$ and predicted citation count $\hat{\boldsymbol{c}}_d$

Use the word2vec model to pretrain a word embedding lookup table
Initialize parameters
while not convergence **do**
  for $d \in \mathcal{D}$ **do**
    Use the convolutional neural network to obtain $\{\boldsymbol{s}_1^r, \ldots, \boldsymbol{s}_i^r, \ldots, \boldsymbol{s}_n^r\}$ and $\{\boldsymbol{s}_1^a, \ldots, \boldsymbol{s}_i^a, \ldots, \boldsymbol{s}_m^a\}$
    for $i \leftarrow 1$ to C **do**
      Initialize feature capsules by Equation (3)
    end for
    Compute the abstract routing weight $\boldsymbol{V}^a$ by Equation (4)
    Obtain the abstract-routed review feature capsules **P** by Equation (5)
    Compute the element-wise maximum function by Equation (6)
    Compute squash function to obtain the feature capsules **U** by Equation (7)
    Compute $\hat{\boldsymbol{u}}_{j|i}$ by Equation (8)
    for $k \leftarrow 1$ to r **do**
      for $j \leftarrow 1$ to T **do**
        for $i \leftarrow 1$ to C **do**
          Compute w_{ij}^c by Equation (10)
          Compute $\widetilde{\boldsymbol{q}}_j$ by Equation (9)
          Compute $\boldsymbol{q}_j$ by Equation (11)
          Compute w_{ij}^a by Equation (12)
          Compute b_{ij} by Equation (13)
        end for
      end for
    end for
    Compute the context vector of the review text $\boldsymbol{z}_d$ by Equation (14)
    Integrate the wide component and deep component to predict the citation count $\hat{c}_d$ by Equation (15)
    Compute L_{pred} by Equation (2)
    Compute $\boldsymbol{\pi}_{d,j}$ by Equation (16)
    Compute L_{aspect} by Equation (17)
    Compute L by Equation (21)
  end for
  Update parameters with stochastic gradient descent
end while
return $\boldsymbol{Q} = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_j, \ldots, \boldsymbol{q}_T]$ and $\hat{c}_d, d \in \mathcal{D}$

attitude toward each aspect and utilize it to generate the summary of review text as the explanation for the predicted citation count.

5 EXPERIMENTS

In this section, we conduct experiments on two real-world datasets to validate the effectiveness of the proposed method for citation count prediction from the following perspectives. First, by comparing with several state-of-the-art methods, we confirm the advantage of the proposed method in terms of various performance metrics. Second, we also show the interpretability of the proposed method through both qualitative and quantitative evaluations.

Table 3. Statistics of Our Datasets After Preprocessing

	NeurIPS (2013–2016)		ICLR (2017–2018)	
	Abstract	Review	Abstract	Review
#Paper	1,739	7,171	534	1,819
#Sentence	10,964	109,674	3,219	25,148
#Word	6,448	16,695	3,846	8,981

5.1 Experimental Setup

5.1.1 Datasets. Most of the academic journals and conferences have not provided the public access to their peer review texts. Fortunately, NeurIPS and ICLR conferences provide the review comments of their papers published in recent years on their official websites. What is more, from 2013 to 2016, NeurIPS asked reviewers not only to provide detailed comments to the papers but also to summarize their review comments in one or two sentences, which can be adopted as review summarization in our model for citation count prediction task. For each paper, we combine such summarization sentences from all the involved reviewers as this paper's summarization text. Similarly, ICLR asks reviewers to provide reasons for the acceptance of each paper from 2017 to 2018, containing one to two sentences, which can be adopted as review summarization for our task.

For other information of the papers, the title and abstract data of NeurIPS 2013–2016 and ICLR 2017–2018 have been collected and shared with the public by Kang et al. [33]. All of the authors' h-index and citation count data for the papers have been provided by Li et al. [40], except for the ICLR 2018. We collected the data for ICLR 2018 with the same approach described by Li et al. [40]. As the citation counts are changing, we re-collect the citation count data by crawling from Google Scholar.[3] All the new citation data was refreshed on November 31, 2019, to guarantee the recency. These citation counts are used as ground truth to train and test the prediction performance of the compared models.

With NLTK,[4] we perform basic text preprocessing including tokenization, lowercase, and stopword and rare words removal. Some statistics of these two datasets after preprocessing are summarized in Table 3. For the NeurIPS dataset, we take the data from NeurIPS 2016 as the test set and the data from NeurIPS 2013–2015 as the training set. For the ICLR dataset, we take 20% of the data in ICLR 2018 as the test set, and the rest of ICLR 2018 and the data of ICLR 2017 are combined as the training set because the data in ICLR 2017 is limited for training. For both datasets, we retain 5% training data as the validation set. Our code and dataset have been released at https://github.com/RUCAIBox/Interpretable-Citation-Count-Prediction.

5.1.2 Implementation Details. For the input layer, we pretrain word embeddings with 300-dimensions as the initial vector representations of words in the vocabulary. The dimension of the sentence vectors d_s is set to be 64. Both the dimensions of feature capsules d_p and aspect capsules d_c are also set to be 64. The size of the sliding window in the feature capsule layer is set to be 3. The number of feature capsules is 16, and the number of aspect capsules is 5. In the aspect capsule layer, we repeat the dynamic routing procedure for three iterations. Both the dimensions of hidden vectors in the aspect decoder d_A and sentence decoder d_H are set to be 32.

After preprocessing the text data, we adopt the Twitter-LDA model [89] to automatically learn the topic distribution over review summarization texts, the topic keywords, and the aspect label of each sentence. During the generation process, we employ the beam search algorithm with the

[3] https://scholar.google.com.
[4] https://www.nltk.org.

