On the Role of Reviewer Expertise in Temporal Review Helpfulness Prediction

Mir Tafseer Nayeem
University of Alberta
mnayeem@ualberta.ca

Davood Rafiei
University of Alberta
drafiei@ualberta.ca

Abstract

Helpful reviews have been essential for the success of e-commerce services, as they help customers make quick purchase decisions and benefit the merchants in their sales. While many reviews are informative, others provide little value and may contain spam, excessive appraisal, or unexpected biases. With the large volume of reviews and their uneven quality, the problem of detecting helpful reviews has drawn much attention lately. Existing methods for identifying helpful reviews primarily focus on review text and ignore the two key factors of (1) **who** post the reviews and (2) **when** the reviews are posted. Moreover, the helpfulness votes suffer from scarcity for less popular products and recently submitted (a.k.a., cold-start) reviews. To address these challenges, we introduce a dataset and develop a model that integrates the reviewer's expertise, derived from the past review history of the reviewers, and the temporal dynamics of the reviews to automatically assess review helpfulness. We conduct experiments on our dataset to demonstrate the effectiveness of incorporating these factors and report improved results compared to several well-established baselines.

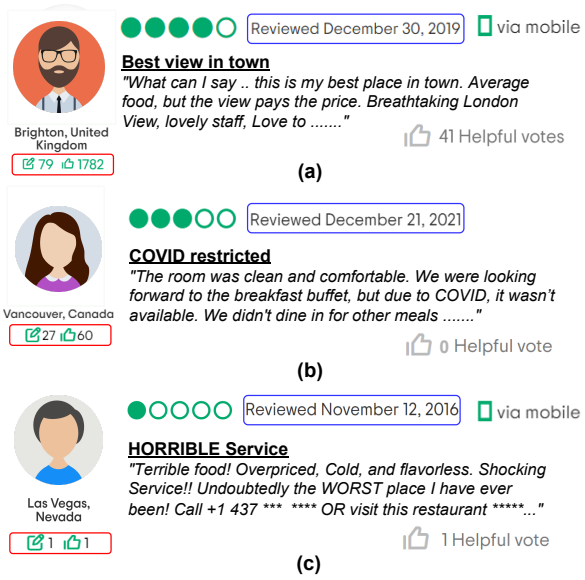

Figure 1: A snapshot of three reviews with the reviewers' history information: Review *a* has accumulated more helpful votes but is posted almost two years before Review *b*; on the other hand, Review *b* (a.k.a., cold-start review) contains time-sensitive information, describing the current conditions and Review *c* is likely a spam review. Photos of the reviewers are replaced with avatars for privacy reasons.

1 Introduction

Many customers rely on online reviews from non-professionals, on daily basis, to decide what products to buy (e.g., *Amazon*), what hotels to stay at (e.g., *TripAdvisor*), what restaurants to eat (e.g., *Yelp*) and even what books to read (e.g., *Goodreads*). A recent survey of Bizrate Insights reward members found that approximately *98%* of online shoppers research a vendor via online reviews before making a purchase decision (Kats, 2018). Since the reviews are expected to describe the actual experiences and opinions of users, they can provide a reliable source of reference, improving other customers' confidence, comfort, and the overall shopping experience (Foo et al., 2017; Gamzu et al., 2021). However, despite their tremendous benefits, online reviews are often of mixed qualities. While many reviews are informative, others provide little value and may contain excessive appraisal or spam (see Figure 1-c). There are multiple factors that affect the quality of a review, including the reviewers' life experience, educational background, and the motive for writing the review (Du et al., 2021), and these factors are not usually explicit in the review text. All these pose challenges for customers who are less experienced in a subject area and need the reviews the most, simply because there is less incentive for more experienced users to use the reviews. Moreover, customers usually have limited patience for reading reviews – most customers read less than 10 reviews before mak-

Findings of the Association for Computational Linguistics: EACL 2023, pages 1684–1692

ing a purchase decision about an item (Murphy, 2016). The large volume of reviews and their unpredictable quality and the limited customer patience demand better review utilization strategies to manage the information overload.

