Exploiting Rich Textual User-Product Context for Improving Personalized Sentiment Analysis

Chenyang Lyu[†] **Linyi Yang**[‡] **Yue Zhang**[‡] **Yvette Graham**[¶] **Jennifer Foster**[†]
[†] School of Computing, Dublin City University, Dublin, Ireland
[‡] School of Engineering, Westlake University, China
[¶] School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland
chenyang.lyu2@mail.dcu.ie, ygraham@tcd.ie, jennifer.foster@dcu.ie
{yanglinyi, zhangyue}@westlake.edu.cn

Abstract

User and product information associated with a review is useful for sentiment polarity prediction. Typical approaches incorporating such information focus on modeling users and products as implicitly learned representation vectors. Most do not exploit the potential of historical reviews, or those that currently do require unnecessary modifications to model architecture or do not make full use of user/product associations. The contribution of this work is twofold: i) a method to explicitly employ historical reviews belonging to the same user/product in initializing representations, and ii) efficient incorporation of textual associations between users and products via a user-product cross-context module. Experiments on the IMDb, Yelp-2013 and Yelp-2014 English benchmarks with BERT, SpanBERT and Longformer pretrained language models show that our approach substantially outperforms previous state-of-the-art.

1 Introduction

It has been repeatedly shown that the user and product information associated with reviews is helpful for sentiment polarity prediction (Tang et al., 2015; Chen et al., 2016; Ma et al., 2017). Just as the same user is expected to have consistent narrative style and vocabulary, the reviews belonging to the same product are expected to exhibit similar vocabulary for specific terms. Most previous work models user and product identities as representation vectors which are implicitly learned during the training process and only focus on the interactions between either the user or product and the review text (Dou, 2017; Long et al., 2018; Amplayo, 2019; Zhang et al., 2021; Amplayo et al., 2022). This brings with it two major shortcomings: i) the associations between users and products are not fully exploited, and, ii) the text of historical reviews is not used.

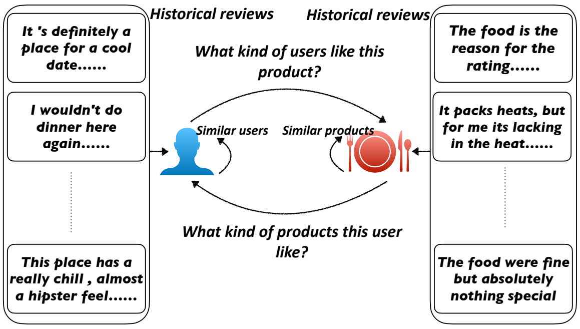
Historical reviews
Historical reviews
It 's definitely a place for a cool date......
I wouldn't do dinner here again......
This place has a really chill , almost a hipster feel......
What kind of users like this product?
Similar users
Similar products
What kind of products this user like?
The food is the reason for the rating......
It packs heats, but for me its lacking in the heat......
The food were fine but absolutely nothing special

Figure 1: Our proposed idea of representing users and products with their historical reviews and incorporating the associations between users and products.

To tackle the first shortcoming, Amplayo et al. (2018) propose to incorporate similar user and product representations for review sentiment classification. However, their approach ignores the associations *between users and products*. To tackle the second shortcoming, Lyu et al. (2020) propose to explicitly use historical reviews in the training process. However, their approach needs to incrementally store review representations during the training process, which results in a more complex model architecture, where the magnitude of the user and product matrix is difficult to control when the number of reviews grow very large.

As shown in Figure 1, we propose two simple strategies to address the aforementioned issues. Firstly, we use pre-trained language models (PLMs) to pre-compute the representations of all historical reviews belonging to the same user or product. Historical review representations are then used to initialize user (or product) representations by average pooling over all tokens before again average pooling over all reviews. This allows historical review text to inform the user and product preference, which we believe is potentially more advantageous than implicitly learned representations. Time and memory costs are minimized compared to (Lyu et al., 2020) since the representations of historical reviews are average pooled and the precomputation is one-time.

Secondly, we propose a user-product cross-

Findings of the Association for Computational Linguistics: ACL 2023, pages 1419–1429
July 9-14, 2023 ©2023 Association for Computational Linguistics

context module which interacts on four dimensions: user-to-user, product-to-product, user-to-product and product-to-user. The former two are used to obtain similar user (or product) information, which is useful when a user (or product) has limited reviews. The latter two are used to model the product preference of the user (what kind of products do they like and what kind of ratings would they give to similar products?) and user preference associated with a product (what kinds of users like such products and what kinds of ratings would they give to this product?).

