ReviewGen: Streamlining Paper Evaluation through Explainable Language Modeling and Joint Feature Prediction

Cynthia Chen
cynthiachen@college.harvard.edu

Eric Hansen
erichansen@college.harvard.edu

Rohan Doshi
rdoshi@mba2023.hbs.edu

Abstract

This work explores the effectiveness of large language models (LLMs) in automating the review of research papers. Our approach aims to build trust in automated predictions by jointly predicting multiple signals to drive performance and offer greater explainability. Specifically, our model predicts ratings as a proxy for paper quality, textual reviews as a proxy for rationale, confidence scores as a proxy for consistency between multiple reviewers, and citation count as a proxy for eventual impact. Using a dataset of ICLR and NeurIPS papers, we find that joint prediction improves performance for predicting reviews, citations, aspect categories, ratings, and confidence scores; however, loss weighting impacts the relative performance of each sub-task. Additionally, we conduct a qualitative analysis of the review output and attention maps to understand model failure modes and prediction explainability. This analysis reveals the "summary" section of reviews tend to be high-quality, whereas others (e.g. clarity) tend to be generic, repetitive (e.g. "the paper is well written"), and occasionally incorrect.

1 Introduction

Reviewing papers is at the heart of the peer review process for scientific research, given how scores establish quality standards while reviews build faith in the process. However, reviewing and assessing papers is extremely time-consuming for researchers, especially given the quickly-growing volume of submissions to both conferences and journals [3].

This trend has led to recent research in leveraging large language models for automating the paper review process (Yuan et al., Bartoli et al.). While past works have tried predicting reviews or ratings in isolation, they fail to build trust by offering limited clarity and interpretability into the paper's quality. For example, generated reviews may simply summarize rather than evaluate a paper whereas an individual rating lacks any explanation, leading to a lack of trust in model output.

In this work, we want to explore the effectiveness of large language models (LLMs) in reviewing a corpus of scientific papers. We aim to develop a model that is able to jointly predict citations, confidence scores, ratings, and textual reviews as outputs to build trust. Specifically, we investigate how the concurrent decoding of multiple signals impacts both performance as well as explainability. Finally, we hope to share the learnings from our error analysis of our joint model to better understand the strengths and shortcomings of our approach and advance future efforts to develop automated paper review systems.

2 Related Work

Previous work in the field of systematic review (SR) automation is discussed by Schulz et al., who claim that “existing tools cannot understand or interpret the paper in the context of the scientific literature” and therefore can only serve as an enhancement to human-led peer review [7]. Most modeling applications in SR automation focus on searching, screening, or summarization (rather than explanation generation) [8]. However, the domain of using automated models (e.g. machine learning models) to aid in the peer review process has been relatively unexplored. Bartoli et al. devised a tool to generate fake paper reviews based on term replacement, and Wang et al. create a method based on knowledge graphs to perform domain-specific information extraction and provide a review score with associated evidence [1] [5]. Notably, Yuan et al. develop an LLM model that aims to generate reviews that are both syntactically and semantically similar to human reviews by adding an "aspect score[1]" token prediction task on top of the normal sequence to sequence (Seq2Seq) approach [11].

Here, we build upon the work of Yuan et al. in [11], specifically using their ASAP-Review dataset and their modeling approach as a starting point. Our baseline implementation pre-processes a dataset to extract relevant parts of the paper and uses a Seq2Seq BART model to generate reviews. Our work seeks to make the following additional improvements.

First, we want to see if we can additionally predict a paper’s impact, using a novel citations dataset using pre-processed paper text as input. Second, we want to study whether the task of predicting different subsets of paper ratings, citations, confidence scores, conference acceptance decisions, aspect scores, and reviews jointly can improve model performance on each of the respective subtasks. Finally, we hope to increase the interpretability of our methods by performing human evaluations of the reviews generated by different models and investigating attention maps to reveal which parts of the paper text the model is attending to in the generated reviews.