One standard method to identify more informative reviews is to ask for feedback from customers or site visitors who read them. By asking, *"Was this review helpful to you?,"* or *"Did you find this review helpful?"* at the end of each review, online platforms can crowdsource helpfulness votes from other customers. As a result, user reviews that gain the most helpful votes are shown first to the potential buyers to make the decision easier. However, the voting data suffers from scarcity (Siersdorfer et al., 2010) since only a tiny proportion of customers are willing to cast helpfulness votes. The scarcity is even more severe in reviews of less popular products and more recently submitted reviews (a.k.a., cold-start reviews) (Liu et al., 2008), despite the fact that more recent reviews may in fact contain more relevant and time-sensitive information (e.g., *"New COVID Restrictions"* or *"Dirty Pool Area"*) as shown in Figure 1-b but no helpfulness vote.

In this paper, we study the confluence of the reviewing history of reviewers and the review text for helpfulness identification. First, we observe that people who post more reviews and earn more helpful votes are more likely to be better reviewers. Second, trustworthy reviewers (e.g., Figure 1-a) are less likely to be posting fake or biased reviews, and their reviews are more likely to earn more helpful votes; otherwise, they will be ruining their reputation. Third, those who have been to more hotels or restaurants across different cities have a better basis for comparison and writing critical reviews. To the best of our knowledge, existing works only focus on review content and neglect the reviewers and their reviewing history. Integrating the review text with the reviewing history of the reviewers is the problem studied in this paper.

Our main contributions are summarized as follows:

- We introduce a new dataset with both review text and reviewer's history, to highlight the importance of integrating the two sources for review helpfulness.
- We propose a model incorporating the reviewer's expertise and temporal information of reviews in helpfulness prediction.
- We present a detailed case-study to interpret the model behavior and highlight potential directions to be addressed in the future.

2 Related work

More traditional approaches on review helpfulness prediction focus solely on the text of reviews, and some consider both text and images to guide the prediction. In general, the task can be addressed using a predictive model based on hand-crafted features such as `structural` (Susan and David, 2010; Xiong and Litman, 2014), `lexical` (Kim et al., 2006; Xiong and Litman, 2011), `syntactic` (Kim et al., 2006), `emotional` (Martin and Pu, 2014), `semantic` (Yang et al., 2015), and `arguments` (Liu et al., 2017) from the review text. These features may be fed into a conventional classifier such as SVM, Random Forest, or gradient boosting to identify helpful reviews. These methods heavily rely on manual feature engineering, which is labor-intensive and time-consuming.

Inspired by the remarkable progress of deep neural networks, more recent studies make use of deep neural models, which can learn both intrinsic and extrinsic features given labeled data. Chen et al. (2018) uses a text-based CNN model to automatically capture the character-level, word-level, and topic-level features for helpfulness prediction. Fan et al. (2018) uses an end-to-end multi-task neural architecture with the help of an auxiliary task, such as rating regression, to boost the performance of the review helpfulness identification. Liu et al. (2021) and Han et al. (2022) use both text and images to guide the review helpfulness prediction. Since the image field is usually optional in reviews, a large volume of reviews contain only text, for which these multimodal models would produce inconsistent results.

3 Review Helpfulness Prediction

3.1 Dataset

To the best of our knowledge, there is no human-annotated dataset that is publicly available for the task of review helpfulness prediction with the reviewers' attributes and review date. Therefore, we build our dataset by scraping reviews from `TripAdvisor`[1]. Out of 225,664 reviews retrieved, close to one third have no helpful votes. We filter such reviews, and this reduces the number of reviews to 161,541. Table 1 presents the summary of

[1] `https://www.tripadvisor.com`

	Train	Valid	Test
Total #Samples	145,381	8,080	8,080
Avg. #Sentences	7.82	7.80	7.81
Avg. #Words	152.37	152.25	148.90

Table 1: Our dataset statistics.

our dataset with train, validation, and test splits[2]. Following (Liu et al., 2021), we leverage a logarithmic scale ($\lfloor \log_2 n_{\text{votes}} \rfloor$) to categorize the reviews based on the number of votes received. Specifically, we map the number of votes into five intervals (i.e., [1,2), [2, 4), [4, 8), [8, 16), [16, ∞)), each corresponding to a helpfulness score $Y \in \{1, 2, 3, 4, 5\}$, where the higher the score, the more helpful the review.

