IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 4 September 2022, accepted 30 October 2022, date of publication 21 November 2022, date of current version 5 January 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3224019

APPLIED RESEARCH

NeurReview: A Neural Architecture Based Conformity Prediction of Peer Reviews

JIE MENG
Innovation Academy for Microsatellites, University of Chinese Academy of Sciences, Huairou, Beijing 101408, China
e-mail: moonjaymengjie@gmail.com

ABSTRACT Peer review is at the heart of scholarly communications and the foundation of scientific evaluation. However, peer review's effectiveness is continuously challenged due to biased and inconsistent peer reviews. Consequently, ensuring the quality of peer reviews is a time-critical problem. In this paper, we investigate the conformity between reviews and meta-reviews. To predict the review conformity and identify the effective features to distinguish the misaligned reviews, we propose NeurReview, which models the review process from the review structure and interactions with authors and other reviewers. Two evaluation datasets are constructed from the ICLR open reviews. The evaluation results verified the efficacy of our proposed model. In addition, we found that the divergence with other reviews and responses, the consistency of sentiment polarity with the recommendation score, etc., are beneficial features for identifying low-conformity reviews, which can assist meta-reviewers in making final decisions.

INDEX TERMS Aspect mining, classification, natural language processing, peer review, review quality.

I. INTRODUCTION

Peer review lies at the foundation of scientific evaluation. However, the effectiveness of peer review is being continuously questioned. Researchers often argue with the quality [1] and reproducibility [2] of the peer-review system due to some biased and inconsistent peer reviews. The well-known NIPS experiment [3] observed observation that 43% of papers accepted to the conference would be accepted again if the conference review process were repeated. The bias of the reviewers [4] leading to inconsistencies between their review reports further aggravates the problem. Apart from that, peer review has also been challenged by the rapid increase of paper submissions. Consider the example of computer science conferences: The Conference on Neural Information Processing Systems (NeurIPS) received 9467 submissions in 2020, which is five times the number of submissions it received in 2010 [5]. The exploding nature of paper submissions leads to paper-vetting by less-experienced researchers from disparate fields due to the shortage of qualified reviewers. The consequences are often unsatisfactory, occasionally leading to substandard research finding a place and good research papers being ignored. Thus, evaluating the quality of peer reviews is a critical problem.

As far as we consider, evaluating peer review quality is rather complicated. It has received attention from multiple perspectives such as justification [5], bias [6], comprehensiveness [4]. In our present work, we would rather focus on the conformity of reviews with meta-reviews, and the corresponding text features. An essential task in the review process occurs when the meta-reviewers must determine whether or not to accept a manuscript based on the opinions of several reviewers and papers. Many unqualified reviews pose a big burden on the meta-reviewers since they not only have to handle submissions but also have to carefully validate the reviews in terms of consistency and justification, etc. The load could be reduced if we can automatically detect low-conformity reviews and further reveal the related features such as aspect coverage, sentiment polarity. It helps the chairs allocate attention to different reviews, allowing them to place a greater emphasis on more worthy reviews and write comprehensive meta-review when making final decisions.

To address the above problem, we propose and investigate a novel review conformity prediction task. We consider modeling conformity directly from the text to build an analyzable and interpretable framework for predicting and making

The associate editor coordinating the review of this manuscript and approving it for publication was Agostino Forestiero.

TABLE 1. An example of peer review text associated with corresponding aspects.

Abstract	Deep networks realize complex mappings that are often understood by their locally linear behavior at or around points of interest. One key challenge is that such derivatives are themselves inherently unstable. In this paper, we propose a new learning problem to encourage deep networks to have stable derivatives over larger regions....
Review-1	The paper considers deep nets with piecewise linear activation functions.[Summary]...With the exception of some minor typos, the exposition is clear and the theoretical claims all appear correct.[Correctness]... A basic complexity analysis or running time comparison would help clarify this.[Clarity]... As always, gradient-based penalties suffer from heavy computational overhead.
	Response:We thank all the reviewers for the insightful comments, suggestions. We believe establishing robust derivatives is important in its own right: stable derivatives serve many roles, including interpretability, but require extra effort to achieve in deep models. Note that robustness of explanations is an open problem in the community.
	Confidence: 5 Rating: 7: Good paper, accept
Review-2	This is a very relevant and timely work related to robustness of deep learning models under adversarial attacks[Summary]... The visualizations show the stability properties nicely, but a bit more explanations of those figures would help the readers...
	Response: ...
	Confidence:3 Rating: 5: Marginally below acceptance threshold
Review-3	....
Meta-Review	The paper aims to encourage deep networks to have stable derivatives over larger regions under networks. All reviewers and AC note the significance of the paper. ...
	Recommendation: Accept (Poster)

connections between the text features and the consistency. Peer review analysis has been investigated in the literature in recent years [7], [8], [9]. Most of them modeled the review text only to predict recommendation scores. However, these methods still suffer from some limitations on review process modeling due to the challenges posed by the complexity of the review process. For illustration, the above block in Table 1 is an example review. (Due to space limit, we have to intercept a small fragment here as an example). When modeling the review process, three challenges will be faced:

- The complexity of the review text. The reviews are expected to follow a well-defined structure, typically starting with a summary of the paper's contributions, then with opinions accessing the quality of a submission from different aspects such as originality and motivation. Reviewers should make constructive criticisms over certain crucial aspects of the paper in a detailed manner, while bringing out the reviewer's stand on the work. Compared with ordinary text, the structure of reviews is more complicated and professional. Moreover, the long length of review text makes it more challenging to capture semantic features.
- The complexity of the review process. The review process is not isolated and usually involves interaction between authors, reviewers and meta-reviewers. Consider the example in Table 1. First, three or four reviewers are required to write reviews and give recommendation scores for a submission. Then the rebuttal phase allows authors to provide responses to address the criticisms and questions raised in the reviews and to defend their work. Finally, a chair writes a meta-review and gives a final decision. Therefore, evaluating a review is based not only on the text's content but also on disagreements between reviewers, conformity with the submission, and inconsistency with the response.
- Multi-source information. In addition to textual data, other information is included, such as confidence scores, review scores, the change of scores after rebuttal, whether double-blinded or not, and the authors' influence. All of the information can affect the final decision.

In this study, we propose a neural framework (i.e., NeurReview) for review conformity prediction and peer review analysis to tackle the above challenges. Specifically, NeurReview consists of three major components. Deep semantic component extracts deep semantic features and models the interaction between authors, reviewers, including disagreements and agreements with other reviews and responses. Aspect-sentiment component handles the review structure by identifying aspects and sentiment polarity. The peripheral component takes into account other features that are not modeled by the deep network. These three components model the review process from multiple perspectives, thus addressing the above challenges well.