Table 4. Parameter Settings of the Three Modules in Our Model

Module	Setting
Feature Capsule Layer	d_s = 64, d_p = 64, batch size = 32, size of the sliding window = 3, number of feature capsules = 16, dropout rate = 0.5, learning rate = 0.01, SGD optimizer
Aspect Capsule Layer	d_c = 64, d_A = 32, batch size = 32, r = 3, number of aspect capsules = 5, dropout rate = 0.5, learning rate = 0.01, SGD optimizer
Interpreting Layer	d_H = 32, batch size = 32, beam size = 4, dropout rate = 0.5, learning rate = 0.01, SGD optimizer

beam size of 4. In the wide component, we pretrain a 100-dimensional topic model by using LDA [5]. Then we calculate the entropy of each topic distribution as its diversity. To avoid overfitting, we adopt the dropout strategy with a rate of 0.5. We optimize the model with the **Stochastic Gradient Descent (SGD)** optimizer, with a learning rate of 0.01. Further details about parameter settings are summarized in Table 4.

5.1.3 Baseline Models. We adopt the following baseline models for comparison:

- *Linear Regression*: The **Linear Regression (LR)** baseline captures the relationship between citation counts and the features of papers by fitting a linear model with observed data.
- *K-Nearest Neighbor* [76]: ***K*-Nearest Neighbor (KNN)** predicts the citation count for a paper as the average of the citation counts of its k-nearest neighbors based on Euclidean distance. The number of neighbors is set to be 60.
- *Support Vector Regression* [76]: **Support Vector Regression (SVR)** learns a model to predict the citation count in a high-dimensional feature space by solving a constrained quadratic optimization problem where the objective function is the combination of the loss function and a regularization term.
- *Gradient Boost Regression Tree*: **Gradient Boost Regression Tree (GBRT)** is a regression tree model in which a greedy optimization process recursively partitions the feature space. The features described in the wide component (Section 4.2.2) are included, the learning rate is set to be 0.1, and the number of boosting stages to perform is set to be 85.
- *Wide & Deep* [16]: The task of predicting the citation count can use both text and handcrafted features, which is similar to some scenarios in recommender systems. Thus, we adopt the widely used Wide & Deep framework from recommender systems as a baseline. We use a feed-forward neural network with a bidirectional RNN module modeling review text for the deep component.
- *HUARN* [37]: HUARN is a method on aspect-aware sentiment analysis that aims to predict a user's sentiment polarities for different aspects of a product in a review. It adopts a hierarchical architecture to encode word-, sentence-, and document-level information. Then, user attention and aspect attention are utilized to learn sentence- and document-level representations. We concatenate the embeddings of the top five keywords in each aspect as the aspect embedding. Since their original task is for sentiment polarity prediction, we modify the last layer as that in our approach. For fairness, we also utilize the Wide & Deep framework and the same wide features for this baseline method as our model.

- *SChuBERT* [64]: SChuBERT uses a pre-trained BERT model [21] to extract features to learn from the features and predict the citation count. It uses contextualized word embeddings instead of context-independent word embeddings. Compared to static embeddings, it enables a more capable modeling for the context that a semantic unit appears in.
- *MILAM* [65]: MILAM is the first model to utilize review text to predict the value of scholarly papers, a similar but not exactly the same task. It proposes an abstract-based memory mechanism to predict the overall decision (accept, reject, or borderline) based on review text and abstract text. We slightly modify its goal to fit the citation count prediction task. To ensure a fair comparison, we also utilize the Wide & Deep framework and the same wide features for this baseline method as our model.
- *Neural CCP* [40]: Neural CCP is the first existing method that utilizes the review text to predict the citation count of a scholarly paper. It adopts the Wide & Deep framework and learns the semantic representation for peer review text with match mechanisms. This method provides a very strong baseline to validate our method.

For LR, KNN, SVR, and GBRT, we adopt features from the wide component as the input for the preceding four baselines, which leads to the similar setting of input for existing methods [76]. For the five deep learning based methods, we also consider both wide features and review text as the input.

5.1.4 Performance Metrics. To evaluate the performance of different methods for citation count prediction, we adopt several different evaluation metrics that are widely used in existing related works: **Mean Absolute Error (MAE)**, **Root Mean Square Error (RMSE)**, **Overlapping Rate@k (OR@10, OR@30, OR@50)**, and **Spearman's Rank (SR)** correlation coefficient:

$$\text{MAE} = \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}|(c_d - \hat{c}_d)|, \tag{22}$$

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}(c_d - \hat{c}_d)^2}, \tag{23}$$

$$\text{OR@k} = \frac{|\#TruePrediction@k|}{k}, \tag{24}$$

$$\text{SR} = \frac{\sum_{d\in\mathcal{D}}(c_d - \bar{c})(\hat{c}_d - \bar{\hat{c}})}{\sqrt{\sum_{d\in\mathcal{D}}(c_d - \bar{c})^2 \sum_{d\in\mathcal{D}}(\hat{c}_d - \bar{\hat{c}})^2}}, \tag{25}$$

where $\mathcal{D}$ is the literature corpus, c_d is the ground-truth citation count of the paper d, and $\hat{c}_d$ is the corresponding predicted citation count. Specifically, MAE and RMSE measure the average error between the predicted citation count and ground-truth citation count. OR@k measures the overlapping rate between the top-k of the predicted paper list and ground-truth paper list both sorted by citation count. The SR correlation coefficient measures the correlation between two sorted lists. These performance metrics measure the citation count prediction ability of methods from different angles and enable us to fairly compare different methods.

5.2 Evaluation on Performance

In this section, we compare the proposed method with several baselines methods for the citation count prediction task. We also conduct an ablation study and parameter sensitivity study to analyze the performance of the proposed method under different settings.