3.2 Proposed Model

Review **H**elpfulness **P**rediction (**RHP**) can be modeled as a supervised machine learning task where the input contains information about the reviews ($\mathcal{R}$) and the reviewers ($\mathcal{U}$). Let $\mathcal{R}_i = ([s_1, \ldots, s_N],\ t_i)$ denote a review posted at time t_i with sentences $s_1, \ldots, s_N$, and $\mathcal{U}_i = (n_i,\ m_i)$ denote a reviewer who posts n_i reviews and earns a total of m_i helpful votes. We formulate the review helpfulness prediction as a multi-class classification where we seek to find a model f that minimizes the loss function $\mathcal{L}$, i.e.

$$min_\theta \mathcal{L}\ (f(\theta, \mathcal{R}, \mathcal{U}),\ Y), \tag{1}$$

where Y is the ground-truth, θ is the model parameter and the output of the model is a helpfulness class $\hat{Y} \in \{1, 2, 3, 4, 5\}$. The learning task is to find the best parameter that minimizes the above equation.

We encode the review sentences using BERT (Devlin et al., 2019; Xu et al., 2019). We concatenate the review sentences together while inserting a [CLS] token at the start and a [SEP] token at the end. If $\mathbf{h}^{\texttt{[CLS]}}$ denotes the embedding vector of the special [CLS] token and $\mathbf{h}^{(i)}$ denotes the embedding vector of the i-th token, we extract the last hidden state of $\mathbf{h}_l^{\texttt{[CLS]}}$ to represent the review sentences and apply a linear transformation to get a final contextualized representation $x_h \in \mathbb{R}^K$, where Θ is a non-linear activation function.

$$[\mathbf{h}^{\texttt{[CLS]}}, \mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \ldots] = \textbf{BERT}(\texttt{[CLS]}\ s_1, \ldots, s_N\ \texttt{[SEP]}), \tag{2}$$

$$x_h = \Theta\ (\textbf{MLP}\ (\mathbf{h}_l^{\texttt{[CLS]}})). \tag{3}$$

Generally, users who post more reviews and earn more helpful votes are likely to be better reviewers. Such users may have been to more hotels and restaurants across the globe and have a better basis for comparison. We define the term *reviewer expertise* as the mean number of helpful votes received per review, written as $e_s = m/n$ for a reviewer who posts m reviews and earns n overall helpfulness votes. We use a linear layer to get a weighted representation of the expertise score (h_s).

$$h_s = \textbf{MLP}\ (e_s) \tag{4}$$

Previous approaches for this task fail to consider the temporal nature of the reviews. Older reviews are more likely to accumulate more helpfulness votes than newer reviews but are not necessarily the most relevant describing the current conditions (e.g., *new COVID restrictions*). One-time problems such as broken bathrooms and dirty pool area are likely to be addressed and to be less relevant. Let t_d be the relative age of a review in days, for example, as of the day the reviews are scraped. We use a linear layer to get a weighted representation of the relative review age.

$$h_t = \textbf{MLP}\ (t_d). \tag{5}$$

It should be noted that both the review age and the reviewer expertise are normalized to a fixed range $[a, b]$ before being used in the linear layers in Equations 4 and 5. If $\mathcal{X}$ denotes a set of scores (e.g., reviewers expertise score), a score $x_i \in \mathcal{X}$ is normalized into z_i as follows:

$$z_i = (b - a)\frac{x_i - min(\mathcal{X})}{max(\mathcal{X}) - min(\mathcal{X})} + a \tag{6}$$

In our case, both review age and reviewer expertise are scaled into the interval $[0, 1]$.

We concatenate the textual representation (x_h), expertise representation (h_s), and temporal representation (h_t) to get a final embedding

$$o_{final} = h_s \oplus x_h \oplus h_t, \tag{7}$$

where $\oplus$ is a concatenation operator. The final helpfulness prediction layer feeds o_{final} into a linear layer and use softmax activation to get the final predicted helpfulness class $\hat{Y}$.

$$\hat{Y} = \textbf{softmax}\ (W_r \cdot o_{final} + b_r), \tag{8}$$

[2]We present our dataset construction details in Section A of the Appendix.