We collect peer review data of ICLR (International Conference on Learning Representations), including ICLR-2017, ICLR-2018, and ICLR-2019. Evaluation results show that our proposed neural network model outperforms several baseline methods and the state-of-art neural models. Moreover, our framework is also analyzable and interpretable, which gains some insights. For example, not-aligned reviews have low correlation between the sentiments expressed in their review text and their decisions. Also, they have less confidence and more disagreements with other reviews. These analysis results can help the meta-reviewers identify potentially inconsistent reviews and further a comprehensive meta-review. And we believe the framework will also contribute to building an automated peer review system.

II. RELATED WORK

A. PEER REVIEW ANALYSIS

As an important paper evaluation mechanism, peer review has been widely adopted in various journals and conferences [10], [11]. Most works on peer review before 2017 were limited to a handful of papers due to the absence of a public domain peer review dataset with sufficient data points. Researchers have explored the usefulness of peer reviews in several aspects based on private review datasets. Xiong et al. examined whether standard product review analysis techniques also apply to our new context of peer reviews [12]. They also proposed an evaluation system that generates assessments on reviewers' reviewing skills regarding the issue of problem

localization [13]. Gender bias in peer-review data has been studied in [6].

More recently, Kang et al. [14] have collected and analyzed openly available peer review data PeerRead for the first time. They also provided several baselines defining major tasks. Based on this dataset, Wang and Wan [15] have employed peer review text to predict the overall decision status for sentiment analysis and recommendation score prediction by using a Multiple Instance Learning Framework with attention mechanism [16]. Gao et al. [17] focused on the role of the rebuttal phase, and proposed a novel task to predict after-rebuttal scores from initial reviews and author responses.

B. REVIEW ASSESSMENT

Relatively little effort can be found on developing automatic tools for review quality assessment and prediction. Methods for review assessment have been mainly developed for e-commerce and education science based on the dataset from online rating platforms [12], [18], [19], [20].

Evaluation and prediction for scholarly reviews is more challenging due to the lack of available datasets for model training, the cognitive complexity of the task, and the highly specialized topic. David Tran et al. quantified reproducibility in the review process by Monte-Carlo simulations [2]. Ines Arous et al. proposed a Bayesian framework that integrates a machine learning model with peer grading to assess the conformity of scholarly reviews [5]. To evaluate the generated review, Yuan et al. proposed a variety of diagnostic criteria for review quality, including review aspect coverage and informativeness [4]. Falkenberg et al. analyzed the characteristics of review text that distinguish high-quality reviews from lower-quality reviews for editors, but only a small sample of reviews was investigated [21]. In summary, the majority of studies on peer review have focused on evaluating papers based on peer review data, such as decisions [14], aspect scores [22], and citation prediction [23]. Comparatively, little research has focused on peer review quality.

III. METHOD

In this section, after formulating the task as a text classification problem, we propose a NeurReview framework modeling reviews with multi-level features, shown in Figure 1. NeurReview mainly consists of three components, i.e., the deep semantic component, aspect-sentiment component and wide component, which model semantic features, aspect and sentiment features and other handcrafted features, respectively. The notations and descriptions are shown in Table 2.

A. PROBLEM FORMULATION

Let I be the set of reviews, for each review $i \in I$, our goal is to learn a classifier ξ which is able to predict the review conformity:

$$\xi(rev_i, res_i, a_i^p, rev_{\{j!=i\}}^p, x_i) \rightarrow \hat{cf}_i$$

TABLE 2. Notions.

Notation	Description
$\{S_{i,j}^r\}_{j=1}^n$	the representation of the i-th review
$\{S_{i,j}^a\}_{j=1}^n$	the representation of the i-th abstract
$\{S_{i,j}^p\}_{j=1}^n$	the representation of the i-th response
D	the orthogonal decomposition map
$\boldsymbol{PE_i}$	the peripheral features of the i-th review
$\widetilde{\boldsymbol{R}}_{\boldsymbol{i}}$	the deep semantic vector of the i-th review
$\boldsymbol{AS_i}$	the aspect-sentiment vector of the i-th review
cf_i	the conformity of i-th review

where rev_i,res_i,a_i^p represents the review text, the corresponding response text and the abstract text of submission p. We also take account into other reviews of the submission p, $rev_{\{j!=i\}}^p$. We assume that all these texts share the same vocabulary V. Besides these text features, we also assume other types of information (e.g., confidence scores) are also available for our task, which is represented by x_i. We consider the ground truth of a review conformity $\hat{cf}_i$ as a binary variable indicated by the alignment between a reviewer decision and the meta-reviewer decision: when both the reviewer and the meta-reviewer decide to accept or reject a submission, the ground truth for the review is set to 1 (align), otherwise to 0 (misalign). Therefore, this task can be regarded as a text classification problem.

B. DEEP SEMANTIC COMPONENT

Deep semantic component aims to learn a deep semantic representation, including the semantics of the review text itself and the interaction with other reviews, abstracts and responses. First, we employ the embedding layer to transform each sentence of the review text and the response text into distributed representations. Next, the convolution layer is used to learn local semantics within a sentence and then a bidirectional LSTM is used to obtain the global and high-level representation of the entire document. Then, we propose the semantic decomposition layer to separate consensus and divergent semantics based on the vectors from LSTM layer. Finally, we concatenate the review vectors and the decomposed vectors.

1) EMBEDDING LAYER

The inputs of our model are the word sequences of the reviews text, abstract text and response text of the review. These texts contain n sentences, and each sentence is composed of several words. We first pretrain the word embeddings using the word2vec model using all the scientific corpus [24], representing each word $w_i \in \mathbb{R}^d$ as a fixed-size vector, where d is the dimension of the word vector. Due to different sizes of texts, we set L as the maximum number of words in a sentence. A sentence is then represented as,

$$S = \boldsymbol{w}_1 \oplus \boldsymbol{w}_2 \oplus \cdots \oplus \boldsymbol{w}_L, S \in \mathbb{R}^{L \times d} \quad (1)$$

where $\oplus$ is the concatenation operator. Thus we represent the review, abstract and response text as $\{S_i^r\}$,$\{S_i^a\}$,$\{S_i^p\}$ respectively.