We test our approach on three benchmark English datasets – IMDb, Yelp-2013, Yelp-2014. Our approach yields consistent improvements across several PLMs (BERT, SpanBERT, Longformer) and achieves substantial improvements over the previous state-of-the-art.

2 Methodology

An overview of our approach is shown in Figure 2. We firstly feed the review text, D, into a PLM encoder to obtain its representation, H_D. H_D is then fed into a *user-product cross-context* module consisting of multiple attention functions together with the corresponding user embedding and product embedding. The output is used to obtain the distribution over all sentiment labels. The architecture design is novel in two ways: 1) the user and product embedding matrices are initialized using representations of historical reviews of the corresponding users/products, 2) a user-product cross-context module works in conjunction with 1) to model textual associations between users and products.

2.1 Incorporating Textual Information of Historical Reviews

For the purpose of making use of the textual information of historical reviews, we initialize all user and product embedding vectors using the representations of their historical reviews. Specifically, assume that we have a set of users $U = \{u_1, ..., u_N\}$ and products $P = \{p_1, ..., p_M\}$. Each user u_i and product p_j have their corresponding historical reviews: $u_i = \{D_1^{u_i}, ..., D_{n_i}^{u_i}\}$ and $p_j = \{D_1^{p_j}, ..., D_{m_j}^{p_j}\}$.

For a certain user u_i, we firstly feed $D_1^{u_i}$ into the transformer encoder to obtain its representation $H_{D_1}^{u_i} \in \mathbf{R}^{L\times h}$, then we average $H_{D_1}^{u_i}$ along its first dimension:

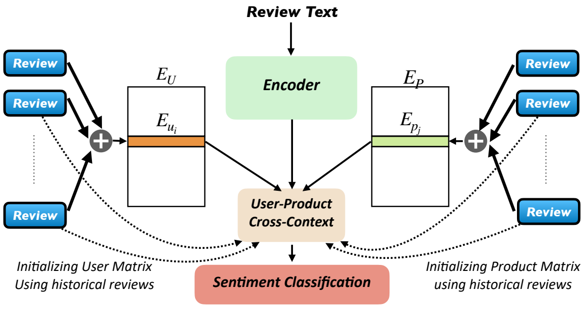
Review Text
Review
Review
Review
E_U
E_{u_i}
Encoder
E_P
E_{p_j}
Review
Review
Review
User-Product Cross-Context
Initializing User Matrix Using historical reviews
Sentiment Classification
Initializing Product Matrix using historical reviews

Figure 2: Our model architecture. We initialize user representation matrix E_U and product representation matrix E_P. The user vector E_{u_i} and product vector E_{p_j} are fed into user-product cross-context module with document representation H_D. The dashed lines indicate the direct interactions of historical reviews in the cross-context module.

$$\bar{H}_{D_1}^{u_i} = \frac{\sum H_{D_1}^{u_i}}{T_{D_1}^{u_i}} \tag{1}$$

where $\bar{H}_{D_1}^{u_i} \in \mathbf{R}^{1\times h}$, L is the maximum sequence length, h is the hidden size of the transformer encoder, $T_{D_1}^{u_i}$ is the total number of tokens in $D_1^{u_i}$ excluding special tokens. Therefore, we sum the representations of all tokens in $D_1^{u_i}$ and then average it to obtain a document vector $\bar{H}_{D_1}^{u_i}$. The same procedure is used to generate the document vectors of all documents in $u_i = \{D_1^{u_i}, ..., D_{n_i}^{u_i}\}$. Finally, we obtain the representation of u_i by:

$$E_{u_i} = \frac{\sum_{k=1}^{n_i} \bar{H}_{D_k}^{u_i}}{n_i} \tag{2}$$

where $E_{u_i} \in \mathbf{R}^{1\times h}$ is the initial representation of user u_i. The same process is applied to generate the representations of all the other users as well as all products. Finally, we have $E_U \in \mathbf{R}^{N\times h}$ and $E_P \in \mathbf{R}^{M\times h}$ as the user and product embedding matrix respectively. Moreover, in order to control the magnitude of E_U, E_P we propose scaling heuristics:

$$\hat{E_U} = f_U E_U, f_U = \frac{\text{F-NORM}(E)}{\text{F-NORM}(E_U)} \tag{3}$$

where F-NORM is Frobenius norm, E is a normal matrix in which the elements $E_{i,j}$ are drawn from a normal distribution $\mathcal{N}(0, 1)$. The same process is applied to E_P as well.