3 Data Preparation & Methodology

3.1 Overview

The goal of our models is to take as input a research paper and provide four meaningful outputs: a numerical rating reflecting the paper’s quality, a textual review providing a rationale for the rating, a confidence score on how likely a paper is to be accepted, and a log citation count. The code developing and evaluating these models can be found at this repository.

3.2 Initial Dataset

Our full dataset is comprised of 28,000 papers and review pairs from the ICLR (2017-2020) and NeurIPS (2016-2019) conferences. The dataset also includes data on which papers were accepted, the rating provided by the reviewer (1-10 scale) for ICLR, and the reviewer’s confidence for ICLR 2017-2019 (1-5 scale) and NIPS 2016 (1-3 scale). All the papers in the dataset from NeurIPS were accepted. This dataset was compiled by Yuan et. al [11] as part of their own paper "Can We Automate Scientific Reviewing?" We filter reviews by length (min 100 words and max 1024 words), yielding 22,000 paper-review pairs.

We augment this dataset by classifying the tokens in each review with one of 15 aspect scores, or "None". We use the BERT model trained in the Yuan et. al paper for this task. Aspect scores refer to categories in which a human reviewer may evaluate an academic paper (e.g. clarity, soundness) and, except for the "summary" category, include a polarity as well (e.g. positive or negative). More details can be found in the Appendix A.2.

We explored expanding our dataset by adding papers from NeurIPS (2013-2017) and Arxiv (2007-2017), which are both used in Kang et. al. The scraping metbyhods used by Kang et. al, however, are outdated due to the relocation of the source files, so these could not be easily obtained. We also attempted to obtain a dataset from Elsevier through the Harvard Data Science Initiative, but were delayed by response times and the need for a formal research agreement. Our experiments were already constrained by compute resources, so we did not make further attempts to expand our dataset.

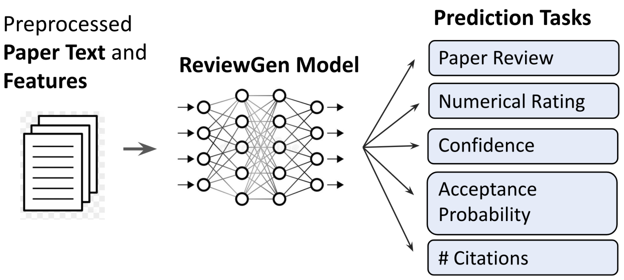
[1] As discussed in section 3.2, aspect scores refer to categories in which a human reviewer may evaluate an academic paper (e.g. clarity, soundness).

3.3 Citations

A novel contribution of this project was obtaining citation data for the majority of unique papers (6323) in this dataset. We fetched this data on Google Scholar, which posed minor challenges due to their limits on automated requests. While reviewer ratings provide a great numerical evaluation of the quality of a paper, including citations as a prediction task also allows us to provide an evaluation of the potential **impact** of a paper.

Since the distribution of papers' citation counts (i.e. CitNum) has a long tail, we decided to predict the logarithm of citation count (i.e. LogCitNum) in our models. Given that we use the Mean Squared Error as our loss function, we wanted a roughly normal distribution for each of our numerical predictors[2]. As shown in figure 2 in the Appendix, the distribution of logCitNum in our dataset is approximately normal.

3.4 Training Algorithm(s)



To jointly predict reviews, ratings, confidence scores, aspect scores, and log citations, we fine-tune BART, a large-language model. The model tokenizes textual inputs, namely reviews, using a 50K token vocabulary. The review prediction head outputs a token sequence of at most 1024 tokens.

Our joint-prediction model has six decoding heads. We use a weighted loss function that includes a cross-entropy loss on tokens for review prediction, a mean square error (MSE) for rating prediction, a binary cross-entropy (CE) loss on conference decision classification, an MSE loss for confidence score prediction, a CE loss for aspect score prediction across 16 aspect score categories, and an MSE loss on the log citation prediction. In future sections, we refer to the Aspect Score prediction task as "NER", which stands for Named-Entity-Recognition.