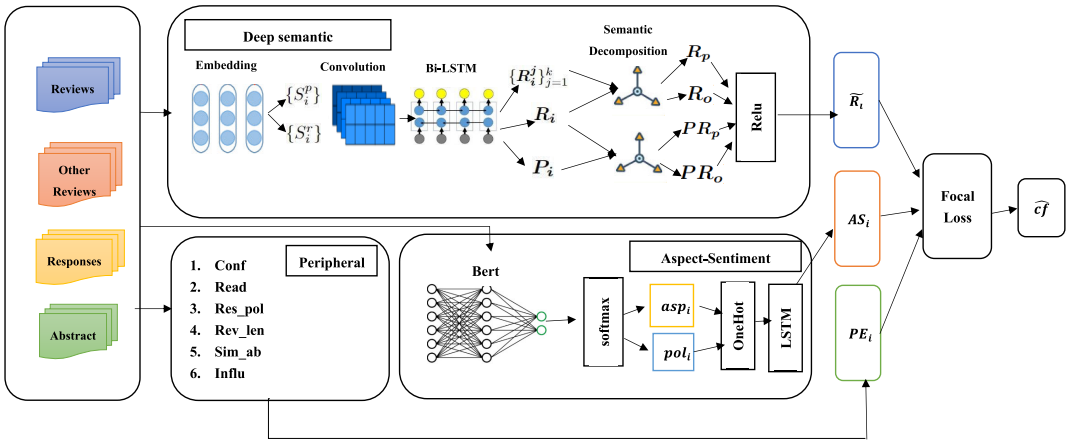


FIGURE 1. Overview of our NeurReview framework.

2) CONVOLUTION LAYER

To effectively encode the review and response sentences, we apply the convolution layer to get high-level representations and capture more abstract and semantic features in addition to sequential information of the input texts. Formally, we use a convolution kernel W_c to perform a convolution operation within each sliding window:

$$f_k = \tanh\left(W_c \cdot W_{k-l+1:k} + b_c\right) \tag{2}$$

Tanh is used as a nonlinear activation function. W_c is the convolution matrix, $W_{k-l+1:k}$ denotes the concatenation of of l word embeddings within the k-th window in word sequences and b_c is the bias. After applying convolution to each possible window of words, we produce a feature map. Then a max-pooling operation is used to obtain the most significant feature:

$$u_q = max\_pool\{f_1^{(q)}, f_2^{(q)}, \cdots, f_{L-l+1}^{(q)}\} \tag{3}$$

where f^q is the output of q-th filter. We use multiple filters so as to extract n-gram features. The output of this layer is the concatenation of each u_q: $[u_1, u_2, \ldots u_m]$. The outputs of the convolution layer of reviews and responses of submission i are denoted as $\boldsymbol{R_i^c}$, $\boldsymbol{P_i^c}$, respectively.

3) BIDIRECTIONAL-LSTM LAYER

In this layer, we aim to obtain the global and high-level representation of the entire document. The feature sequences obtained from the convolution operation in the preceding layer is the deficit in providing the sequential information. We use a bidirectional-LSTM to extract sequential information and detect long sequential patterns. We use separate LSTM modules to produce forward and backward hidden vectors

$$\begin{aligned} \overrightarrow{h_i} &= \overrightarrow{\text{LSTM}}\left(v_i\right) \\ \overleftarrow{h_i} &= \overleftarrow{\text{LSTM}}\left(v_i\right) \\ h_i &= \overrightarrow{h_i} \,\|\, \overleftarrow{h_i} \end{aligned} \tag{4}$$

where v_i denotes $\boldsymbol{R_i^c}$ or $\boldsymbol{P_i^c}$ and the last equation represents that h_i is the concatenated results of the above hidden vectors. All the sentences appear in peer reviews have different contributions for prediction. Therefore, we use the attention mechanism to enforce the model to attend the important part of the reviews by assigning different weights to individual sentences. The importance of each sentence is measured as follows

$$\begin{aligned} h_i' &= \tanh\left(W_a \cdot h_i + b_a\right) \\ a_i &= \frac{\exp\left(h_i'\right)}{\sum_j \exp\left(h_j'\right)} \end{aligned} \tag{5}$$

This equation produces an attention weight a_i for the i-th sentence. After weighted summing, we can obtain the output of this layer, denoting as $\boldsymbol{R_i}$,$\boldsymbol{P_i}$, respectively.

4) SEMANTIC DECOMPOSITION LAYER

When making final decisions, meta-reviewers usually compare the statements of different reviewers. The coverage and divergence can be considered for the acceptance decision of a paper. Besides, authors' responses to reviewers can have an effect on the final decision. An effective rebuttal reflects the quality of the reviews and the accuracy of the recommendations.

Given a submission, we take account into other reviews of the submission $\{\boldsymbol{R_i^j}\}_{j=1}^k$. We propose the semantic decomposition layer to learn consensus and divergent semantics between the target review text R_i and other reviews. Considering that the disagreement between review and response also reflects the recommendation bias of the reviewer to the submission, we also add the response text P_i to the input data, and then extract divergence features.

Inspired by [25] and [26], we separate vectors into similar and dissimilar components based on sentence similarity learning method. To be specific, two phases are included: decomposition and composition. In the decomposition phase, we define a decomposition operator using

orthogonal decomposition strategy $D : (v, \tilde{v})$:

$$D_p : v_p = \frac{v \cdot \tilde{v}}{\tilde{v} \cdot \tilde{v}} \tilde{v}, \tag{6}$$

$$D_o : v_o = v - v_p, \tag{7}$$

where $(v, \tilde{v})$ is decomposed into parallel vectors v_p and the orthogonal ones v_o. Among them, v_p could be seen the similar component, and v_o could be consider as the dissimilar component.

we firstly use the decomposition operator to learn consensus and divergent semantics among various reviews: $(R_p, R_o)^j = D(R_i, R_i^j)$. Next, we apply mean-pooling over to obtain the final similar and dissimilar component.

$$(\boldsymbol{R_p}, \boldsymbol{R_o}) = \frac{1}{j} \sum_{j=1}^{k} D(\boldsymbol{R_i}, \boldsymbol{R_i}^j) \tag{8}$$

Similarly, we can also obtain the converge and diverge features between the response text and the review text: $(PR_p, PR_o) = D(P_i, R_i)$. Afterward, we combine all decomposed component by a fully connected layer to get the refine representation:

$$\widetilde{\boldsymbol{R_i}} = Relu(W * [\boldsymbol{R_p}, \boldsymbol{R_o}, \boldsymbol{PR_p}, \boldsymbol{PR_o}] + b) \tag{9}$$

C. ASPECT-SENTIMENT COMPONENT

The reviewers largely follow a well-defined structure while writing reviews identifying the pros and cons of the paper. A good review should be well-organized, typically starting with a brief summary of the paper's contributions, then following with opinions gauging the quality of a paper from different aspects, together with evidence [4]. Moreover, a peer review should be sentiment rich and the reviewers would tend to express varying sentiments across various aspects.