2.2 User-Product Information Integration

Having enriched user and product representations with historical reviews, we propose a user-product cross-context module for the purpose of garnering sentiment clues from textual associations between users and products. We use MULTI-HEAD ATTENTION (Vaswani et al., 2017) in four attention operations: *user-to-user, product-to-product, user-to-product* and *product-to-user*. Specifically, for MULTI-HEAD ATTENTION(Q,K,V), we use the user representation E_{u_i} or product representation E_{p_j} as *Q* and the user matrix E_U and product matrix E_P as *K* and *V*. For example, we obtain *user-to-user attention* output by:

$$E_{u_i}^{uu} = Attn_{uu}(E_{u_i}, E_U, E_U) \tag{4}$$

We follow the same schema to get $E_{p_j}^{pp}$, $E_{u_i}^{up}$ and $E_{p_j}^{pu}$. Additionally, we also employ two MULTI-HEAD ATTENTION operations between E_{u_i}/E_{p_j} (query) and H_D (key and value). The corresponding outputs are $E_{u_i}^D$ and $E_{p_j}^D$. We then combine the output of the user-product cross-context module and H_{cls} to form the final representations. In $Attn_{uu}$ and $Attn_{pp}$, we add attention masks to prevent E_{u_i} and E_{p_j} from attending to themselves. Thus we also incorporate E_{u_i} and E_{p_j} as their *self-attentive* representations:

$$\begin{aligned} H_d =& g(E_{u_i}^{uu}, E_{p_j}^{pp}, E_{u_i}^{up}, E_{p_j}^{pu}, E_{u_i}^D, E_{p_j}^D, \\ & E_{u_i}, E_{p_j}, H_{cls}) \end{aligned} \tag{5}$$

H_d is fed into the classification layer to obtain the sentiment label distribution. During the training process, we use cross-entropy to calculate the loss between our model predictions and the gold labels.

3 Experiments

3.1 Datasets

Our experiments are conducted on three benchmark English document-level sentiment analysis datasets: IMDb, Yelp-13 and Yelp-14 (Tang et al., 2015). Statistics of the three datasets are shown in Appendix A.1. All three are fine-grained sentiment analysis datasets: Yelp-2013 and Yelp-2014 have 5 classes, IMDb has 10 classes. Each review is accompanied by its corresponding anonymized user ID and product ID.

3.2 Experimental Setup

The pre-trained language models we employed in experiments are BERT (Devlin et al., 2019), SpanBERT (Joshi et al., 2020) and Longformer (Beltagy et al., 2020). We use the implementations from Huggingface (Wolf et al., 2019). The hyperparameters are empirically selected based on the performance on the dev set. We adopt an early stopping strategy. The maximum sequence is set to 512 for all models. For evaluation, we employ two metrics *Accuracy* and *RMSE* (Root Mean Square Error). More training details are available in Appendix A.2

3.3 Results

Results on the dev sets of IMDb, Yelp-2013 and Yelp-2014 for the BERT, SpanBERT and Longformer PLMs are shown in Table 1. We compare our approach to a vanilla user and product attention baseline where 1) the user and product representation matrices are randomly initialized and 2) we simply employ multi-head attention between user/product and document representations without the user-product cross-context module. Our approach is able to achieve consistent improvements over the baseline with all PLMs on all three datasets. For example, our approach gives improvements over the baseline of 4.3 accuracy on IMDb, 1.6 accuracy on Yelp-2013 and 1.7 accuracy on Yelp-2014 for BERT-base. Moreover, our approach can give further improvements for large PLMs such as Longformer-large: improvements of 4.8 accuracy on IMDb, 2.8 accuracy on Yelp-2013 and 2.1 accuracy on Yelp-2014. The improvements over the baseline are statistically significant ($p < 0.01$)[1].