The review generation head leverages the default sequence prediction layer in the BART model to output a sequence of tokens. The rating, confidence, decision, and citation remaining prediction heads are derived from a 1024 dim embedding generated after applying 1D convolutions to the transformer's final hidden layer outputs. Each head applies its own, separate MLP with two hidden layers (4096, 2048) and a terminal fully connected layer to output a final 1-D predicted output. The exception is the aspect score head, whose MLP takes the 1024-dim embedding for each token directly (no convolution over the sequence) and leverages a similar two-hidden-layer MLP to map each token to a 16-dim output with normalized class probabilities.

Regarding hyperparameter and architecture optimization, we tuned our hyperparameters via experimentation on sub-samples of data due to compute limitations. These experiments found that a per-head MLP with two hidden layers outperforms one with one hidden layer. Additionally, we set weight decay to 0.001 and learning rate to 5e-5 and epochs to 3 for the initial sole review generation task. We increased epochs to 5-10 for joint prediction tasks because training took longer to converge, likely because of competing gradients from each prediction task during backpropagation.

[2] Notably, we did not adjust citation counts for years since publication, but since it has been at least 3 years and at most 6 years since each of these papers appeared in NEURIPS or ICLR, we reasoned that this adjustment would change the relative citation count of a paper by no more than a factor of 2.5 in the worst case and significantly less in the average case, so this omission is a relatively minor concern, especially given we are predicting the logarithm

3.5 Paper Text Preprocessing

A key challenge associated with large document analysis using LLMs is that transformer models have maximum input lengths. We replicated the Yuan et. al paper's approach of fine-tuning the bart-large-cnn pre-trained model on the review generation task, which restricts our input sequence length to 1024 tokens. [11] Most papers significantly exceed this threshold, so we must use an extraction method to down-sample the text while still retaining the content necessary to provide an informative review.

We replicated and evaluated three extraction methods from Yuan et. al to create a dataset of representative text inputs for each paper under 1024 tokens. In the first approach, we consider only the introduction section from the paper alone. In the second approach, we leverage a method (i.e. cross-entropy extraction) that identifies sentences with certain informative keywords (e.g. propose) from the paper and then selects a subset of those sentences that have diverse keyword coverage and satisfy the length constraint. The third approach combines the cross-entropy extracts with the paper abstract. Notably, the cross-entropy method took >16h to run locally across 16 cores for 8,877 papers, which is longer per-paper than obtaining outputs from a trained model. After applying these extraction methods, a few results still had a greater number of words than the maximum length of 1024, so we omitted them. We also applied a minimum review and input text length of 100 tokens. After evaluating the three extraction methods, as shown in table 4 in the Appendix, we selected the "hybrid" approach for subsequent joint prediction experiments.

3.6 Summary Statistics & Correlation Analysis

To contextualize our experimental results and also illustrate a few interesting insights into the relationship between ratings, conference acceptances, and citations, we share the following data analysis results, detailed in tables 2 and 3 in the Appendix. First, the mean paper has roughly 5000 words, showing that we were significantly downsampling our input text. More than 75% of papers have more than 4096 tokens, so even if we were to use the Longformer LLM from Huggingface, which has a maximum input size of 4096 compared to BART's 1024 token limit, we would still require truncation or pre-processing to downsample the input text [2]. Second, ratings and conference acceptances have a notably high correlation of 0.55 and both correlate significantly (0.4) with citation count, suggesting reviewers and conferences are relatively good predictors of a paper's impact[3].

3.7 Evaluation Metrics

To evaluate the quality of our generated reviews, we use standard Seq2Seq evaluation metrics from the existing literature, Rouge-1 and Rouge-2 scores, which measure the overlap of unigrams and bigrams between the generated review and the actual review. We also report an unscaled BertScore f1_score value (BS F1), which was developed by Zhang et. al [12] and reflects the overall similarity of two phrases, the actual and generated reviews in the embedding space of a fine-tuned BERT model.

For decision classification, we use ROC AUC and for ratings, citation counts, and confidence scores, we use root MSE (RMSE) for evaluation. For aspect scores, we use accuracy.