Baseline Models	Acc. (↑)	MAE (↓)	MSE (↓)
ARH	58.73	0.476	0.619
UGR + BGR	62.76	0.464	0.674
TextCNN	62.82	0.444	0.608
MTNL	62.77	0.458	0.653
BERTHelp	63.03	0.432	0.591
Our Ablations	**Acc. (↑)**	**MAE (↓)**	**MSE (↓)**
RHP (*ours*)	**65.18**†	**0.393**†	**0.491**†
- *w/o Expertise*	63.87	0.421†	0.550†
- *w/o Temporal*	63.40	0.437†	0.592
- *w/o Expertise + Temporal*	62.92	0.446	0.617

Table 2: Performance compared to our baseline models and the result of our ablation study (↑ indicates higher values for a better performance and ↓ indicates lower values for a better performance). † reported results are statistically significant in paired t-test by taking BERTHelp (Xu et al., 2020) as a reference with the confidence of 95% (p-value < 0.05).

$$\mathcal{L} = \mathcal{L}_{CE}\left(\hat{Y}, Y\right) \tag{9}$$

where $W_r \in \mathbb{R}^{K \times K}$ and $b_r \in \mathbb{R}^{K}$ denote the projection parameter and a bias term respectively. We use the cross-entropy loss function $\mathcal{L}_{CE}$ with respect to the ground truths helpfulness class (Y).

3.3 Experiments

We evaluated the performance of the proposed model[3] compared to well-established baselines. We compare our system with ARH (Kim et al., 2006), UGR + BGR (Xiong and Litman, 2011), TextCNN (Chen et al., 2018), MTNL (Fan et al., 2018), and BERTHelp (Xu et al., 2020). We didn't perform any explicit preprocessing of the review text. We discuss the baseline systems, preprocessing, and hyperparameters used for our experiments in Appendix (Section B & Section C).

3.3.1 Results

As part of a detailed evaluation of our algorithm, we report our model's performance compared with the baselines in terms of Accuracy (**Acc.**), Mean Average Error (**MAE**), and Mean Squared Error (**MSE**). As shown in Table 2, our final model outperforms the baselines in terms of all the metrics. Our ground-truth values consist of 5 classes which correspond to five helpfulness scores $\{1, 2, 3, 4, 5\}$, where the higher the score, the more helpful the review. To gain more insights into the performance of our prediction model, we also evaluate our algorithm in terms of **MAE** and **MSE**, which assess the fine-grained differences between the ground-truth and the predicted helpfulness scores. Our **RHP** model consistently outperforms the baselines with a good margin, which means when misclassified, our model predictions are very close to the actual helpfulness scores. We conduct detailed ablation studies to demonstrate the effects of different components of our **RHP** model by removing expertise (denoted as *w/o Expertise*) and removing temporal information (denoted as *w/o Temporal*). The ablation test results on our dataset are summarized in Table 2. We can observe that the temporal feature has the largest impact on the performance of our model, and the impact of expertise is also significant. This suggests that the reviewer's expertise and temporal information of the reviews play a key role in review helpfulness prediction. Therefore, it is no surprise that combining all components achieves the best performance on our proposed dataset.

3.3.2 Analysis

We also present a detailed analysis to provide more supportive evidence of our arguments. To this end, we randomly selected m examples for each class of reviews considering helpfulness votes. Then, we extract Top K (where $K = 5$) n-grams from each class of reviews to identify the most relevant keywords or topics in reviews to assess what aspects are most talked about the items (e.g., hotels or restaurants).

Preprocessing Our preprocessing step includes tokenization, lemmatization, removal of stopwords, Part-Of-Speech (POS) tagging, and filtering punctuation marks. We use the NLTK[4] to preprocess each sentence and obtain a more accurate representation of the information. Moreover, we also add 'hotel' and 'restaurant' in the stopwords list as they frequently occur in every review and are not meaningful in our context.