In Table 2, we compare our approach to previous approaches on the test sets of IMDb, Yelp-2013 and Yelp-2014. These include pre-BERT neural models – RRP-UPM (Yuan et al., 2019) and CHIM (Amplayo, 2019) – and state-of-the-art models based on BERT – IUPC (Lyu et al., 2020), MA-BERT (Zhang et al., 2021) and Injectors (Amplayo et al., 2022).[2] We use BERT-base for a fair comparison with IUPC, MA-BERT and Injectors, which all use BERT-base. Our model obtains the best performance on IMDb, Yelp-2013 and Yelp-2014, achieving absolute improvements in accuracy of 0.1, 1.2 and 0.9 respectively, and improvements in RMSE of 0.011, 0.018 and 0.010 respectively.

3.4 Ablation Study

Results of an ablation analysis are shown in Table 3. The first row results are from a BERT model without user and product information. The next

[1] We use a paired t-test to determine the significance of our method's improvements over the baseline models.

[2] More results are shown in Appendix A.4.

	IMDB		Yelp-2013		Yelp-2014	
	Acc. (%)	RMSE	Acc. (%)	RMSE	Acc. (%)	RMSE
Vanilla BERT-base Attention	55.4	1.129	69.1	0.617	70.7	0.610
+ Our approach	**59.7**	**1.006**	**70.7**	**0.589**	**72.4**	**0.559**
Vanilla BERT-large Attention	55.7	1.070	69.9	0.590	71.3	0.579
+ Our approach	**60.3**	**0.977**	**71.8**	**0.568**	**72.3**	**0.567**
Vanilla SpanBERT-base Attention	56.6	1.055	70.2	0.589	71.3	0.571
+ Our approach	**60.2**	**1.026**	**71.5**	**0.578**	**72.6**	**0.562**
Vanilla SpanBERT-large Attention	57.6	1.009	71.6	0.563	72.5	0.556
+ Our approach	**61.0**	**0.947**	**72.7**	**0.552**	**73.7**	**0.543**
Vanilla Longformer-base Attention	56.7	1.019	71.0	0.573	72.5	0.554
+ Our approach	**59.6**	**0.990**	**72.6**	**0.558**	**73.3**	**0.548**
Vanilla Longformer-large Attention	57.0	0.967	70.7	0.571	72.2	0.555
+ Our approach	**61.8**	**0.931**	**73.5**	**0.540**	**74.3**	**0.529**

Table 1: Results of our approach on various PLMs on the dev sets of IMDb, Yelp-2013 and Yelp-2014. We show the results of the baseline vanilla attention model for each PLM as well as the results of the same PLM with our proposed approach. We report the average of five runs with two metrics, Accuracy (↑) and RMSE (↓).

	IMDB		Yelp-2013		Yelp-2014	
	Acc. (%)	RMSE	Acc. (%)	RMSE	Acc. (%)	RMSE
RRP-UPM (Yuan et al., 2019)	56.2	1.174	69.0	0.629	69.1	0.621
CHIM (Amplayo, 2019)	56.4	1.161	67.8	0.641	69.2	0.622
IUPC (Lyu et al., 2020)	53.8	1.151	70.5	0.589	71.2	0.592
MA-BERT (Zhang et al., 2021)	57.3	1.042	70.3	0.588	71.4	0.573
Injectors (Amplayo et al., 2022)	58.9	N/A	70.9	N/A	71.7	N/A
Ours	**59.0**	**1.031**	**72.1**	**0.570**	**72.6**	**0.563**

Table 2: Experimental Results on the test sets of IMDb, Yelp-2013 and Yelp-2014. We report the average results of of five runs of two metrics Accuracy (↑) and RMSE (↓). The best performance is in bold.