Table 5. Performance Comparisons of Different Methods for Citation Count Prediction on Two Datasets

Dataset	Method	MAE ("↓")	RMSE ("↓")	OR@10 ("↑")	OR@30 ("↑")	OR@50 ("↑")	SR ("↑")
NeurIPS	LR	0.1753	0.1893	0.10	0.27	0.33	0.4796
	KNN	0.1683	0.1882	0.20	0.33	0.36	0.4908
	SVR	0.1649	0.1801	0.10	0.33	0.40	0.5307
	GBRT	0.1784	0.1920	0.10	0.27	0.34	0.5371
	Wide & Deep	0.1453	0.1828	0.10	0.30	0.38	0.5388
	HUARN	0.1440	0.1799	0.10	0.33	0.38	0.5402
	SChuBERT	0.1429	0.1781	0.10	0.33	0.38	0.5443
	MILAM	0.1417	0.1752	0.20	0.37	0.38	0.5499
	Neural CCP	0.1348	0.1722	0.20	0.40	0.42	0.5581
	Our method	**0.1309***	**0.1697***	**0.20**	**0.42**	**0.44**	**0.5712**
ICLR	LR	0.2364	0.2691	0.20	0.40	0.70	0.1873
	KNN	0.2255	0.2620	0.20	0.40	0.72	0.1985
	SVR	0.2215	0.2511	0.20	0.40	0.70	0.1598
	GBRT	0.2204	0.2594	0.20	0.43	0.70	0.1749
	Wide & Deep	0.2106	0.2587	0.20	0.47	0.72	0.2582
	HUARN	0.2101	0.2556	0.20	0.47	0.72	0.2598
	SChuBERT	0.2052	0.2529	0.30	0.47	0.72	0.2613
	MILAM	0.2008	0.2486	0.30	0.47	0.72	0.2622
	Neural CCP	0.1849	0.2251	0.30	0.50	0.76	0.3107
	Our method	**0.1753***	**0.2144***	**0.30**	**0.53**	**0.78**	**0.3279**

"↑" ("↓") indicates that a larger (smaller) value corresponds to a better performance. The best performances are marked bold in the table. We conducted a two-tailed t-test, and "$*$" indicates the statistical significance for $p < 0.05$ compared to the best baseline (significance test cannot apply to ranking-based metrics).

5.2.1 Main Results. In Table 5, we report the performance of different methods on the citation count prediction task. From this table, we can observe the following:

(1) Among the four traditional baselines (i.e., LR, KNN, SVR, GBRT), SVR achieves the best performance with MAE, RMSE, and OR@k. But GBRT outperforms SVR with the metric of SR. These results show that SVR predicts better in a point-wise way, whereas GBRT performs better in a list-wise way. Furthermore, all of these traditional methods perform worse than the five deep learning baselines (i.e., Wide & Deep, HUARN, SChuBERT, MILAM, Neural CCP). This is because these traditional baselines cannot learn useful latent representations from abstract and review texts, not to mention capture the interaction between abstract and review texts.

(2) Among the deep learning baselines, Neural CCP performs the best. For the Wide & Deep baseline, it only models reviews text with a feed-forward neural network and a bi-directional RNN component. HUARN outperforms Wide & Deep because it uses the attention mechanism to capture the aspect information in review text, whereas SChuBERT utilizes BERT to better learn the semantic information from rich context so it performs better. MILAM and Neural CCP are able to explicitly model the interaction between review text and abstract text, learning useful information from text-based interaction for citation count prediction. As mentioned earlier, MILAM is not specially designed for the citation count prediction task, so Neural CCP achieves the best performance among this category of baseline methods.

(3) Finally, it is obvious that our proposed model performs consistently better than all the baselines under different metrics. Such advantages are brought by the fact that we design a novel abstract routing approach to identify the core information in review texts. For example, Neural

Table 6. Ablation Study of the Proposed Method

Dataset	Method	MAE	RMSE	OR@10	OR@30	OR@50	SR
NeurIPS	Our full model	**0.1309**	**0.1697**	**0.20**	**0.42**	**0.44**	**0.5712**
	W/o feature capsules	0.1366	0.1759	0.20	0.40	0.38	0.5368
	W/o aspect capsules	0.1356	0.1728	0.20	0.40	0.40	0.5469
	W/o topic supervision	0.1347	0.1720	0.20	0.40	0.42	0.5553
ICLR	Our full model	**0.1753**	**0.2144**	**0.30**	**0.53**	**0.78**	**0.3279**
	W/o feature capsules	0.1908	0.2423	0.30	0.47	0.72	0.2779
	W/o aspect capsules	0.1887	0.2307	0.30	0.50	0.74	0.2846
	W/o topic supervision	0.1854	0.2279	0.30	0.50	0.74	0.3013

The best performances are marked bold in the table.

CCP only considers the interaction between abstract and review at sentence level, ignoring the mixed aspect information. Different from it, the proposed model learns the aspect-level representation of review text in a fine-grained way. Furthermore, the interpretation part of the proposed method, which we will discuss in the next section, enhance its ability to capture the core aspects in review text for citation count prediction.

5.2.2 Ablation Study. The major novelty of our model is that it utilizes two capsule network layers to gradually learn the aspect information of peer review texts. In addition, it uses the topic distribution of review summarization texts. To examine the contributions of these three parts, we test the performance of three variants of the proposed method by removing each module from the full model. Specifically, we consider the following variants for the ablation study:

- *W/o feature capsules*: In this variant, we remove the feature capsule layer and the corresponding abstract routing layer, and directly learn the aspect capsules from raw texts.
- *W/o aspect capsules*: We remove the aspect capsule layer, directly predicting the citation count with the feature capsules.
- *W/o topic distribution*: This variant learns the vector representation of aspect capsules without considering the topic distribution of review summarization texts as supervision.

We report the experimental results of our full model and these variants in Table 6. As we can see, the performance of our full model is better than all of these variants. The performance of the variant without feature capsules is the worst one, showing that the abstract routing approach can effectively identify the important information from raw review texts. The results of the variant without aspect capsules demonstrate the necessity of learning aspects based on feature capsules to enhance the representations of useful information as the aspect-aware representations can be easily used for the final citation count prediction. The results of the variant without topic supervision are still good and indicate that our model has learned a meaningful representation of the aspect-aware information. In summary, all of these modules can work together to extract useful information from review texts, and they enable the proposed model to significantly improve the citation count prediction task.

5.2.3 Parameter Sensitivity Study. In this section, we further investigate the influence of model parameters on the performance to verify its robustness. For simplicity, we select the best two neural network baselines as the comparison methods. We report the results on the NeurIPS dataset in Figure 2 and results on the ICLR dataset in Figure 3.