4 Experimental Results & Analysis

4.1 Joint Prediction Model Performance

In the table 4 in the Appendix, we share the Seq2Seq evaluation results for the BART model fine-tuned using the three different paper pre-processing approaches, showing significant improvements over the baseline BART model and best results for the CE and hybrid methods. We use the hybrid processing method for subsequent models. Regarding model training, we found that models with more prediction tasks, especially if more disparate (e.g. Review+Citations required more than Review+NER), require more training epochs to converge and that scaling loss between different prediction tasks had a

[3] As an aside, ratings, but not conference acceptances or citations, also had a non-negligible correlation (0.2) with paper text length. A possible explanation is that longer papers include more technical details, comparisons, etc. that satisfy reviewers who may look for soundness and completeness in their evaluations rather than just impact

significant impact on the model's relative performance on each task. To normalize the impact of each prediction task's loss on the overall loss function, we normalized citations, ratings, and confidence scores to be distributed Normal(0, 1) and then scaled their losses respectively to be roughly 0.5, 0.25, and 0.25 respectively the magnitude of the review generation loss on average after convergence.

In the table 1 below, we show the quantitative performance of BART models trained on different subsets of the review generation, confidence, rating, citation, decision, and aspect score prediction tasks on the relevant evaluation metrics. We grouped confidence scores, ratings, and conference decisions as "Metadata."

Model	R1	R2	BS F1	NER Acc	Cit RMSE	Dec AUC	Rat RMSE
Review Only	31.3	7.9	83.5				
Review + NER	39.8	10.4	84.1	73.4			
Citations Only					1.22		
Review + Citations	35.4	9.1	83.5		1.21		
Citations + Metadata					1.22	0.53	0.95
All	38.5	9.7	83.7	72.2	0.98	0.64	1.0

Table 1: Rouge-1, Rouge-2, Unscaled BertScore F1, Named Entity Recognition (NER) Classification accuracy on Aspect Scores, Citation RMSE, Decision AUC, and Ratings RMSE

As shown, we can significantly improve our ability to predict citations (1.22 to 0.98 RMSE) and conference acceptances (0.53 to 0.64 AUC) with no significant change in rating prediction or Rouge and BertScores by jointly generating reviews with NER classification. One possible explanation is that including multiple prediction tasks (e.g. citations, ratings, conference acceptance) improves generalization since the model's training loss will be maximized when all prediction task outputs are aligned, in which case that data point has a higher signal-to-noise ratio. A second possible explanation is that given the BART model architecture is optimized for Seq2Seq applications, including a core Seq2Seq prediction task is necessary for proper tuning of model weights during training, which was corroborated by our seeing minimal validation loss decreases after the first epoch in the Citation and Citation+Metadata alone models during training. Notably, given that our input data is distributed roughly standard normally, even with outliers, a 0.98 and 0.95 RMSE is still relatively high. A 0.64 decision AUC is also relatively low for binary classification, showing that using LLMs, even after these improvements, to automatically predict a paper's quality and impact given pre-processed paper text alone is not a silver bullet.

4.2 Qualitative Evaluation of Generated Reviews

We also carried out a qualitative evaluation of the output reviews from these different models. Our primary findings are that the reviews from the Review+NER model are the most cogent. We found that the output reviews from the "All" model (Review+NER+Citation+Metadata), despite offering similarly strong summaries of the paper's content and showing similar Rouge-1, Rouge-2, BertScores, and NER accuracy, contain more frequent "nonsense" sentences such as "*Finally, there is no comparison between the method and the method used in the paper. It is unclear why the authors didn't compare the method to the method proposed in this paper. For instance, what is the difference between missing data and missing data?*"