Extracting Candidate n-grams We remove the sentiment words and emojis using VADER[5] (Hutto and Gilbert, 2014), a *"gold-standard"* sentiment lexicon especially attuned to microblog-like contexts. As the sentiment expressed in reviews are highly subjective, we are interested in extracting only the aspects or topics (e.g., *room*, *location*, *customer service* etc.) for which the opinions are

[3] Code, dataset, and model checkpoints: https://github.com/tafseer-nayeem/RHP

[4] https://www.nltk.org/

[5] https://github.com/cjhutto/vaderSentiment

Helpfulness Class	Unigram	Bigram
Class #1 Helpful Votes [1, 2)	**'room'**	**'front desk'**
	'staff'	**'coffee maker'**
	'location'	'breakfast buffet'
	'time'	'sofa bed'
	'service'	'swim pool'
Class #2 Helpful Votes [2, 4)	**'room'**	**'front desk'**
	'staff'	'shampoo conditioner'
	'service'	**'customer service'**
	'location'	**'resort fee'**
	'time'	'pool area'
Class #3 Helpful Votes [4, 8)	**'room'**	**'front desk'**
	'staff'	**'resort fee'**
	'time'	**'customer service'**
	'service'	**'coffee maker'**
	'view'	'city view'
Class #4 Helpful Votes [8, 16)	**'room'**	**'front desk'**
	'staff'	**'resort fee'**
	'service'	**'customer service'**
	'time'	'minute walk'
	'pool'	'life jacket'
Class #5 Helpful Votes [16, ∞)	**'room'**	**'front desk'**
	'time'	**'resort fee'**
	'service'	'bed bug'
	'staff'	'beach chair'
	'pool'	'cable car'

Table 3: Top 5 unigrams and bigrams extracted from five different classes of reviews divided according to helpfulness votes. For each column, **green** color indicates the overlap with all 5 classes, whereas **blue** for 4, **orange** for 3, and **red** for 2 overlaps.

expressed. Therefore, we keep only the nouns[6] (with POS tags 'NN' and 'NNS') for extracting the aspects or topics.

Ranking Candidate n-grams We extract the unigrams and bigram collocations for each of the review classes. Then, we rank the unigrams by counting the frequency of occurrences and bigrams using likelihood ratios (Manning and Schutze, 1999) to obtain Top K. We present the Top 5 unigrams and bigrams in Table 3 grouped according to helpfulness classes and ordered by descending ranking scores.

Table 3 shows a high overlap of n-grams among different classes of reviews, which further strengthen our argument that helpfulness does not entirely depend on the review text but rather the confluence of the review text, reviewing history of reviewers (***who*** *post the reviews*), review age (***when*** *the reviews are posted*). Generally, older reviews (i.e., review age) were present longer than the newer reviews in the platform and had more time to accumulate helpful votes.

[6]As adjectives and adverbs may contain sentiment towards aspects.

Figure 2: Top 10 ranked tokens of the **RHP** model shown in green colors with the color intensity indicating the importance of the tokens in the overall prediction.

3.4 Case Study

To gain more insights into the review helpfulness prediction task, we present a detailed case-study to interpret the model behavior and highlight the most important features of this task. Models are interpretable when humans can readily comprehend the reasoning behind model predictions and decisions made (Kim et al., 2016). To this end, we randomly selected a sample with Helpfulness Class = 3 from our test set and used `Captum`[7] to interpret the words/tokens that contributed the most to the prediction. As can be seen in Figure 2, the top-ranked words are highly representative of the aspects/facilities listed on the restaurant page. We can conclude from this observation that users tend to look for specific aspects in reviews to find them helpful. We also notice that the use of personal pronouns (e.g., I, we, they, etc.), describing personal experiences, contributes to the helpfulness prediction. People often find reviews useful if it comes from others' experiences and personal pronouns are a good indicator of it.

4 Conclusion and Future Work

In this paper, we develop a model incorporating the reviewer's expertise and temporal information in reviews to predict the helpfulness, especially for unreliable and cold-start reviews. Furthermore, we present a detailed analysis to interpret the model behavior and provide reasoning behind model predictions. For future work, we will look into the problem of personalized review helpfulness prediction to model the demographics and cultural differences of the reviewers.

[7]Captum (`https://captum.ai/`) is an open-source, extensible library for model interpretability that uses the integrated gradients method (Sundararajan et al., 2017).

Limitations

Despite the effectiveness of incorporating the reviewer's history and temporal information of the reviews in helpfulness prediction, our current studies still have several limitations, which can pave the path for future research.