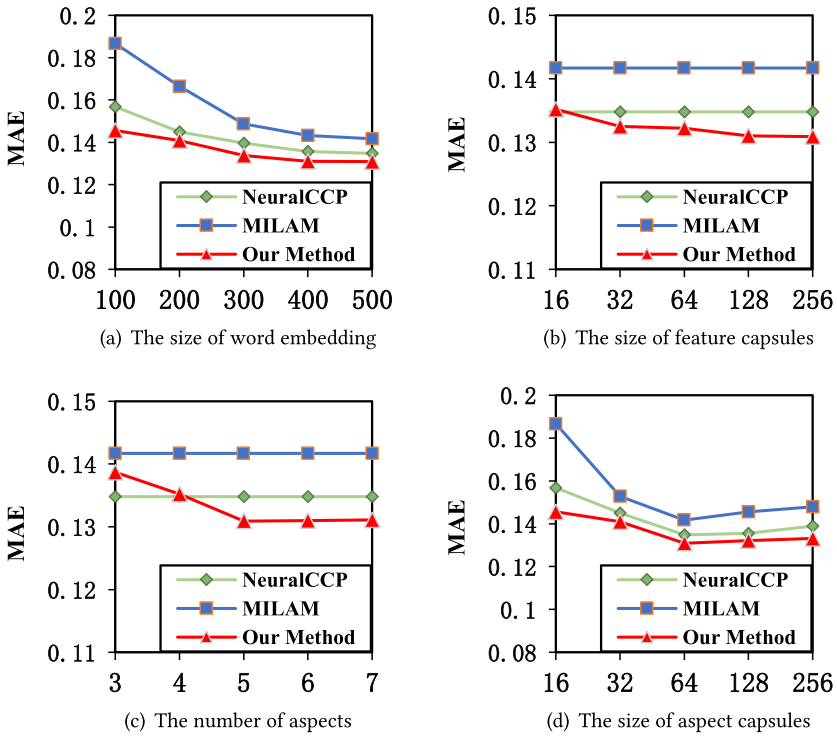

Fig. 2. Performance of different parameters in terms of MAE on the NeurIPS dataset.

Figure 3(a) and (b) present the results for tuning the size of word embedding. We vary the dimensionality from 100 to 500 with a gap of 100 to examine how the performance of each model changes. It can be seen that by increasing the embedding size, the performance of all the methods become better, whereas our model is consistently better than other baselines, indicating the stability of the proposed method.

Figure 2(b) and Figure 3(b) show the results for tuning the dimensionality of feature capsules, which controls how much information can be learned from review texts. We vary the dimension in a set {16, 32, 64, 128, 256}. It can be seen that the performance increases with the growth of feature capsule dimensionality and becomes stable around 64. The performance of our model is consistently better than other baselines.

We further investigate the performance of the proposed method with respect to two major parameters in our model (e.g., the number of aspect capsules and the dimensionality of aspect capsules). Figure 2(c) and Figure 3(c) show the experiment results of varying the number of aspect capsules from 3 to 7, and we can see that our model is consistently better than baselines. The performance becomes stable when the number of aspects is larger than 4.

Figure 2(d) and Figure 3(d) show the performance for tuning the dimensionalities of the final review representation, aspect capsule, which decides how much information of review can be kept for the citation count prediction. In addition, our method consistently outperforms other methods. We can also observe that the aspect capsule size should be set appropriately: a small one may lose some information for final prediction, whereas a large one may introduce some noise.

These results have shown that our model is robust with different parameter settings.

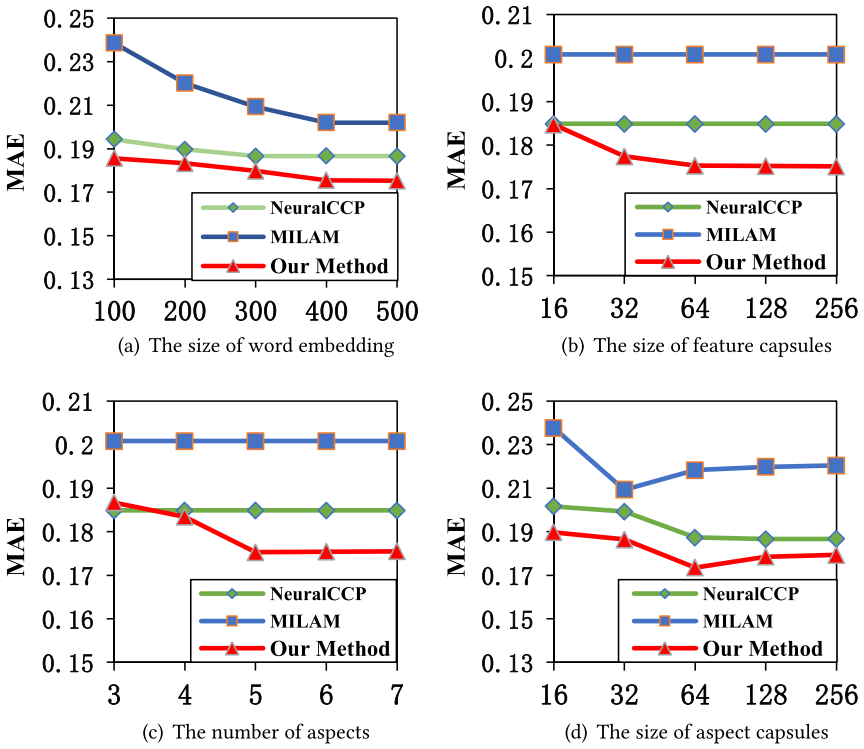

Fig. 3. Performance of different parameters in terms of MAE on ICLR dataset.

5.2.4 Study on Varied-Length Prediction Periods. Recall that we have used the collected citation counts in the year 2019 as ground truth in previous experiments. An important issue is whether our approach can make stable predictions with varied-length prediction periods. To examine this issue, we adopted the citation counts released by Li et al. [40] as a comparison, which were collected in the year 2018. With the same training and testing papers, we set up the ground truth to be predicted according to two different time periods (i.e., until 2018 and until 2019). For simplicity, we select the best traditional method and best neural network baseline as the comparison methods. We only present the results on MAE and SR.

The comparison results using the statistics of the years 2018 and 2019 are presented in Table 7. Overall, we can see that our approach outperforms the compared baselines in both settings. Comparing the results in the two settings, the prediction performance of our approach seems to be relatively stable with two different time periods. In this article, predicting the dynamic change is not our focus, and we mainly focus on accumulative citation count prediction, which will be useful to estimate the long-term impact of a paper.