Across all models, the model-generated reviews perform well in **summarizing** the paper, but struggle to **critically evaluate** its clarity, technical soundness, originality, etc. The addition of the NER aspect score loss on individual tokens seems to ensure that the review output comments on the paper's clarity, originality, etc., but these sentences are usually generic, repetitive, and occasionally inaccurate. For instance, the phrase "*the paper is well-written and easy to follow*", commenting on clarity, appears in a majority of reviews and variations on the phrase "*the paper should compare its results to the current state-of-the-art*" also appear frequently. More concerning, occasionally the review output is false, for instance, claiming that "*the authors do not provide a clear explanation of how the proposed ResNet is different from the existing ResNet architecture*" for Feichtenhofer et. al's paper [4]. As an additional experiment, we created isolated datasets for each aspect score consisting of the sentences from the training set reviews that include a token labeled for that aspect score. We found that training models on this reduced dataset, i.e. training a model to only comment on the clarity of an input paper, exacerbated the issue of outputting generic, repetitive, and inaccurate text for all categories except "summary."

Reflecting on this finding, we discussed how it may be unrealistic to expect general language models, such as BART, fine-tuned only on paper+review data to be able to make intelligent critical evaluations of academic papers, especially commenting on the technical soundness of methods and comparing results to the existing literature, which requires up-to-date domain knowledge. Second, there is wide variability in the dataset in terms of the number of tokens assigned to each aspect score, which makes training more challenging because of the range of potential loss outcomes. Finally, the Seq2Seq training approach, which uses a cross-entropy loss on individual tokens, is likely not the best approach to evaluate the clarity, substance, etc. of a paper because of the repetition of common phrases and words, which induces the model to output repetitive sentences without any critical content. Training a model to predict a **numerical score** for each of these categories could potentially provide a more interesting and interpretable evaluation if such a dataset could be developed.

We include the full generated review output from the Review Only and Review+NER models in section A.3 of the Appendix as reference. Additional review output from different models can be found in this file from our GitHub repository.

4.3 Attention Maps

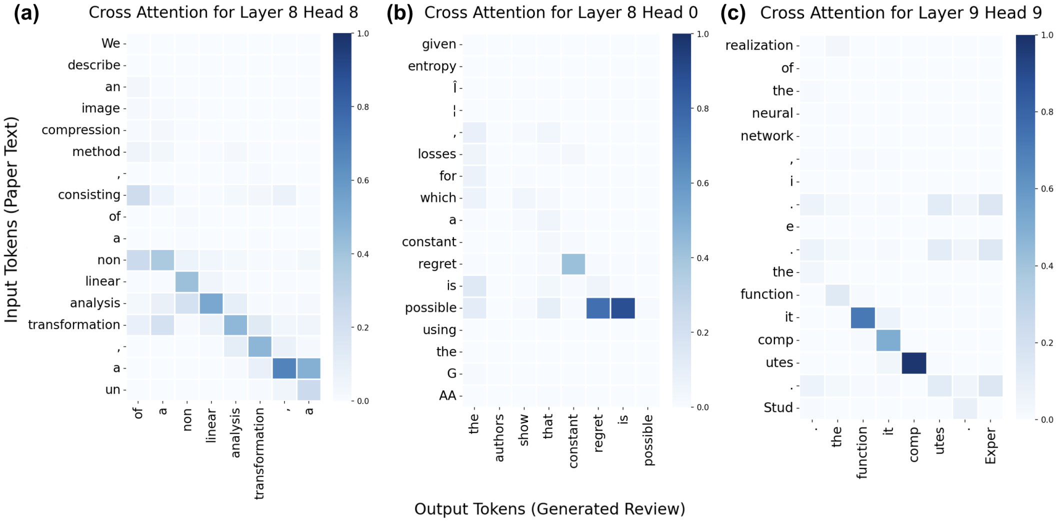


Figure 1: Heatmaps of the cross attention placed on various input tokens (from the original paper text) to the output tokens (in the model-generated review). Here, we display a subset of the patterns found in the attention heatmaps for three example attention heads (layer 8 head 9, layer 8 head 0, and layer 9 head 9) but the attention maps can easily be generated for other attention heads.