	IMDB		Yelp-2013		Yelp-2014	
	Acc. (%)	RMSE	Acc. (%)	RMSE	Acc. (%)	RMSE
BERT	50.8	1.187	67.2	0.639	67.8	0.629
+ User-Product Information	55.4	1.129	69.1	0.617	70.7	0.610
+ Textual Information	56.9	1.089	70.1	0.593	71.9	0.563
+ User-Product Cross-Context	59.7	1.006	70.7	0.589	72.4	0.559

Table 3: Results of ablation studies on the dev sets of IMDb, Yelp-2013 and Yelp-2014.

three rows correspond to: 1) *User-Product Information*, where we use the same method in the baseline vanilla attention model in Table 1 to inject user-product information; 2) *Textual Information*, our proposed approach of using historical reviews to initialize user and product representations; 3) *User-Product Cross-Context*, our proposed module incorporating the associations between users and products. The results show, firstly, that user and product information is highly useful for sentiment classification, and, secondly, that both textual information of historical reviews and user-product cross-context can improve sentiment classification. *Textual Information* gives ~1 accuracy improvement on the three datasets, while giving ~0.04 RMSE improvement on IMDb and Yelp-2014 and ~0.02 RMSE improvement on Yelp-2013. *User-Product Cross-Context* achieves further improvements on IMDb of 2.8 accuracy and improvements on Yelp-2013 and Yelp-2014 of 0.6 and 0.5 accuracy respectively.

3.5 Varying Number of Reviews

We investigate model performance with different amounts of reviews belonging to the same user/product. We randomly sample a proportion of each user's reviews (from 10% to 100%). Then we use the sampled training data, where each user only has part of their total reviews (e.g. 10%), to train sentiment classification models. We conduct experiments on Yelp-2013 and IMDb using IUPC (Lyu et al., 2020), MA-BERT (Zhang et al., 2021) and our approach. The results are shown in Figure 3,

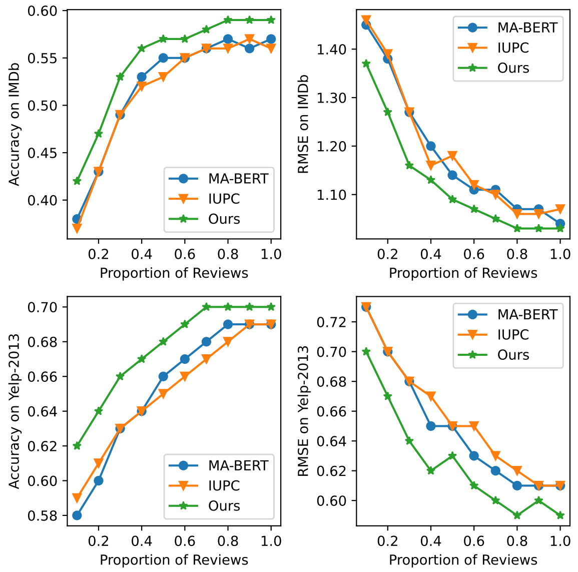

Figure 3: Experimental results of IUPC, MA-BERT and our approach under different proportions of reviews from 10% to 100% on the dev sets of IMDb (top) and Yelp-2013 (bottom).

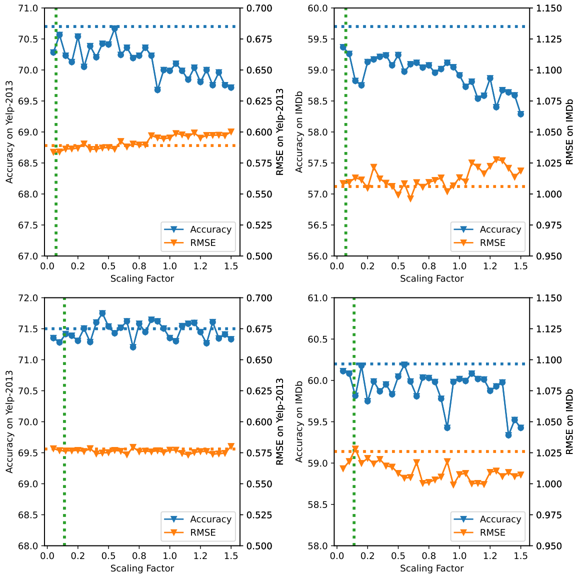

Figure 4: Effect of varying the scaling factor for the user/product matrices on the dev sets of Yelp-2013 (left) and IMDb (right), with BERT-base (top) and SpanBERT-base (bottom). The left and right y-axis in each subplot represent *Accuracy* and *RMSE* respectively. The x-axis represents the scaling factor. The vertical green dashed line is the scaling factor from the Frobenius norm heuristic. The blue and orange horizontal dashed lines are the accuracy and RMSE produced by the Frobenius norm heuristic respectively.

where the x-axis represents the proportion of reviews that we used in experiments. When the proportion of reviews lie between 10% and 50%, our approach obtains superior performance compared to MA-BERT and IUPC while the performance gain decreases when users have more reviews. The results show the advantage of our approach under a low-review scenario for users.