5.2.5 Study on Sentence Embedding. Our approach can be easily extended to use other sentence embedding methods or networks. Since BERT [22] is a popular and effective sentence embedding method, we tried using BERT instead of TextCNN to learn the sentence embeddings and explore whether such embeddings could be useful to further improve the prediction performance. Since we used the scientific dataset, we further conducted an experiment with SciBERT [3], which has been fine tuned in scientific text, for comparison with vanilla BERT. For these two BERT

Table 7. Performance of Different Methods in Terms of MAE on Data Collected in 2018 and 2019

Dataset	Method	Till 2018		Till 2019	
		MAE	SR	MAE	SR
NeurIPS	SVR	0.1677	0.5279	0.1675	0.5285
	NeuralCCP	0.1349	0.5561	0.1336	0.5566
	Our method	**0.1319**	**0.5694**	**0.1314**	**0.5696**
ICLR	SVR	0.2226	0.1328	0.2221	0.1329
	NeuralCCP	0.1866	0.3026	0.1865	0.3029
	Our method	**0.1769**	**0.3230**	**0.1766**	**0.3236**

The best performances are marked bold in the table.

Table 8. Performance of Different Sentence Embeddings on NeurIPS

Method	MAE	RMSE	OR@10	OR@30	OR@50	SR
Ours (original)	0.1309	0.1697	0.20	0.42	0.44	0.5712
Ours (BERT)	0.1301	0.1688	0.20	0.42	0.46	0.5769
Ours (SciBERT)	**0.1264**	**0.1611**	**0.30**	**0.44**	**0.48**	**0.5824**

The best performances are marked bold in the table.

methods, we used the embeddings of "[CLS]"token in each sentence at the last layer as the contextual embeddings of each sentence. To summarize, we consider the following variants:

- *Ours (BERT)*: In this variant, we use BERT to learn the sentence embeddings.
- *Ours (SciBERT)*: In this variant, we use SciBERT to learn the sentence embeddings.

We report the experimental results of our original method and the two variants in Table 8. As we can see, BERT slightly improves the performances since it is a powerful pretrained language model. Comparing the two BERT-based variants, it can be observed that SciBERT performs better than BERT, since it has been specially pretrained on scholarly papers. As a comparison, the vanilla BERT was trained on the general Wikipedia and BookCorpus, which may not be suitable in our case with scholarly papers.

5.3 Evaluation on Interpretability

As mentioned earlier, besides performance, interpretability is also very important for citation count prediction as it can explain the learned model and the prediction results. In this section, we present the evaluation on interpretability to show that our proposed model is able to effectively learn the aspect information, and such information can be further used to generate review summarization.

In Table 9, we present three sampled review summarizations generated by the proposed method. To make the meanings of aspects clear, Table 10 presents the top five words for each aspect on the ICLR dataset. The second line is the manual annotations of aspects for understanding. From Table 9, we can see that for three different levels of citation count, our model is able to cover the major aspects, and the generated review summarization can explain the citation count to some extent. Next, we quantitatively evaluate the generated review summarization in terms of the aspect coverage and correlation with citation count.

Table 9. Samples of Review Summarization Generated by the Proposed Method (CC = Citation Count)

CC	Generated Review Summarization	Ground-Truth Review Summarization
502	this paper presents an algorithm$_{method}$ widely used. the paper is well written$_{clarity}$. the experiment$_{experiment}$ proof the methods. it is very important i strongly recommend$_{quality}$ it	this paper analyzes a problem with the convergence of Adam and presents a solution. the fix$_{method}$ is both principled and practical. the paper is clearly written$_{clarity}$. this is a strong$_{quality}$ paper and i recommend acceptance$_{quality}$.
47	this is a solid$_{quality}$ work on molecular dynamic. it propose the method$_{method}$ with the stochastic gradients. a clear$_{clarity}$ and accessible read though only small scale experiment$_{experiment}$	this paper builds on much of the solid$_{quality}$ work in molecular dynamics by the likes of Leimkuhler. the evaluations experimentally$_{experiment}$ are a bit simplistic though the final DBM was interesting. the algorithm$_{method}$ is incremental over SGNHT and it has a flaw$_{quality}$ which has not been fully addressed
8	this is a theoreticaled$_{method}$ work. the paper results$_{experiment}$ on matrix problem results$_{experiment}$ are similar. it is not clearly$_{clarity}$ justified	very nice theoreticalod$_{method}$ generalization of recovery results for matrix completion under general structural constraints. not clear$_{clarity}$ how tight are the bounds achieved. there are also some inconsistencies in the results$_{experiment}$ that need to be addressed or discussed

Table 10. Top Five Words of Each Aspect

Aspect 1	Aspect 2	Aspect 3	Aspect 4	Aspect 5
Method	Experiment	Network	Quality	Clarity
algorithm	results	learning	good	written
optimization	experimental	neural	accept	clear
bounds	interesting	data	novel	writing
theoretical	performance	network	contribution	figure
convergence	case	deep	approach	section

Table 11. Aspect Coverage Evaluation

Dataset	Method	Coverage Rate
NeurIPS	Our full model	**0.78**
	W/o aspect decoder	0.67
	ABS	0.63
ICLR	Our full model	**0.72**
	W/o aspect decoder	0.65
	ABS	0.61

5.3.1 Aspect Coverage Evaluation. Following the previous aspect-aware text generation work [46], we evaluate the aspect coverage of different methods by comparing the generated review summarization with the corresponding ground truth. To be specific, we compute the average of the percentage of aspects in ground truth that are covered in the generated texts. As the word distributions of aspects are available based on Twitter-LDA, we say a generated text covers an aspect if any of the top 20 words of that aspect appear in the generated text.

We adopt a variant of the proposed method by removing the aspect decoder and an existing method, **Attention-Based Summarization (ABS)** [54], as baselines for comparison. Table 11 reports the experiment results of different methods in terms of average coverage rate. We can see that our model can identify more aspects than the other two baselines, because the aspect capsules are able to learn useful aspect-level information from peer review text. Furthermore, with the

Table 12. Correlation Evaluation Between Review Summarization and Citation Count on the NeurIPS and ICLR Datasets

Dataset	Ground Truth	Our Method
NeurIPS	0.72	0.61
ICLR	0.77	0.68

guidance of topic distribution learned by LDA, the useful aspects for the citation count prediction task are effectively clustered to specific capsules.