To investigate patterns between the original paper text and the model-generated review, we developed methods to visualize the cross-attention between input and output tokens. The input tokens correspond to the paper text, while the output tokens correspond to the model's review. To visualize attention, we obtained the encoder and decoder tokens for a sample paper and its associated model-generated review, extracted the cross-attention scores (attention placed from encoder to decoder tokens) from the BART model, and then visualized these attention values across the entire paper and review. Since the paper and review consist of hundred of tokens, we then visualize a subset of the attention matrix in heatmap format, as shown in Figure 1.

Using these heatmaps, we observed patterns of high attention being placed from input tokens to similar output tokens. We specifically note the behavior of decoder tokens placing attention on the token before its corresponding token in the input, as seen by the distinct diagonal pattern in Figure 1a and 1c. These are interesting patterns that elucidate how the model associates tokens in the review with tokens in the paper itself, and provide useful insight into how exactly the model comes up with the generated review. Further investigation could seek to analyze these attention patterns at a more global scale across more heads and sample papers.

5 Conclusion

In this project, we developed a set of models to quantitatively and qualitatively evaluate academic papers by outputting a predicted rating, citation count, conference acceptance decision, confidence score, and explanatory natural language review. We found that predicting reviews alongside quantitative metrics improved performance on these metrics with slight negative impact on the review quality. Generally, we found that including Aspect Score prediction also increased the performance of Seq2Seq evaluation metrics and generated more comprehensive reviews, but that the critical sections of these reviews, as opposed to the summary sections, were repetitive and generic, demonstrating that automated approaches using existing datasets and LLMs may not be able to provide fully trustworthy evaluations of scientific papers.

5.1 Limitations

We noted a few key limitations throughout this paper including the lack of numerical scores for "aspect" categories, limited hyperparameter optimization and additional experimentation due to compute restrictions, and the need to down-sample paper text due to transformer model maximum input lengths. We would also note that our dataset consists only of papers in the field of computer science, so our model may not generalize well to other types of papers. Moreover, evaluating review generation is a complex task. Rouge-1, Rouge-2, and BERT scores, even if the best quantitative metrics available for Seq2Seq tasks, may not capture the quality of a review since they cannot evaluate its correctness and cogency.

5.2 Future Work

As noted, improving the model's ability to critically evaluate papers in terms of their clarity, substance, soundness, and other aspect score categories noted in section A.2 is an important area of future work and could benefit from pre-trained models on a wider body of text such as GPT-4. Alternatively, developing a dataset of numerical scores for each category or a fixed number of tokens for each Aspect Scores could provide modest improvements in developing a more interpretable model evaluation. Finally, developing a novel Seq2Seq loss function to down-weight predicting common, generic phrases could provide additional improvement in the review quality.

Additional areas of future work could include more extensive user studies on the generated reviews, a sample of which we make available in our repository, to reveal additional failure modes, and more analysis on optimal hyperparameter parameters (loss scaling, epochs, MLP dimensions, weight decay), or adapting other pre-trained models (e.g. Longformer) with a greater maximum input text length for this Seq2Seq task [2].

Author Contributions

Eric Hansen (A1), Rohan Doshi (A2), and Cynthia Chen (A3) came up with the problem statement. A1, A2, and A3 all came up with the key ideas of the main methodology of joint-training. A1 led efforts to pre-process and analyze the dataset and obtain citations data. A2 led efforts to implement the modeling code for joint-training, and A3 oversaw the cross-attention mechanism visualization effort for error analysis. A1 led the qualitative analysis of sample outputted reviews from the model. A1 and A2 designed and implemented the experimental evaluation. A3 led the preparation of the final presentation with inputs from A1 and A2. A1 and A2 led the writing of the final project report along with inputs from A3. A1 documented the code and created scripts for running the entire pipeline with just a handful of commands. All the authors also helped each other with brainstorming as well as double-checking their work.

References

[1] A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao. Your paper has been accepted, rejected, or whatever: Automatic generation of scientific paper reviews. In *Availability, Reliability, and Security in Information Systems: IFIP WG 8.4, 8.9, TC 5 International Cross-Domain Conference, CD-ARES 2016, and Workshop on Privacy Aware Machine Learning for Health Data Science, PAML 2016, Salzburg, Austria, August 31-September 2, 2016, Proceedings*, pages 19–28. Springer, 2016.

