Research Paper Publishability Assessment Report

Team Subhashreekedia

Introduction

The exponential growth in the number of research papers being published across various domains has posed significant challenges for reviewers tasked with assessing their quality and suitability for publication. Manual evaluation processes are not only time-intensive and labor-intensive but also prone to subjectivity and inconsistencies, making it increasingly difficult to maintain high standards of accuracy and objectivity. This has created an urgent need for automated systems that can streamline the evaluation process while ensuring fairness, transparency, and scalability.

Our work addresses this challenge by leveraging advanced computational techniques, including **knowledge graphs, semantic chunking, cluster detection, node ranking algorithms, vector embeddings, and large language models (LLMs)**. These methods enable the systematic extraction and analysis of critical features from research papers, such as the coherence of arguments, appropriateness of methodologies, and robustness of evidence. By integrating these technologies, the proposed framework not only enhances the efficiency and accuracy of the review process but also provides detailed rationales for its decisions, **tackling the critical issue of explainability in AI systems.**

The framework utilized **Pathway's Drive Connector** for seamless data ingestion and its **VectorStore** for efficient management and retrieval of embeddings, ensuring scalability and streamlined processing.

By addressing issues such as inappropriate methodologies, incoherent arguments, and unsubstantiated claims, the framework ensures consistent evaluation standards across diverse research domains. It demonstrates the potential to revolutionize the peer-review process, providing an automated, scalable, and objective solution to the challenges posed by the rapidly growing volume of academic publications.

Methodology

Motivation

- **Task 1:** The classification of research papers into publishable and non-publishable categories presented unique challenges due to the inherent differences between the dataset provided and the unpublished papers available on the internet. The given dataset contained unpublished papers that often combined abstract concepts from entirely different fields, whereas the majority of unpublished papers accessible online were predominantly focused on a single domain. This disparity highlighted the need for a nuanced approach that could accurately evaluate the coherence and conceptual connectedness of papers, especially in cases where diverse fields were intertwined.

During our analysis, we observed that the primary reasons for rejection of papers as unpublishable were a lack of coherence and an overall disconnect between the concepts presented. This insight motivated us to incorporate knowledge graphs as a means to systematically analyze and visualize these parameters. Using **Neo4j** for graph visualization, we identified two significant patterns: publishable papers exhibited well-connected nodes, with clusters that were at least connected through **2-3 intermediary nodes**, whereas unpublishable papers were characterized by **disconnected clusters**. In cases where unpublishable papers had a single cluster, it often comprised **1-2 dominant nodes**, with other nodes diverging and exhibiting minimal connectedness.

To quantify these relationships, we applied a variety of ranking algorithms to the graph nodes, including chromatic numbers, connected components, in-degree, out-degree, graph density, and PageRank. Among these, **graph density and PageRank** emerged as the most effective metrics for encapsulating the underlying relationships between nodes, providing valuable insights into the connectedness and coherence of the papers.

Additionally, we leveraged **zero-shot classification techniques** for the initial categorization of papers. This approach further emphasized the importance of knowledge graphs in bridging the gap between disparate concepts and highlighted their potential for improving the robustness and accuracy of the classification framework. These observations and subsequent validations shaped our methodology, driving the development of an explainable system capable of addressing the unique challenges associated with research paper evaluation.

- **Task 2:** Increase objectivity in evaluating diverse research outputs. The motivation for this method arises from the growing challenges in efficiently classifying research papers into specific conferences amidst an ever-expanding volume of scientific publications. Traditional methods struggle to scale and capture the nuanced, domain-specific language of research papers. To address this, the proposed approach integrates automated text extraction, semantic processing, and classification techniques, ensuring scalability and accuracy. Leveraging SciBERT, a language model optimized for scientific text, enables deep semantic understanding, surpassing the limitations of keyword-based methods by capturing contextual relationships and thematic nuances.