5.3.2 Correlation Evaluation. We further conduct an experiment to quantify the correlation between the generated review summarization and citation count of papers. We randomly choose 176 papers from the test sets of the NeurIPS and ICLR datasets, and divide the papers into three levels sorted by their ground-truth citation counts. So the top one-third of papers are assigned to level 1, the last one-third of papers are assigned to level 3, and the rest of the papers are assigned to level 2. Each paper has been associated with the ground-truth review texts and the generated review summarization texts. We shuffle these texts and then invite two human annotators who are familiar with computer science research areas to label which citation count level a paper belongs to given either the ground-truth review or the generated review summarization text. Then we calculate the accuracy of the assigned levels of papers given ground-truth review and generated review, respectively. Table 12 presents the experiment results in terms of correlation. The results show that the performance of the generated summarization text is a bit lower than the ground-truth summarization text. This might be because our method can only learn aspects guided by the topic distributions, whereas the ground-truth review texts can discuss the advantages and disadvantages of the paper in a more detailed and complicated way. Despite this, the generated review summaries still have a high correlation with the citation count, and thus they are helpful to predict the citation count and enhance the interpretability.

6 CONCLUSION

In this article, we presented an aspect-aware capsule network for extracting useful semantic representations from review text for citation count prediction. Our model contained two capsule layers, namely the feature capsule layer and the aspect capsule layer, with two different routing approaches, respectively. By adopting a hierarchical representation architecture (i.e., *sentence* → *feature capsule* → *aspect capsule*), our model is able to gradually extract useful semantic characteristics from fine-grained units to high-level aspects. In addition, we proposed two strategies to enhance the model interpretability, namely aspect alignment and aspect-based review summarization. Extensive experiments on real-world datasets have demonstrated the superiority of the proposed model in terms of both performance and interpretability.

Currently, we studied to predict the long-term impact of research papers—that is, predict the accumulate citation count after a fixed time period (the mostly adopted task definition in the literature for citation count prediction). As future work, it would be meaningful to extend the current approach in a dynamic manner so that it can capture the temporal trends of citations. In addition, we used topic models as isolated components for providing a useful signal to improve our model. We will consider how to integrate topic models with the capsule network in a more principled way. Finally, due to the limitation of the access to review texts, we conducted experiments on two available datasets. In the future, we will consider collecting more public peer reviews and test the performance of our approach on new datasets.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful and constructive comments.

REFERENCES

[1] Kristen M. Altenburger and Daniel E. Ho. 2019. Is Yelp actually cleaning up the restaurant industry? A re-analysis on the relative usefulness of consumer reviews. In *Proceedings of the World Wide Web Conference (WWW'19)*. 2543–2550.

[2] Sebastian Arnold, Betty van Aken, Paul Grundmann, Felix A. Gers, and Alexander Löser. 2020. Learning contextualized document representations for healthcare answer retrieval. In *Proceedings of the 2020 Web Conference (WWW'20)*. 1332–1343.

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 3606–3611.

[4] Harish S. Bhat, Li-Hsuan Huang, Sebastian Rodriguez, Rick Dale, and Evan Heit. 2015. Citation prediction using diverse features. In *Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW'15)*. 589–596.

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

[6] Samuel Brody and Noemie Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 804–812.

[7] Carlos Castillo, Debora Donato, and Aristides Gionis. 2007. Estimating number of citations using author reputation. In *Proceedings of the International Symposium on String Processing and Information Retrieval (SPIRE'07)*. 107–117.

[8] Muthusamy Chelliah, Yong Zheng, and Sudeshna Sarkar. 2019. Recommendation for multi-stakeholders and through neural review mining. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 2979–2981.

[9] Cen Chen, Minghui Qiu, Yinfei Yang, Jun Zhou, Jun Huang, Xiaolong Li, and Forrest Sheng Bao. 2019. Multi-domain gated CNN for review helpfulness prediction. In *Proceedings of the World Wide Web Conference (WWW'20)*. 2630–2636.

[10] Junpeng Chen and Chunxia Zhang. 2015. Predicting citation counts of papers. In *Proceedings of the 2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing (ICCI* CC'15)*. 434–440.

[11] Xinchi Chen, Chunchuan Lyu, and Ivan Titov. 2019. Capturing argument interaction in semantic role labeling with capsule networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 5418–5428.

[12] Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013. Discovering coherent topics using general knowledge. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*. 209–218.

[13] Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013. Exploiting domain knowledge in aspect extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 1655–1667.

[14] Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2013. Leveraging multi-domain prior knowledge in topic models. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*. 2071–2077.

[15] Zhuang Chen and Tieyun Qian. 2019. Transfer capsule network for aspect level sentiment classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL'19)*. 547–556.

[16] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, et al. 2016. Wide and deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (DLRS'16)*. ACM, New York, NY, 7–10.

[17] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C. Kanjirathinkal, and Mohan Kankanhalli. 2019. MMALFM: Explainable recommendation by leveraging reviews and images. *ACM Transactions on Information Systems* 37, 2 (2019), 1–28.

[18] Zhiyong Cheng, Ying Ding, Lei Zhu, and Mohan Kankanhalli. 2018. Aspect-Aware latent factor model: Rating prediction with ratings and reviews. In *Proceedings of the 2018 World Wide Web Conference*. 639–648.

[19] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1724–1734.

[20] Feruz Davletov, Ali Selman Aydin, and Ali Cakmak. 2014. High impact academic paper prediction using temporal and topological features. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'14)*. 491–498.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long and Short Papers)*. 4171–4186.

[23] Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Chun Wang, and Bing Ma. 2019. Investigating capsule network and semantic feature on hyperplanes for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 456–465.

[24] Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Tong Xu, and Ming Liu. 2019. Capsule network with interactive attention for aspect-level sentiment classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 5492–5501.