[2] I. Beltagy, M. E. Peters, and A. Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[3] L. Bornmann and R. Mutz. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *Journal of the Association for Information Science and Technology*, 66(11):2215–2222, 2015.

[4] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal residual networks for video action recognition. *CoRR*, abs/1611.02155, 2016.

[5] M. Kachuee, K. Kärkkäinen, O. Goldstein, S. Darabi, and M. Sarrafzadeh. Generative imputation and stochastic prediction. *CoRR*, abs/1905.09340, 2019.

[6] 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. *CoRR*, abs/1804.09635, 2018.

[7] R. Schulz, A. Barnett, R. Bernard, N. J. Brown, J. A. Byrne, P. Eckmann, M. A. Gazda, H. Kilicoglu, E. M. Prager, M. Salholz-Hillel, et al. Is the future of peer review automated? *BMC Research Notes*, 15(1):1–5, 2022.

[8] G. Tsafnat, P. Glasziou, M. K. Choong, A. Dunn, F. Galgani, and E. Coiera. Systematic review automation technologies. *Systematic reviews*, 3:1–15, 2014.

[9] J. Vig. Bertviz: A tool for visualizing multihead self-attention in the bert model. In *ICLR workshop: Debugging machine learning models*, 2019.

[10] Q. Wang, Q. Zeng, L. Huang, K. Knight, H. Ji, and N. F. Rajani. Reviewrobot: Explainable paper review generation based on knowledge synthesis. *arXiv preprint arXiv:2010.06119*, 2020.

[11] W. Yuan, P. Liu, and G. Neubig. Can we automate scientific reviewing? *Journal of Artificial Intelligence Research*, 75:171–212, 2022.

[12] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

A Appendix

A.1 Dataset Analysis & Paper Pre-Processing Results

	Review Tokens	Paper Text Tokens	Rating	Confidence	CitNum
Mean	375.0	4959.0	5.0	4.0	185.0
Std Dev	246.0	1510.0	2.0	1.0	1025.0
5%	104.0	3424.0	1.0	2.0	2.0
25%	208.0	4172.0	3.0	3.0	15.0
50%	317.0	4684.0	5.0	4.0	44.0
75%	477.0	5340.0	6.0	4.0	128.0
95%	843.0	7362.0	8.0	5.0	670.0
Max	3900.0	29151.0	10.0	5.0	71907.0

Table 2: Dataset Summary Statistics

	LogCitNum	Decision	Confidence
Rating	0.39	0.56	-0.12
LogCitNum		0.38	0.0
Decision			-0.23

Table 3: Correlations across Paper Evaluation/Impact Metrics

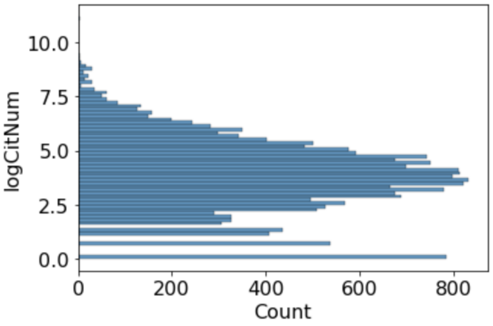


Figure 2: Log Citation Distribution

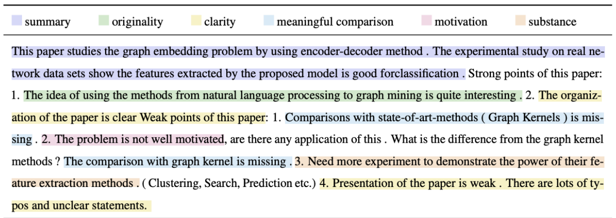
	Rouge-1	Rouge-2	Rouge-L	BS F1
Base BART Model	20.5	3.0	11.0	82.2
Intro Only	30.0	7.7	16.2	82.7
Cross-Entropy	32.6	8.4	17.2	83.6
Hybrid	31.3	7.9	16.2	83.5

Table 4: Rouge-1, Rouge-2, Rouge-L, and unscaled BertScores on Review Generation task using different Pre-Processing Methods

A.2 Aspect Score Details

As developed in Yuan et. al [11], we use a trained BERT token classification model to identify words and phrases in each of the reviews in the training set that correspond to one of the following categories: summary, clarity, meaningful comparison, originality, soundness, substance, motivation, replicability. For the latter 7 categories, the model separately labels tokens as positive or negative(e.g.