The fine-grained text structure and sentiment information lays an essential role in review evaluation. For example, a comprehensive review should touch on the quality of different aspects of the paper; for an informative review, its negative sentences should be accompanied by corresponding evidence. The structural information reflects reviewers' professionalism, and thus relates to the accuracy of the recommendation. Therefore, we extract sequential aspect and sentiment features.

In this paper, we adopt the reviewing aspects used in the ACL review guidelines,[1] which are Summary (SUM), Motivation (MOT),Originality (ORI), Soundness (SOU), Substance (SUB), Replicability (REP), Meaningful Comparison (CMP) and Clarity (CLA). We also take into account the sentiment polarity for each aspect, which is positive or negative (except summary). All summary sentences are neutral.

Our goal is to identify the aspect and polarity for each sentence of a review. The burden required for annotating each sentence manually is heavy. To this end, we hope to train a tagger to annotate automatically. The annotation process can be formulated as a sequence labeling problem where the input is a sentence consisting of n words $S = \{w_1, w_2, \ldots, w_n\}$. Yuan et al. have conducted manual annotation for sample reviews of ICLR, ACL [4]. Taking their annotations as labels, we can annotate the rest of the data using a pre-trained model. Specifically, the architecture of aspect-sentiment component contains a pre-trained model BERT [27] and a multi-layer perceptron. The BERT model is used to get a contextualized representation for each token

$$\boldsymbol{e_i} = BERT(w_i)$$

Then a multi-layer preceptron with softmax activation function can be used for token classification,

$$\mathbf{p}_i = \text{softmax}\,(\mathbf{W}\boldsymbol{e_i} + \mathbf{b})$$

where W and b are tunable parameters of the multilayer perceptron. $\boldsymbol{p_i}$ is a vector that represents the probability of token i being assigned to different aspects. After training with the negative log likelihood loss function, an aspect-sentiment tagger Φ is constructed. We tag each sentence in the review vector $\{S_j^r\}_{j=1}^n$ as a tuple of aspects and sentiments with the tagger Φ.

$$(asp_j, pol_j) = \Phi(S_j^r)$$

where asp_j and pol_j represent the aspect and polarity of j-th sentence,respectively. After that, One-hot encoding method is used to encoding the discrete features. Recall that we have eight aspects and three sentiment polarity, we construct an 11 dimensional aspect-sentiment vector. Then we employ a LSTM to capture sequential information:

$$\boldsymbol{AS_i} = LSTM(\text{OneHot}\{(asp_j, pol_j)\}_{j=1}^n)$$

D. PERIPHERAL COMPONENT

In addition to the central features extracted by the deep network, we also consider integrating other hand-engineered features as peripheral features. We propose the following six categories of feature, which is denoted as $\boldsymbol{PE_i}$.

- **Confidence Score** *(Conf)*. We use all confidence scores given a submission to build an array of score-based features. These include their difference with the scores of the other reviews on the same paper.
- **Readability** *(Read)*. The New Dale-Chall (NDC) Readability Formula [28] is used to calculate readability. Unlike other formulas that use word-length to assess word difficulty, the NDC is based on the number of words per sentence and the proportion of difficult words that are not part of a list of ''common words''. The smaller the readability score, the easier it is to be understood.
- **Sentiment Polarity of Response** *(Res_pol)* We use the sentiment polarity of the response as a feature because it captures the author's perspective on the review. In this work, a widely applied sentiment analysis techniques, NaiveBayesAnalyzer, is adopted to learn sentiment polarity. The sentiment analyzer is implemented by a Python module NLTK and trained on a reviews corpus

[1] https://acl2018.org/downloads/acl 2018 review form. html

for sentiment analysis using the Naïve Bayes classifier method [29]. The polarity score is ranging from -1 to 1, where -1 represents the most negative sentiment while 1 represents the most positive sentiment.

- **Review Length** *(Rev_len)* We use the logarithm of the number of tokens of reviews as a feature.
- **Review-Abstract Similarity** *(Sim_ab)* The first step of a review is summarizing the paper's contribution. A Lack of similarity between an abstract and its review may indicate that the review is ''off-topic''. Therefore, the similarity with abstract can be used to measure the review quality. We use the cosine similarity formula to calculate it. The smaller the angle between the two vectors, the higher the cosine similarity between the two documents. To be specific, the first step is embedding each document as a vector. A dictionary is extracted from the corpus. Then we compute the term frequency and inverse document frequency using TF-IDF method. In this way, we represent the document including abstracts and reviews as sparse vectors, a_i,r_i. The similarity between reviews and abstracts can be calculated as

$$sim\_ab = \frac{< a_i, r_i >}{|a_i||r_i|}$$

- **Author Influence** (*Influ*) Reviewers may be biased because of the different influence of the authors. We use the number of authors and the citations of the author as the influence features.

E. PREDICTION AND TRAINING

Finally, we integrate these three components into a unified model. we consider the recommendation accuracy prediction based on the deep semantic vectors, aspect-sentiment vectors and wide vectors as:

$$y = softmax\left(\boldsymbol{w}_{\text{deep}}^{\top} \cdot \widetilde{\boldsymbol{R}}_i + \boldsymbol{w}_{\text{peri}}^{\top} \cdot \boldsymbol{PE}_i + \boldsymbol{w}_{\text{as}}^{\top} \cdot \boldsymbol{AS}_i + b\right),$$

where w and b are the parameters to learn. Considering the data is unbalanced, we use focal loss cross-entropy loss function to train this model [30]:

$$\text{FL}\,(p_t) = -\alpha\,(1 - p_t)^{\gamma}\log\,(p_t)$$

where p_t is cross entropy loss function for binary classification. Focal loss applies a modulating term to the cross entropy loss in order to focus learning on hard misclassified examples.