[25] Miao Fan, Chao Feng, Lin Guo, Mingming Sun, and Ping Li. 2019. Product-aware helpfulness prediction of online reviews. In *Proceedings of the World Wide Web Conference (WWW'19)*. 2715–2721.

[26] Martin Fisher, Stanford B. Friedman, and Barbara Strauss. 1994. The effects of blinding on acceptance of research papers by peer review. *JAMA* 272, 2 (1994), 143–146.

[27] Lawrence D. Fu and Constantin Aliferis. 2008. Models for predicting and explaining citation count of biomedical articles. *AMIA Annual Symposium Proceedings* 2008 (2008), 222–226.

[28] Xinyu Guan, Zhiyong Cheng, Xiangnan He, Yongfeng Zhang, Zhibo Zhu, Qinke Peng, and Tat-Seng Chua. 2019. Attentive aspect modeling for review-aware recommendation. *ACM Transactions on Information Systems* 37, 3 (2019), 1–27.

[29] Geoffrey E. Hinton, Alex Krizhevsky, and Sida D. Wang. 2011. Transforming auto-encoders. In *Proceedings of the IInternational Conference on Artificial Neural Networks (ICANN'11)*. 44–51.

[30] Alfonso Ibáñez, Pedro Larrañaga, and Concha Bielza. 2009. Predicting citation count of bioinformatics papers within four years of publication. *Bioinformatics* 25, 24 (2009), 3303–3309.

[31] Zhuoren Jiang, Jian Wang, Lujun Zhao, Changlong Sun, Yao Lu, and Xiaozhong Liu. 2019. Cross-domain aspect category transfer and detection via traceable heterogeneous graph representation learning. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 289–298.

[32] Yohan Jo and Alice H. Oh. 2011. Aspect and sentiment unification model for online review analysis. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*. 815–824.

[33] Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. A dataset of peer reviews (PeerRead): Collection, insights and NLP applications. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers) (ACL'18)*. 1647–1661.

[34] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1746–1751.

[35] Chenliang Li, Xichuan Niu, Xiangyang Luo, Zhenzhong Chen, and Cong Quan. 2019. A review-driven neural model for sequential recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*.

[36] Chenliang Li, Cong Quan, Li Peng, Yunwei Qi, Yuming Deng, and Libing Wu. 2019. A capsule network for recommendation and explaining what you like and dislike. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*.

[37] Junjie Li, Haitong Yang, and Chengqing Zong. 2018. Document-level multi-aspect sentiment classification by jointly modeling users, aspects, and overall ratings. In *Proceedings of the 27th International Conference on Computational Linguistics*. 925–936.

[38] Junyi Li, Wayne Xin Zhao, Ji-Rong Wen, and Yang Song. 2019. Generating long and informative reviews with aspect-aware coarse-to-fine decoding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL'19)*. 1969–1979.

[39] Pengfei Li, Hua Lu, Gang Zheng, Qian Zheng, Long Yang, and Gang Pan. 2019. Exploiting ratings, reviews and relationships for item recommendations in topic based social networks. In *Proceedings of the World Wide Web Conference (WWW'19)*. 995–1005.

[40] Siqing Li, Wayne Xin Zhao, Eddy Jing Yin, and Ji-Rong Wen. 2019. A neural citation count prediction model based on peer review text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 4916–4926.

[41] Jiangming Liu and Yue Zhang. 2017. Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*. 572–577.

[42] Gary Marchionini. 2008. Reviewer merits and review control in an age of electronic manuscript management systems. *ACM Transactions on Information Systems* 26, 4 (2008), 25.

[43] Zhengjie Miao, Yuliang Li, Xiaolan Wang, and Wang-Chiew Tan. 2020. Snippext: Semi-supervised opinion mining with augmented data. In *Proceedings of the 2020 Web Conference (WWW'20)*. 617–628.

[44] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS'13)*. 3111–3119.

[45] Rajdeep Mukherjee, Hari Chandana Peruri, Uppada Vishnu, Pawan Goyal, Sourangshu Bhattacharya, and Niloy Ganguly. 2020. Read what you need: Controllable aspect-based opinion summarization of tourist reviews. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20)*.

[46] Jianmo Ni and Julian McAuley. 2018. Personalized review generation by expanding phrases and attending on aspect-aware representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (ACL'18)*. 706–711.

[47] Slava Novgorodov, Ido Guy, Guy Elad, and Kira Radinsky. 2019. Generating product descriptions from user reviews. In *Proceedings of the World Wide Web Conference (WWW'19)*. 1354–1364.

[48] Iyiola E. Olatunji, Xin Li, and Wai Lam. 2019. Context-aware helpfulness prediction for online product reviews. In *Proceedings of the Asia Information Retrieval Symposium (AIRS'19)*.

[49] Minh Hieu Phan and Philip O. Ogunbona. 2020. Modelling context and syntactical features for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3211–3220.

[50] Nataliia Pobiedina and Ryutaro Ichise. 2014. Predicting citation counts for academic literature using graph pattern mining. In *Proceedings of the International Conference on Industrial, Engineering, and Other Applications of Applied Intelligent Systems (IEA/AIE'14)*. 109–119.

[51] Lakshmi Ramachandran and Edward F. Gehringer. 2011. Automated assessment of review quality using latent semantic analysis. In *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'11)*. IEEE, Los Alamitos, CA, 136–138.

[52] Joseph S. Ross, Cary P. Gross, Mayur M. Desai, Yuling Hong, Augustus O. Grant, Stephen R. Daniels, Vladimir C. Hachinski, Raymond J. Gibbons, Timothy J. Gardner, and Harlan M. Krumholz. 2006. Effect of blinded peer review on abstract acceptance. *JAMA* 295, 14 (2006), 1675–1680.

[53] Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 999–1005.

[54] Alexander M. Rush, Sumit Chopra, and Jason Weston. 2017. A neural attention model for sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP'17)*.

[55] Sara Sabour, Nicholas Frosst, and Geoffrey Hinton. 2018. Matrix capsules with EM routing. In *Proceedings of the 6th International Conference on Learning Representations (ICLR'18)*. 1–15.

[56] Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. 2017. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems (NIPS'17)*. 3856–3866.

[57] Noveen Sachdeva and Julian McAuley. 2020. How useful are reviews for recommendation? A critical review and potential improvements. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20)*.