3.6 Scaling Factor for User/Product Matrix

We conduct experiments with different scaling factor (see Equations 3) on the dev sets of Yelp-2013 and IMDb using BERT-base. We apply the same scaling factor to both user and product matrix. The results are shown in Figure 4, where we use scaling factor ranging from 0.05 to 1.5 with intervals of 0.05. The results show that our proposed scaling factor (green dashed lines in Figure 4) based on the Frobenius norm can yield competitive performance: best accuracy according to the blue dashed line. Although the RMSE of the Frobenius norm heuristic is not always the optimal, it is still a relatively lower RMSE compared to most of the other scaling factors (except the RMSE of SpanBERT-base on IMDb). Moreover, the Frobenius norm heuristic can reduce the efforts needed to tune the scaling factor, since the optimal scaling factor is varying for different models on different data, whereas the Frobenius norm heuristic is able to consistently provide a competitive dynamic scaling factor.

4 Conclusion and Future Work

In order to make the best use of user and product information in sentiment classification, we propose a text-driven approach: 1) explicitly utilizing historical reviews to initialize user and product representations 2) modeling associations between users and products with an additional user-product cross-context module. The experiments conducted on three English benchmark datasets – IMDb, Yelp-2013 and Yelp-2014 – demonstrate that our approach substantially outperforms previous state-of-the-art approaches and is effective for several PLMs. For future work, we aim to apply our approach to more tasks where there is a need to learn representations for various types of attributes, and to explore other compositionality methods for generating user/product representations.

Acknowledgements

This work was funded by Science Foundation Ireland through the SFI Centre for Research Training in Machine Learning (18/CRT/6183). We thank the reviewers for helpful feedback.

Limitations

The method introduced in this paper applies to a specific type of sentiment analysis task, where the item to be analysed is a review, the author of the review and the product/service being reviewed are known and uniquely identified, and the author (user) and product information is available for all reviews in the training set.

While our approach is expected to perform well on other languages beyond English, the experimental results do not necessarily support that since our evaluation is only carried out on English data.

References

Reinald Kim Amplayo. 2019. Rethinking attribute representation and injection for 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)*, pages 5602–5613, Hong Kong, China. Association for Computational Linguistics.

Reinald Kim Amplayo, Jihyeok Kim, Sua Sung, and Seung-won Hwang. 2018. Cold-start aware user and product attention for sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2535–2544, Melbourne, Australia. Association for Computational Linguistics.

Reinald Kim Amplayo, Kang Min Yoo, and Sang-Woo Lee. 2022. Attribute injection for pretrained language models: A new benchmark and an efficient method. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1051–1064, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016. Neural sentiment classification with user and product attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1650–1659, Austin, Texas. Association for Computational Linguistics.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Cícero dos Santos and Maíra Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Zi-Yi Dou. 2017. Capturing user and product information for document level sentiment analysis with deep memory network. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 521–526, Copenhagen, Denmark. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Yunfei Long, Mingyu Ma, Qin Lu, Rong Xiang, and Chu-Ren Huang. 2018. Dual memory network model for biased product review classification. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 140–148, Brussels, Belgium. Association for Computational Linguistics.

Chenyang Lyu, Jennifer Foster, and Yvette Graham. 2020. Improving document-level sentiment analysis with user and product context. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6724–6729, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Dehong Ma, Sujian Li, Xiaodong Zhang, Houfeng Wang, and Xu Sun. 2017. Cascading multiway attentions for document-level sentiment classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 634–643, Taipei, Taiwan. Asian Federation of Natural Language Processing.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Learning semantic representations of users and products for document level sentiment classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*

(Volume 1: Long Papers), pages 1014–1023, Beijing, China. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Zhen Wu, Xin-Yu Dai, Cunyan Yin, Shujian Huang, and Jiajun Chen. 2018. Improving review representations with user attention and product attention for sentiment classification. *CoRR*, abs/1801.07861.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

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 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7,2019*, pages 2341–2344.

You Zhang, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2021. MA-BERT: Learning representation by incorporating multi-attribute knowledge in transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2338–2343, Online. Association for Computational Linguistics.