IV. EXPERIMENT

A. DATASET

Peer review data is not publicly available for the majority of mainstream journals and conferences. Fortunately, several conferences and workshops have undergone open peer review. OpenReview[2] is a primary source of data, which provides open access to reviews and evaluation scores for all submissions of ICLR. We have crawled 5,575 submissions and 17,028 official reviews from ICLR 2017-2020 venues on the OpenReview. We use the review data since 2017 because the data before 2017 is very noisy and incomplete. For each submission, we include the following metadata information that we can obtain from the review web page: abstract text, review text, meta-review text, response text, review scores, confidence scores and decisions. Besides, multiple sources to enable an analysis of many factors are also scraped, including the citation data from Semantic Scholar.[3] This includes citations for individual papers and individual authors, in addition to the publication counts of each author. ICLR-2017, ICLR-2018 and ICLR-2019 contain 489, 910 and 1418 submissions respectively. We merged ICLR-2017 into the ICLR-2018 dataset due to the inconsistent size of the data for these three years. Finally, we complie the ICLR-2018 (2017,2018) and ICLR-2019 (2019) as our evaluation datasets. A summary of statistics of these two datasets is shown in Table 3.

TABLE 3. The statistics of datasets.

Dataset	ICLR-2018	ICLR-2019
# Reviews	4234	4329
# Submissions	1399	1418
# Sentences	80906	87564
# Words	11059	13126
# Acc	632	478
# Rej	729	916

B. EVALUATION METRICS

To evaluate the performance of different methods on the task, we adopt five evaluation metrics. Typically, the evaluation metrics for the binary classification task include macro-precision (MP), macro-recall (MR), and macro-F1 (F1). Moreover, we adopt other two evaluation metrics for imbalanced data, including AUC and average (AP).

AUC represents the area under receiver operating characteristic curve (ROC). ROC is a probability curve and AUC represents the degree or measure of separability, which ranges from 0.5 to 1. It tells how much the model is capable of distinguishing between classes. The average precision (AP) is a way to summarize the precision-recall curve into a single value representing the average of all precisions. The AP is calculated as the weighted mean of precisions at each threshold, the weight is the increase in recall from the prior threshold.

C. BASELINE

We compare our proposed model against a number of baseline methods, including some of state-of-art deep learning methods.

- **SVM**: Support Vector Machine (SVM) is a kernel machine method that has been widely used for classification in many text domains. We use the unigrams and bigrams in the review text as features and then use SVM to train a text classifier. The feature weight is a

[2]https://openreview.net/

[3]https://www.semanticscholar.org/

TABLE 4. Performance comparison with baseline methods.

Model	ICLR 2018					ICLR 2019				
	AUC	AP	MP	MR	F1	AUC	AP	MP	MR	F1
SVM	0.584	0.884	0.553	0.584	0.546	0.619	0.893	0.567	0.622	0.458
RF	0.551	0.871	0.560	0.552	0.554	0.628	0.895	0.560	0.634	0.547
CNN	0.702	0.906	0.615	0.712	0.574	0.676	0.910	0.684	0.585	0.493
CNN+Bi-Lstm	0.751	0.921	0.636	0.751	0.610	0.721	0.920	0.621	0.726	0.634
DeepSenti	0.676	0.910	0.582	0.682	0.494	0.661	0.906	0.576	0.661	0.523
MILAM	0.762	0.934	0.626	0.721	0.598	0.758	0.931	0.662	0.764	0.681
NeurReview	**0.835**	**0.952**	**0.689**	**0.841**	**0.700**	**0.821**	**0.947**	**0.713**	**0.826**	**0.752**

binary value indicating the occurrence of the unigrams and bigrams.

- **RF**: Random Forest (RF) is an ensemble learning method for classification, which shows good performance in the text-related task. Here, we also extract the unigrams and bigrams from the review text as the input features.
- **CNN**: Convolutional Neural Network (CNN) is widely used for text classification [31], which can extract high-level semantic features of text, especially local contextual features. And CNN have also demonstrated excellent performance.
- **CNN+Bi-LSTM**: In this method, a hierarchical neural network is used. First of all, CNN is used to learn a presentation for each sentence in the review text, and then a bidirectional LSTM and the attention mechanism are used to obtain the high-level representation of the entire document based on the sentences' representations.
- **DeepSenti**: DeepSenti is a deep neural architecture based system, which takes into account three channels of information: the paper, the corresponding peer review texts, and the review polarity to predict the overall recommendation score and the final decision of the scholarly submissions [7].
- **MILAM**: It is a multiple instance learning network with a novel abstract-based memory mechanism to predict the overall decision (accept, reject, or borderline) and further identify the sentences with positive and negative sentiment polarities based on review text [15].

Finally, to further validate the performance of each component in our model, we also design some simplified variants, including:

- **RM-Deep** removes the deep semantic component.
- **RM-AspSen** removes the aspect-sentiment component.
- **RM-Peri** removes the peripheral component.
- **RM-Decom** removes the semantic decomposition layer in the deep semantic component.

D. EXPERIMENT SETUP

First, when extracting labels, reviewers' decisions are classified into two classes: accept ($1 \leq$ score ≤ 5) and reject ($6 \leq$ score ≤ 10). Afterward, we adopt the word2vec [32] to pre-train word embeddings with embedding size 200. We set the maximum sentences in a document to 100. The word vectors were fixed during the training process. The number of CNN filters and LSTM hidden state units are set to 36 and the sizes of MLP hidden units are all set to 32. Two parameters of focal loss are set to $\alpha = 0.35$, $\gamma = 2.5$, respectively. For training, the learning rate of the Adam optimizer [33] is initialized as 0.001. We utilize PyTorch to implement the proposed model.

V. RESULT AND ANALYSIS

A. COMPARISON AGAINST BASELINES

From the comparison results in Table 4, we can see that our proposed model outperforms all baseline methods. The auc and f1-score achieved by our model are promisingly high, exceeding 80% and 70% over the two datasets. The two baselines only utilize n-grams with traditional machine learning models (SVM, RF) perform worse than the other four deep learning baselines on most metrics. It indicates that the deep model can learn a better representation.

Interestingly, the basic CNN methods do not perform well, while the use of the hierarchical Bi-LSTM architecture and attention mechanism can much improve the performance, demonstrating that the hierarchical Bi-LSTM architecture and attention mechanism can be used to learn a better document representation and capture more semantic information. Though DeepSenti utilizes sentiment information, it still performs worse than CNN+LSTM since DeepSenti only uses CNN to learn a representation for an entire document. Meanwhile, MILAM performs consistently better than other baselines. It's because MILAM not only learns comprehensive representations but also leverages sentence-level sentiment information and abstracts. Compared with MILAM, NeurReview can obtain sentence-level aspects and sentiment information. It also leverages other reviews and responses to learn diverge semantics, which is the key to the performance improvement over baselines.