[58] Tian Shi, Vineeth Rakesh, Suhang Wang, and Chandan K. Reddy. 2019. Document-level multi-aspect sentiment classification for online reviews of medical experts. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 2723–2731.

[59] Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long and Short Papers)*. 380–385.

[60] Peijie Sun, Le Wu, Kun Zhang, Yanjie Fu, Richang Hong, and Meng Wang. 2020. Dual learning for explainable recommendation: Towards unifying user preference prediction and review generation. In *Proceedings of the 2020 Web Conference (WWW'20)*. 837–847.

[61] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. 3298–3307.

[62] Yufei Tian, Jianfei Yu, and Jing Jiang. 2019. Aspect and opinion aware abstractive review summarization with reinforced hard typed decoder. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 2061–2064.

[63] Quoc-Tuan Truong and Hady Lauw. 2019. Multimodal review generation for recommender systems. In *Proceedings of the International Conference on World Wide Web (WWW'19)*. 1864–1874.

[64] Thomas van Dongen, Gideon Maillette de Buy Wenniger, and Lambert Schomaker. 2020. SChuBERT: Scholarly document chunks with BERT-encoding boost citation count prediction. In *Proceedings of the 1st Workshop on Scholarly Document Processing*. 148–157.

[65] Ke Wang and Xiaojun Wan. 2018. Sentiment analysis of peer review texts for scholarly papers. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'18)*. 175–184.
[66] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 606–615.
[67] Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019. Aspect-level sentiment analysis using as-capsules. In *Proceedings of the World Wide Web Conference (WWW'19)*. 2033–2044.
[68] Chuhan Wu, Fangzhao Wu, Junxin Liu, Yongfeng Huang, and Xing Xie. 2019. ARP: Aspect-Aware neural review rating prediction. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 2169–2172.
[69] Haifeng Xia, Zengmao Wang, Bo Du, Lefei Zhang, Shuai Chen, and Gang Chun. 2019. Leveraging ratings and reviews with gating mechanism for recommendation. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 1573–1582.
[70] Shuai Xiao, Junchi Yan, Changsheng Li, Bo Jin, Xiangfeng Wang, Xiaokang Yang, Stephen M. Chu, and Hongyuan Zha. 2016. On modeling and predicting individual paper citation count over time. In *Proceedings of the International Joint Conference on Artifical Intelligence (IJCAI'16)*. 2676–2682.
[71] Wenting Xiong and Diane Litman. 2011. Automatically predicting peer-review helpfulness. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers) (ACL'11)*. 502–507.
[72] Wenting Xiong, Diane Litman, and Christian Schunn. 2010. Assessing reviewers' performance based on mining problem localization in peer-review data. In *Proceedings of the International Conference on Educational Data Mining (ICEDM'10)*. 211–220.
[73] Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1)*.
[74] Xueke Xu, Xueqi Cheng, Songbo Tan, Yue Liu, and Huawei Shen. 2013. Aspect-level opinion mining of online customer reviews. *China Communications* 10, 3 (2013), 25–41.
[75] Xueke Xu, Songbo Tan, Yue Liu, Xueqi Cheng, and Zheng Lin. 2012. Towards jointly extracting aspects and aspect-specific sentiment knowledge. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. 1895–1899.
[76] Rui Yan, Jie Tang, Xiaobing Liu, Dongdong Shan, and Xiaoming Li. 2011. Citation count prediction: Learning to estimate future citations for literature. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'11)*. 1247–1252.
[77] Yinfei Yang, Cen Chen, Minghui Qiu, and Forrest Bao. 2017. Aspect extraction from product reviews using category hierarchy information. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*. 675–680.
[78] Zhengxin Yang, Jinchao Zhang, Fandong Meng, Shuhao Gu, Yang Feng, and Jie Zhou. 2019. Enhancing context modeling with a query-guided capsule network for document-level translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 1527–1537.
[79] Zhigang Yuan, Fangzhao Wu, Junxin Liu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2019. Neural review rating prediction with user and product memory. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'19)*. 2341–2344.
[80] Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and S. Yu Philip. 2019. Joint slot filling and intent detection via capsule neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL'19)*. 5259–5267.
[81] Lei Zhang and Bing Liu. 2014. Aspect and entity extraction for opinion mining. In *Data Mining and Knowledge Discovery for Big Data*. Springer, 1–40.
[82] Wenxuan Zhang, Wai Lam, Yang Deng, and Jing Ma. 2020. Guided helpful answer identification in e-commerce. In *Proceedings of the 2020 Web Conference (WWW'20)*. 2620–2626.
[83] Xuan Zhang, Zhilei Qiao, Aman Ahuja, Weiguo Fan, Edward A. Fox, and Chandan K. Reddy. 2019. Discovering product defects and solutions from online user generated contents. In *Proceedings of the World Wide Web Conference (WWW'19)*. 3441–3447.
[84] Yating Zhang, Adam Jatowt, and Katsumi Tanaka. 2016. Causal relationship detection in archival collections of product reviews for understanding technology evolution. *ACM Transactions on Information Systems* 35, 1 (2016), 1–41.
[85] Cheng Zhao, Chenliang Li, Rong Xiao, Hongbo Deng, and Aixin Sun. 2020. CATN: Cross-domain recommendation for cold-start users via aspect transfer network. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20)*.

[86] Lujun Zhao, Kaisong Song, Changlong Sun, Qi Zhang, Xuanjing Huang, and Xiaozhong Liu. 2019. Review response generation in e-commerce platforms with external product information. In *Proceedings of the World Wide Web Conference (WWW'19)*. 2425–2435.

[87] Qihang Zhao. 2020. Utilizing citation network structure to predict citation counts: A deep learning approach. arXiv:2009.02647.

[88] Wei Zhao, Haiyun Peng, Steffen Eger, Erik Cambria, and Min Yang. 2019. Towards scalable and reliable capsule networks for challenging NLP applications. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL'19)*. 1549–1559.

[89] 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 European Conference on Information Retrieval (ECIR'11)*. 338–349.

[90] Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP'10)*. 56–65.

Received August 2020; revised March 2021; accepted May 2021