For simplicity, like existing works, we assume that all the users rate reviews unanimously. However, the diversity of demographics, age, and cultural background also affect how users give, receive, and understand the sentiments expressed in reviews. Users may focus on different review aspects based on their preferences (i.e., *"5 stars, party every night"* **vs** *"5 stars, always quiet and peaceful"*). It would be interesting to see how to incorporate personal preferences for the helpfulness prediction task.

Another limitation of our work is that we only worked with reviews written in English. As a result, we filter out the reviews written in other languages and notice code-switched reviews when the reviewers alternate between two or more languages in a single review. We aim to extend this work to support more languages.

Ethics Statement

In our data scraping process, we took into account ethical considerations. We obtained data at an appropriate pace, avoiding any potential DDoS attacks. Additionally, we eliminated any Personal Identifying Information, such as names, telephone numbers, and email addresses, from the data set.

Acknowledgements

We thank all the anonymous reviewers for their valuable feedback and constructive suggestions for improving this work. This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and by a grant from Huawei. Mir Tafseer Nayeem is also supported by a Huawei Doctoral Scholarship.

References

Cen Chen, Yinfei Yang, Jun Zhou, Xiaolong Li, and Forrest Sheng Bao. 2018. Cross-domain review helpfulness prediction based on convolutional neural networks with auxiliary domain discriminators. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 602–607, New Orleans, Louisiana. 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.

Jiahua Du, Jia Rong, Hua Wang, and Yanchun Zhang. 2021. Neighbor-aware review helpfulness prediction. *Decision Support Systems*, 148.

Miao Fan, Yue Feng, Mingming Sun, Ping Li, Haifeng Wang, and Jianmin Wang. 2018. Multi-task neural learning architecture for end-to-end identification of helpful reviews. In *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '18, page 343–350. IEEE Press.

Sheng Khoo Foo, Lee Teh Phoey, and Boon Ooi Pei. 2017. Consistency of online consumers' perceptions of posted comments: An analysis of tripadvisor reviews. *Journal of information and Communication Technology*, 16(2):374–393.

Iftah Gamzu, Hila Gonen, Gilad Kutiel, Ran Levy, and Eugene Agichtein. 2021. Identifying helpful sentences in product reviews. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 678–691, Online. Association for Computational Linguistics.

Wei Han, Hui Chen, Zhen Hai, Soujanya Poria, and Lidong Bing. 2022. SANCL: Multimodal review helpfulness prediction with selective attention and natural contrastive learning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5666–5677, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 507–517, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

C. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225.

Rimma Kats. 2018. Surprise! most consumers look at reviews before a purchase. Accessed: May 10, 2022.

Been Kim, Rajiv Khanna, and Oluwasanmi Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 2288–2296, Red Hook, NY, USA. Curran Associates Inc.

Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. 2006. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, page 423–430, USA. Association for Computational Linguistics.

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.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Haijing Liu, Yang Gao, Pin Lv, Mengxue Li, Shiqiang Geng, Minglan Li, and Hao Wang. 2017. Using argument-based features to predict and analyse review helpfulness. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1358–1363, Copenhagen, Denmark. Association for Computational Linguistics.

Junhao Liu, Zhen Hai, Min Yang, and Lidong Bing. 2021. Multi-perspective coherent reasoning for helpfulness prediction of multimodal reviews. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5927–5936, Online. Association for Computational Linguistics.

Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. 2008. Modeling and predicting the helpfulness of online reviews. In *2008 Eighth IEEE International Conference on Data Mining*, pages 443–452.

Christopher Manning and Hinrich Schutze. 1999. *Foundations of statistical natural language processing*. MIT press.

Lionel Martin and Pearl Pu. 2014. Prediction of helpful reviews using emotions extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, page 43–52, New York, NY, USA. Association for Computing Machinery.

Rosie Murphy. 2016. Local consumer review survey 2016. Accessed: May 10, 2022.

Stefan Siersdorfer, Sergiu Chelaru, Wolfgang Nejdl, and Jose San Pedro. 2010. How useful are your comments? analyzing and predicting youtube comments and comment ratings. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 891–900, New York, NY, USA. Association for Computing Machinery.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR.

M Mudambi Susan and Schoff David. 2010. What makes a helpful online review? a study of customer reviews on amazon. com. *MIS Quarterly*, 34(1):185–200.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

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*, pages 502–507, Portland, Oregon, USA. Association for Computational Linguistics.