A Appendix

A.1 Datasets

Datasets	Train	Dev	Test	Words/Doc
IMDB	67,426	8,381	9,112	394.6
Yelp-2013	62,522	7,773	8,671	189.3
Yelp-2014	183,019	22,745	25,399	196.9

Table 4: Number of documents per split and average doc length of IMDb, Yelp-2013 and Yelp-2014.

Our experiments are conducted on three benchmark English document-level sentiment analysis datasets: IMDb, Yelp-13 and Yelp-14 (Tang et al., 2015). Statistics of the three datasets are shown in Table 4. The IMDb dataset has the longest documents with an average length of approximately 395 words. The average number of reviews for each user/product is shown in Table 5.

Datasets	Users	Products	Docs/User	Docs/Product
IMDB	1,310	1,635	64.82	51.94
Yelp-2013	1,631	1,633	48.42	48.36
Yelp-2014	4,818	4,194	47.97	55.11

Table 5: Number of users and products with average amount of documents for each user and product in IMDb, Yelp-2013 and Yelp-2014.

A.2 Hyperparameters

The metrics are calculated using the scripts in Pedregosa et al. (2011). All experiments are conducted on Nvidia GeForce RTX 3090 GPUs. We show the Learning Rate and Batch Size used to train our models on all datasets in Table 6.

A.3 Training Objective

We use *Cross-Entropy* to calculate the loss between our model predictions and the gold labels.

$$J(\theta) = -\sum_{i=1}^{n}\sum_{j=1}^{m} y_{i,j} log(p(y_{i,j}|D_i, u_i, p_i)) \quad (6)$$

where n is the number of samples and m is the number of all classes, $y_{i,j}$ represents the actual probability of the i-th sample belonging to $class_j$, $y_{i,j}$ is 1 only if the i-th sample belongs to $class_j$ otherwise it's 0. $p(y_{i,j}|D_i, u_i, p_i)$ is the probability the i-th sample belongs to $class_j$ predicted by our model.

A.4 Evaluation Results

We compare our approach to previous approaches on the test sets of IMDb, Yelp-2013 and Yelp-2014. These include pre-BERT neural baseline models using CNN (dos Santos and Gatti, 2014; Kim, 2014) and LSTM (Yang et al., 2016) – UPNN (Tang et al., 2015), NSC (Chen et al., 2016), UPDMN (Dou, 2017), CMA (Ma et al., 2017), HCSC (Amplayo et al., 2018), DUPMN (Long et al., 2018), HUAPA (Wu et al., 2018), RRP-UPM (Yuan et al., 2019), CHIM (Amplayo, 2019) – and two state-of-the-art models based on BERT including IUPC (Lyu et al., 2020) and MA-BERT (Zhang et al., 2021). Results are shown in Table 7.

A.5 Examples

Some cases sampled from the dev set of Yelp-2013 and corresponding predictions from Vanilla BERT w/o user and product information, IUPC (Lyu et al., 2020), MA-BERT (Zhang et al., 2021) and our model are shown in Table 8.

	IMDB		Yelp-2013		Yelp-2014	
	BS	LR	BS	LR	BS	LR
BERT-base	16	6e-5	16	6e-5	16	6e-5
BERT-large	8	3e-5	8	3e-5	8	3e-5
SpanBERT-base	16	6e-5	16	6e-5	16	6e-5
SpanBERT-large	8	3e-5	8	3e-5	8	3e-5
Longformer-base	16	3e-5	16	3e-5	16	3e-5
Longformer-large	4	2e-5	4	3e-5	4	3e-5

Table 6: The hyperparameters used to fine-tune all models on all datasets including Learning Rate (LR) and Batch Size (BS).