B. COMPARISON AGAINST VARIANTS OF NeurReview

Here we want to investigate the contribution of each part in this task. First, we examine the performance of the model variants by removing each component from the complete model. The result of NeurReview and its four variants are shown in Fig. 2. As we can see, all components are useful to improve the final performance.

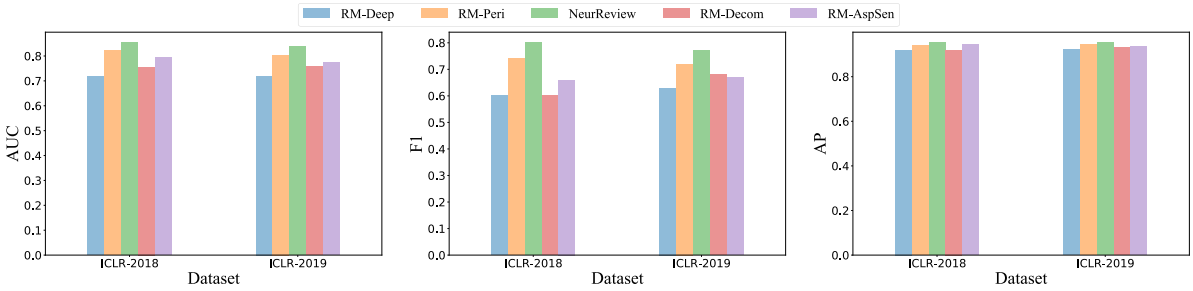


FIGURE 2. Results of NeurReview and its variants.

We observe that the RM-Peri outperforms the RM-Deep by 10.3% AUC and by 14.0% F1-score on average. This result indicates that the deep model captures more semantic representations than the RM-Deep. We also observe that NeurReview outperforms RM-Peri and RM-AspSen by 4.1%(6.5%) AUC and 5.0%(10%) F1-score. The result shows the complementary predictive power of hand-engineered features and aspect-based sentiment features, i.e., the structure information, sentiment polarity and peripheral-level features also improve the model performance substantially.

We further examine the contribution of the main module in deep semantic component: the semantic decomposition layer. It can be seen that the incorporation of the decomposition module improves the model performance more significantly, by 8.9% on the F1 and 8.0% on the AUC. The result indicates that the semantic decomposition function learns the disagreement and consensus among the reviewers, which is the key to prediction.

C. ANALYSIS OF DECOMPOSED LAYER

Our model adopts the semantic decomposition method to learn consistent and divergent features among various reviews. As we can see from the previous section, this module considerably improves the model's performance. To intuitive understand this module and the similar, dissimilar component, we compute the cosine similarity between the vector representation of review R_i and the similar component R_p (s_p), dissimilar component R_o (s_o) of other reviews. Table 5. illustrates the results on a sample of reviews, where Review-0 are the misaligned reviews, Review-1 and Review-2 are aligned reviews. As we can see, reviews with more common aspect have higher s_p, and tend to be high-conformity reviews as well. In contrast, misaligned reviews have higher s_o with other reviews, suggesting that low-conformity reviews have more disagreements with other reviews. The above case analysis demonstrates that this layer could well separate agreements and disagreements.

To quantitatively understand the divergence and similarity of the various types of reviews, we further calculated $\frac{s_o}{s_p}$ on ICLR-2019, a higher value of which indicates greater disagreement with other reviews. As illustrated from Fig. 3, there is a noticeable difference between the aligned and misaligned reviews, and their averages are 0.137, 0.109, respectively.

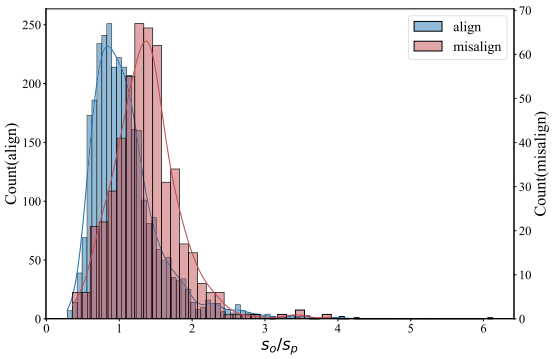


FIGURE 3. Histogram of s_o/s_p for the two types of reviews.

Wilcoxon test showed that the result is significant ($p = 1.78 * 10^{-48} < 0.05$). This means that divergence is the key to distinguishing between the two types of reviews and the final decision complies with the principle of majority rule to some extent.

D. ANALYSIS OF ASPECTS AND SENTIMENTS

1) CLASSIFICATION RESULT

Overall, we achieved 88.45% aspect precision and 83.14% aspect recall. We further presents the aspect-level sentiment predication performance in Table 6. Note that the scores for replicability are lower than the scores of the other aspects. This is due to the fact that there are lower number of training instances.

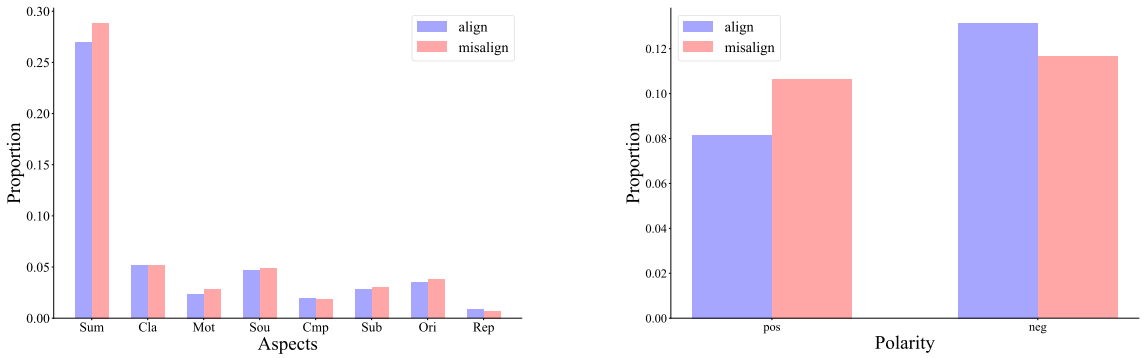
The results show that the tagger's fidelity is reasonably good and that it can be reliably used to annotate the entire dataset. Consequently, we annotate the all sentences in the dataset and perform a further analysis.