"clarity_positive"). In figure 3 below, the authors from Yuan et. al provide an example of sentences that correspond to different categories.



summary originality clarity meaningful comparison motivation substance

This paper studies the graph embedding problem by using encoder-decoder method . The experimental study on real network data sets show the features extracted by the proposed model is good forclassification . Strong points of this paper: 1. The idea of using the methods from natural language processing to graph mining is quite interesting . 2. The organization of the paper is clear Weak points of this paper: 1. Comparisons with state-of-art-methods (Graph Kernels) is missing . 2. The problem is not well motivated, are there any application of this . What is the difference from the graph kernel methods ? The comparison with graph kernel is missing . 3. Need more experiment to demonstrate the power of their feature extraction methods . (Clustering, Search, Prediction etc.) 4. Presentation of the paper is weak . There are lots of typos and unclear statements.

Figure 3: Aspect Score Labelled Text

In table 5, we note the correlation between the number of aspect score tokens in a review and the number of citations a paper receives. Though all of these correlations are slight, it suggests that soundness, clarity, and substance are the most important review categories in terms of predicting a paper's impact.

	Clarity	Comp	Motiv	Orig	Repl	Sound	Subst
Num Pos Tokens	0.06	0.03	0.04	0.03	0.02	0.07	0.08
Num Neg Tokens	-0.06	-0.03	-0.05	-0.03	-0.01	-0.07	-0.04

Table 5: Correlation of Citations with Number of Aspect Score Tokens in Reviews

A.3 Generated Review Output

Paper Title: Generative Imputation and Stochastic Prediction (Kachuee et. al 2019)

Review Only Output: This paper proposes a method for imputing missing features and estimating the distribution of target assignments given incomplete data. The method is evaluated on CIFAR-10 image dataset as well as three real-world tabular classification datasets, under different missingness rates and structures.The paper is well-written and well-motivated. The proposed method is interesting and the experimental results are convincing. However, the paper is not well-suited for ICLR due to the following reasons: 1) The paper is mainly about imputing features and it is not clear whether the proposed method can be applied to other datasets. For example, in Table 1, the method is applied only to Cifar-10 and Table 2, it does not work for other datasets, e.g., UCI Landsimage dataset or Diabetes classification dataset. 2) There is no comparison with other imputation methods in the literature. For instance, in Section 3.2, it is stated that: "The authors propose a method based on deep latent variable models and importance sampling that offers a tighter likelihood bound compared to the standard VAE bound." This is a strong statement, but the authors do not compare with other methods in terms of performance.

Review+NER Output: This paper proposes a method for imputing missing features and estimating the distribution of target assignments given incomplete data. The proposed method is evaluated on CIFAR-10 image dataset as well as three real-world tabular classification datasets, under different missingness rates and structures.The paper is well written and easy to follow. The method is well-written and well-motivated. The experimental results show the effectiveness of the proposed method in generating imputations and providing estimates for the class uncertainties in a classification task when faced with missing values. The paper is clearly written and well written. The experiments are well-constructed. The results are interesting and the paper is easy to read. However, there are a few things that I would like to see more of. For example, I would have liked to see the impact of different imputations for different features on the final hypothesis. For instance, for a specific test sample at hand, if a certain missing feature is a strong indicator of the target class, it would be interesting to see how the imputations are affected by the missing features. I would also like the authors to provide more details on the training process of the generator and predictor networks. I am not sure how to interpret the results of the experiments.