	IMDB		Yelp-2013		Yelp-2014	
	Acc. (%)	RMSE	Acc. (%)	RMSE	Acc. (%)	RMSE
Pre-BERT models						
UPNN (Tang et al., 2015)	43.5	1.602	59.6	0.784	60.8	0.764
NSC (Chen et al., 2016)	53.3	1.281	65.0	0.692	66.7	0.654
UPDMN (Dou, 2017)	46.5	1.351	63.9	0.662	61.3	0.720
CMA (Ma et al., 2017)	54.0	1.191	66.3	0.677	67.6	0.637
HCSC (Amplayo et al., 2018)	54.2	1.213	65.7	0.660	67.6	0.639
DUPMN (Long et al., 2018)	53.9	1.279	66.2	0.667	67.6	0.639
HUAPA (Wu et al., 2018)	55.0	1.185	68.3	0.628	68.6	0.626
RRP-UPM (Yuan et al., 2019)	56.2	1.174	69.0	0.629	69.1	0.621
CHIM (Amplayo, 2019)	56.4	1.161	67.8	0.641	69.2	0.622
BERT-based models						
IUPC (Lyu et al., 2020)	53.8	1.151	70.5	0.589	71.2	0.592
MA-BERT (Zhang et al., 2021)	57.3	1.042	70.3	0.588	71.4	0.573
Injectors (Amplayo et al., 2022)	58.9	N/A	70.9	N/A	71.7	N/A
Ours	**59.0**	**1.031**	**72.1**	**0.570**	**72.6**	**0.563**

Table 7: Experimental Results on the test sets of IMDb, Yelp-2013 and Yelp-2014. We report the average results of of five runs of two metrics Accuracy (↑) and RMSE (↓). The best performance is in bold.

Example 1 This is a straightforward positive review since it clearly conveys the satisfaction towards the restaurant. Thus all models make the correct prediction.

Example 2 This is similar to the first example in narrative style, but the ground-truth sentiment label is Positive rather than Very Positive since this user tends not to give very high ratings. This example shows the importance of user information.

Example 3 This review conveys a very negative attitude. However, the author tends not to give very poor ratings plus the reviews this store received are not bad. With both user and product information, our model makes the correct prediction of Neutral.

Example 4 All models, regardless of whether they use user and product information, predict Neutral or Negative while in fact the review label is Very Positive. This is a difficult example where the sentiment is subtly expressed.

Review	Vanilla BERT	IUPC	MA-BERT	Ours
Took travis here for one of our first dates and just love cibo. It ’s situated in a home from 1913 and has colored lights wrapped all around the trees. You can either sit inside or on the gorgeous patio. Brick oven pizza and cheese plates offered here and it ’s definitely a place for a cool date. (VP)	VP (✓)	VP (✓)	VP (✓)	VP (✓)
a great sushi bar owned and operated by maggie and toshi who are both japanese. their product is always consistent and they always have a few good specials. service is great and the staff is very friendly and cheerful. value is really good particularly within their happy hour menu. our kids love it and they are always spoiled rotten by maggie and toshi so it is their favorite place. lastly we did a sake tasting there a few weeks ago and really had a great time. we all sat family style int he middle of the restaurant and got to experience some really interesting rice wines. we had a blast. great place (P)	VP (✗)	P (✓)	P (✓)	P (✓)
well , i was disappointed. i was expecting this one to be a jazzed up container store. but ... it was just average. i used to visit container store in houston near the galleria. it has a nice selection of things. people are always ready to help etc.. but , this one has an aloof sort of customer service crowd. they say nice things about your kid but do not offer to help. hmm ... i have seen similar things they were selling at ikea. the quality did seem a little better than ikea but if you are buying a laundry room shelf for your laundry detergent ... who the hell cares. its a shelf ! does n’t matter if it has 15 coats of paint on the metal or 2 coats. i found one of those sistema lunch boxes that i have been looking for over here and it was on sale. will i go back ? probably not. too far out for me , plus i like ikea better (Ne)	VN (✗)	N (✗)	VN (✗)	Ne (✓)
Unfortunately tonight was the last night this location was open. The only two locations left in the valley are desert ridge and arrowhead. Please support them. (VP)	Ne (✗)	N (✗)	VN (✗)	N (✗)

Table 8: Example reviews from the dev sets of Yelp-2013 and the corresponding predictions of each model. Very Negative (VN), Negative (N), Neutral (Ne), Positive (P), Very Positive (VP).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
The last section.

☑ A2. Did you discuss any potential risks of your work?
Section 3.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 3.

☑ B1. Did you cite the creators of artifacts you used?
Section 3.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 3.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 3.

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 3.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 3 and appendix.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 3 and appendix.

C ☑ Did you run computational experiments?

Section 3 and appendix.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 3 and appendix.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 3 and appendix.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 3 and appendix.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 3 and appendix.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.