Wenting Xiong and Diane Litman. 2014. Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1985–1995, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

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 (Long and Short Papers)*, pages 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Shuzhe Xu, Salvador E. Barbosa, and Don Hong. 2020. Bert feature based model for predicting the helpfulness scores of online customers reviews. In *Advances in Information and Communication*, pages 270–281, Cham. Springer International Publishing.

Yinfei Yang, Yaowei Yan, Minghui Qiu, and Forrest Bao. 2015. Semantic analysis and helpfulness prediction of text for online product reviews. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 38–44.

A Dataset Construction

Publicly available datasets which are mostly used for this task are `Amazon`[8] (He and McAuley, 2016; McAuley et al., 2015) and `Yelp`[9]. In Yelp dataset, the user votes are distributed among three categories such as *"Useful"*, *"Funny"* or *"Cool"*, where *"Useful"* voting feature was introduced much later than the other two categories. Therefore, many good reviews already in the dataset may not have been marked useful. On the other hand, the Amazon dataset does not contain the reviewers' reviewing history and helpfulness votes to evaluate our hypothesis studied in this paper. Moreover, for Amazon, the samples come from various categories such as Books, Electronics, Clothing, Beauty, Shoes and Jewelry, Grocery, Pet Supplies, etc – the total helpfulness votes for the reviewers are coming from different categories and it's not explicit in the fields from Amazon website. Therefore, it's hard to devise expertise because of domain diversity.

We build our dataset by scraping reviews from `TripAdvisor`[10], a travel site that offers online hotel and restaurant reservations and a platform for sharing the travel experiences of users. We take reviews from January 1st, 2015 until January 1st, 2020, and extract only those written in English. For each review, we extract the review text, the total helpfulness votes and the posting time, and for each reviewer, we extract the number of reviews contributed and the cumulative helpfulness votes. The attributes we extracted are summarized as follows:

- **Reviews**
  - Review Text
  - Total Review Helpful Votes
  - Review Posting Time
- **Reviewers**
  - Total Number of Reviews Contributed
  - Cumulative Helpful Votes

B Baseline Systems

We compare our system performance with the following baselines.

- **`ARH`** (Kim et al., 2006) & **`UGR + BGR`** (Xiong and Litman, 2011) use machine learning-based methods with hand-crafted features such as *structural*, *lexical*, *syntactic*, *emotional*, *semantic*, and *meta-data* from the review text to address this task. These features are fed into conventional classifiers such as SVM, Random Forest, and gradient boosting to identify helpful reviews.
- **`TextCNN`** (Chen et al., 2018) employs a text-based CNN model (Kim, 2014) to automatically capture the character-level, word-level, and topic-level features for helpfulness prediction.
- **`MTNL`** (Fan et al., 2018) utilizes end-to-end multi-task neural learning (MTNL) architecture for classifying helpful reviews. They take the help of an auxiliary task, such as rating regression, to boost the performance of the original task, which is review helpfulness identification.
- **`BERTHelp`** (Xu et al., 2020) develop their helpfulness prediction model using pre-trained BERT (Devlin et al., 2019). They design a regression model using BERT-based features extracted from review texts, star rating, and product type information from Amazon product review dataset (He and McAuley, 2016).

C Preprocessing & Hyperparameters

Preprocessing We didn't perform any explicit preprocessing of the review text. Instead, we use BertTokenizer to avoid the out-of-vocabulary (**OOV**) problem, which uses `WordPiece` (Wu et al.,

[8] `http://jmcauley.ucsd.edu/data/amazon/index_2014.html`
[9] `https://www.yelp.com/dataset`
[10] `https://www.tripadvisor.com`

2016) for tokenizing the sentences into words or subwords. In addition, we add special tokens to the start (e.g., [CLS]) and end of each review text (e.g., [SEP]) and truncate all sentences to a single constant length (e.g., 512).

Hyperparameters We use Adam optimizer (Kingma and Ba, 2015) with a learning rate of $3 \times e^{-5}$ and a batch size of 32. We use $\text{BERT}_{\text{BASE}}$ (Wolf et al., 2020) pre-trained model with a fixed vocabulary. We run the training for 5 epochs and check the improvement of validation (*dev set*) loss to save the latest best model during training.