2) EVALUATIONS OF ASPECT-BASED METRICS

We first compare the distributions of the sentiments and aspects. (Fig. 4). An interesting finding is that the misaligned reviews have a higher percentage of summaries, 15% higher than aligned reviews. At the same time, the proportion of other aspects is lower than that of aligned samples. At the same time, the proportion of negative comments of low-conformity reviews is lower than that of high-conformity reviews(11.6% vs. 13.3%). The results indicate

TABLE 5. Samples of the reviews. $s_p(s_o)$ represents the cosine similarity of similar component(dissimilar component) between Review-0 and Review-1(Review-2).

Review-0	s_p/s_o	Review-1	s_p/s_o	Review-2
The choice of LSTMs is understandable but other experiments could have been done in order to make clearer why it has been chosen.	0.22/0.83	comparison against 'easy to beat' baselines. The comparison should also include as baselines the very relevant methods listed in the last paragraph of the related work section.	0.33/0.76	Slightly unfair baselines? One of the first things that struck me in the experimental results was how competitive word2vec by itself was across all three datasets.
The authors should make this dataset available for replicability. Also, why have the authors not used this embedding for eval on standard datasets.	0.21/ 0.73	There isn't a clear evaluation that shows the utility of the added OOV Handler. And the coreference experiment isn't that clearly described.	0.18/0.86	The introduction and related work part are clear with strong motivations to me. But section 4 and 6 need a lot of details.



(a) Aspect distribution

(b) Sentiment distribution

FIGURE 4. The distributions of aspect and sentiment.

that low-conformity reviews have more summaries and relatively fewer comments on other aspects, especially negative ones. As mentioned in the introduction, this may be due to the reviewer's lack of relevant background knowledge.

To further understand the relationship between aspects, sentiments and recommendations, inspired by [4] and [22], we calculate two aspect-based indicators:

- **Aspect Coverage** (ACOV). Given a review R, aspect coverage measures how many aspects (e.g. clarity) in a predefined aspect typology have been covered by R.
- **Sentiment Consistency** (SCON). SCON measures the consistency of sentiment polarity with recommendation scores, which is calculated by Spearman correlation between the sentiment scores with the recommendation scores given by a reviewer.

For ACOV, we calculate each group's average value and perform the Mann-Whitney U test. The results of this analysis are shown in Table 7. We found that aligned reviews have higher SCON, and the p-value showed the result is significant, which indicates that the sentiment of an aligned review is more consistent with the recommendation scores. Although the mean ACOV of aligned reviews is slightly higher, the statistical test result showed no significant difference in ACOV between the two types of reviews.

E. ANALYSIS OF PERIPHERAL COMPONENT

In order to study the importance of various features in the peripheral component when predicting, we investigate

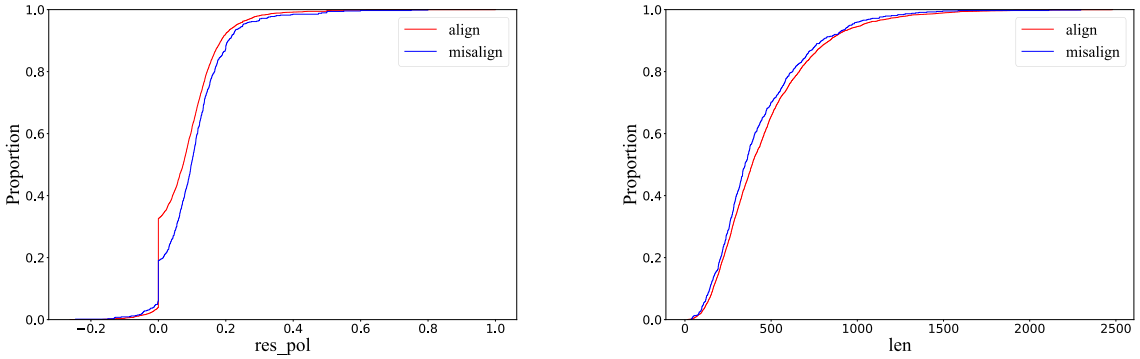
TABLE 6. Result of the aspect-level sentiment predication.

Aspects	Precision	Recall
CLA	0.92	0.84
ORI	0.87	0.82
CMP	0.82	0.93
REP	0.78	0.72
MOT	0.90	0.79
SUB	0.84	0.82
SOU	0.88	0.91

TABLE 7. Aspect-based metrics.

Metric	Review Type	value	p-value
SCON	align	0.636	0.00
	misalign	0.507	$2.84 * 10^{-45}$
ACOV	align	61.3%	0.59
	misalign	60.1%	0.59

the weight of each feature. After normalization, the result is shown in Table 8. For illustration, we also present the cumulative distribution of some features in Fig. 5. From Table 8, *res_pol*,*sim_ab* are considerably more informative than other features. As shown in Fig. 5(a), on average, the emotions expressed by the responses of the aligned reviews (0.092) are more neutral than the misaligned ones (0.131). In addition, misaligned reviews have lower *sim_ab* scores (0.064 vs. 0.045) than the aligned reviews, which implies the low-conformity reviews tend to be more inconsistent with the abstract of the paper. Moreover, *rev_len* and *confi* are



(a) The distribution of sentiment polarity of response

(b) The distribution of review length

FIGURE 5. The cumulative distributions of some peripheral features.

TABLE 8. Peripheral feature importance.

Features	Importance
res_pol	0.394
sim_ab	0.255
len	0.158
confi	0.096
read	0.056
influ	0.041

informative features. The cumulative distribution in Fig. 5(b) shows that the aligned reviews, with 464 words on average, tend to be longer than the misaligned ones that averaged 423 words. In addition, low-conformity reviews have lower confidence scores (3.89 vs. 3.76).

VI. CONCLUSION

In this paper, we proposed a neural framework (i.e., NeurReview) that modeled the review process to address the challenging task of review conformity prediction and analysis. Specifically, three components of NeurReview modeled the interaction among reviewers and authors, the complex review text structure, and other information. We constructed two evaluation datasets from the ICLR open reviews and discussed evaluation results extensively. We verified the efficacy of our proposed model. We found that low-conformity reviews have more disagreements with other reviews and responses, and their sentiment polarity is less consistent with the recommendation scores. They also have a higher percentage of summaries and positive comments. All these observations can help the meta-reviewers identify potentially inconsistent reviews and further a comprehensive meta-review. In future work, we would like to collect more peer reviews for training and testing in different research areas, and we will also try more advanced deep learning techniques. We want to extend this research work to build an efficient AI-enabled system that can assist editors or meta-reviewers in making final decisions and writing meta-reviews automatically.

ACKNOWLEDGMENT

The author would like to thank Prof. Lou for her advice on scientific writing. This work could not be accomplished without the help of Wen Lou.

REFERENCES

[1] S. Van Rooyen, N. Black, and F. Godlee, "Development of the review quality instrument (RQI) for assessing peer reviews of manuscripts," *J. Clin. Epidemiology*, vol. 52, no. 7, pp. 625–629, 1999.

[2] D. Tran, A. Valtchanov, K. Ganapathy, R. Feng, E. Slud, M. Goldblum, and T. Goldstein, "An open review of OpenReview: A critical analysis of the machine learning conference review process," 2020, *arXiv:2010.05137*.

[3] C. Cortes and N. D. Lawrence, "Inconsistency in conference peer review: Revisiting the 2014 NeurIPS experiment," 2021, *arXiv:2109.09774*.

[4] W. Yuan, P. Liu, and G. Neubig, "Can we automate scientific reviewing?" 2021, *arXiv:2102.00176*.

[5] I. Arous, J. Yang, M. Khayati, and P. Cudre-Mauroux, "Peer grading the peer reviews: A dual-role approach for lightening the scholarly paper review process," in *Proc. Web Conf.*, Apr. 2021, pp. 1916–1927.

[6] M. Helmer, M. Schottdorf, A. Neef, and D. Battaglia, "Gender bias in scholarly peer review," *eLife*, vol. 6, Mar. 2017, Art. no. e21718.

[7] T. Ghosal, R. Verma, A. Ekbal, and P. Bhattacharyya, "DeepSentiPeer: Harnessing sentiment in review texts to recommend peer review decisions," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1120–1130.

[8] T. Pradhan, C. Bhatia, P. Kumar, and S. Pal, "A deep neural architecture based meta-review generation and final decision prediction of a scholarly article," *Neurocomputing*, vol. 428, pp. 218–238, Mar. 2021.

[9] T. Ghosal, S. Kumar, P. K. Bharti, and A. Ekbal, "Peer review analyze: A novel benchmark resource for computational analysis of peer reviews," *PLoS ONE*, vol. 17, no. 1, Jan. 2022, Art. no. e0259238.

[10] J. S. Ross, C. P. Gross, M. M. Desai, Y. Hong, A. O. Grant, S. R. Daniels, V. Hachinski, R. J. Gibbons, T. J. Gardner, and H. M. Krumholz, "Effect of blinded peer review on abstract acceptance," *JAMA*, vol. 295, no. 14, pp. 1675–1680, 2006.

[11] M. Fisher, S. B. Friedman, and B. Strauss, "The effects of blinding on acceptance of research papers by peer review," *JAMA*, vol. 272, no. 2, pp. 143–146, 1994.

[12] W. Xiong and D. J. Litman, "Automatically predicting peer-review helpfulness," in *Proc. ACL*, 2011, pp. 502–507.

[13] W. Xiong, D. J. Litman, and C. D. Schunn, "Assessing reviewer's performance based on mining problem localization in peer-review data," in *Proc. 3rd Int. Conf. Educ. Data Mining*, 2010, pp. 211–220.

[14] D. Kang, W. Ammar, B. Dalvi, M. Van Zuylen, S. Kohlmeier, E. H. Hovy, and R. Schwartz, "A dataset of peer reviews (peerread): Collection, insights and NLP applications," in *Proc. NAACL*, 2018, pp. 1–15.

[15] K. Wang and X. Wan, "Sentiment analysis of peer review texts for scholarly papers," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 175–184.

[16] S. Angelidis and M. Lapata, "Multiple instance learning networks for fine-grained sentiment analysis," *Trans. Assoc. Comput. Linguistics*, vol. 6, pp. 17–31, Dec. 2018.
[17] Y. Gao, S. Eger, I. Kuznetsov, I. Gurevych, and Y. Miyao, "Does my rebuttal matter? Insights from a major NLP conference," in *Proc. NAACL*, 2019, pp. 1–18.
[18] W. Zhang, W. Lam, Y. Deng, and J. Ma, "Review-guided helpful answer identification in E-commerce," in *Proc. Web Conf.*, Apr. 2020, pp. 2620–2626.
[19] I. E. Olatunji, X. Li, and W. Lam, "Context-aware helpfulness prediction for online product reviews," in *Proc. AIRS*, 2019, pp. 56–65.
[20] L. Ramachandran and E. F. Gehringer, "Automated assessment of review quality using latent semantic analysis," in *Proc. IEEE 11th Int. Conf. Adv. Learn. Technol.*, Jul. 2011, pp. 136–138.
[21] L. J. Falkenberg and P. A. Soranno, "Reviewing reviews: An evaluation of peer reviews of journal article submissions," *Limnology Oceanogr. Bull.*, vol. 27, no. 1, pp. 1–5, Feb. 2018.
[22] S. Chakraborty, P. Goyal, and A. Mukherjee, "Aspect-based sentiment analysis of scientific reviews," in *Proc. ACM/IEEE Joint Conf. Digit. Libraries*, Aug. 2020, pp. 207–216.
[23] S. Li, W. X. Zhao, E. J. Yin, and J.-R. Wen, "A neural citation count prediction model based on peer review text," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 4914–4924.
[24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. NIPS*, 2013, pp. 1–9.
[25] X. Wang, W. Huang, Q. Liu, Y. Yin, Z. Huang, L. Wu, J. Ma, and X. Wang, "Fine-grained similarity measurement between educational videos and exercises," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 331–339.
[26] Z. Wang, H. Mi, and A. Ittycheriah, "Sentence similarity learning by lexical decomposition and composition," in *Proc. COLING*, 2016, pp. 1–10.
[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[28] J. S. Chall and E. Dale, *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline, MA, USA: Brookline Books, 1995.
[29] C. W. Tseng, J. J. Chou, and Y. C. Tsai, "Text mining analysis of teaching evaluation questionnaires for the selection of outstanding teaching faculty members," *IEEE Access*, vol. 6, pp. 72870–72879, 2018.
[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[31] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1746–1751. [Online]. Available: https://aclanthology.org/D14-1181, doi: 10.3115/v1/D14-1181.
[32] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. ICLR*, 2013, pp. 1–12.
[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2015, *arXiv:1412.6980*.

JIE MENG is currently pursuing the master's degree with the University of Chinese Academy of Sciences. His research interests include informetrics, the science of science, and computational social science.