  This approach combines text chunking, embeddings for contextual representation, and sequence classification models to differentiate between the subtle thematic and stylistic variations across conferences like NeurIPS, CVPR, EMNLP, TMLR and KDD. By utilizing vector stores and similarity searches, it effectively handles unlabeled data, bridging gaps in supervised classification. Additionally, the method facilitates researchers by providing structured insights and clustering related work, advancing discovery and collaboration. Through automation, domain-specific optimization, and scalability, this pipeline addresses key challenges in academic research classification while remaining adaptable to diverse domains.

  The rationale generation aims to enhance interpretability and trust in research paper classification by providing context-specific justifications. It aligns papers with the thematic focus of conferences and ensures transparency, aiding researchers in validating classification results.

Method for Task 1

To classify research papers into "Publishable" and "Non-Publishable" categories, we employed a multi-step methodology leveraging advanced algorithms, natural language processing techniques, and graph-based approaches. This section details the components of our method, including a novel text extraction technique developed for this task.

Novel Text Extraction Method

The initial step involved extracting text from research papers in PDF format. Due to the variations in PDF formatting and the challenges of preserving semantic structure, we developed a novel text extraction algorithm to handle whitespace, character alignment, and ordering issues. Our method sorts characters based on their positional attributes and reconstructs the text with appropriate spacing. This ensures that the extracted text maintains coherence and readability, even in complex layouts.

The algorithm is implemented as follows:

- Extract characters from a PDF page using the `pdfplumber` library.
- Sort characters by their top and left (x0) coordinates to reconstruct the correct reading order:
$$\text{Sort Key: } (\text{char}[\text{''}top\text{''}], \text{char}[\text{''}x0\text{''}])$$
- Calculate the horizontal gap (`gap`) between consecutive characters to handle whitespace:
$$\text{If gap} > 1, \text{ add a space.}$$
- Concatenate characters into a complete text block, preserving semantic structure.

This approach proved to be highly effective for extracting clean, structured text, especially in scenarios where other methods struggled with non-standard PDF layouts.

Semantic Chunking using SciBERT

Semantic chunking was employed to process large research papers by breaking them into smaller, manageable segments, enabling efficient handling and analysis. Each document was tokenized into chunks of a fixed size $k = 512$ tokens, ensuring compatibility with SciBERT's input constraints while preserving semantic coherence.

The process can be mathematically represented as follows:

$$C_i = \{t_{(i-1)k+1}, t_{(i-1)k+2}, \ldots, t_{ik}\}, \quad \text{for } i = 1, 2, \ldots, \lceil n/k \rceil$$

where C_i represents the i-th chunk, n is the total number of tokens in the document, and t_j denotes the j-th token.

Knowledge Graph Construction and Analysis

We constructed knowledge graphs for each research paper to evaluate the coherence and connectedness of concepts. Graph nodes represented key concepts, while edges represented semantic relationships between them. The **LangChain LLMGraphTransformer with the Gemma2-9b-It model** was used to generate the graph structure.

Graph metrics were computed to analyze the coherence of each paper:

- **Density:**
$$\text{Density} = \frac{2E}{N(N-1)}$$

- **Degree Centrality:**
$$C_D(v) = \frac{\deg(v)}{N-1}$$

- **PageRank:**
$$PR(v) = \frac{1-d}{N} + d \sum_{u \in \text{In}(v)} \frac{PR(u)}{\deg^+(u)}$$
where d is the damping factor (set to 0.85), N is the number of nodes, and $\deg^+(u)$ is the out-degree of node u.

Graph Clustering and Node Ranking

Clusters within the graphs were identified using the Leiden algorithm, which optimizes modularity:

$$Q = \frac{1}{2E} \sum_{i,j} \left[A_{ij} - \frac{\deg(v_i)\deg(v_j)}{2E} \right] \delta(c_i, c_j)$$

where A_{ij} is the adjacency matrix, and $\delta(c_i, c_j)$ is 1 if nodes i and j are in the same cluster.

We ranked nodes using various algorithms:

- **Graph Density and PageRank:** Identified well-connected clusters and critical nodes.
- **Cluster Coefficient:**
$$C(v) = \frac{2T(v)}{\deg(v)(\deg(v)-1)}$$

Classification Threshold

A threshold of **0.0562** for the average top 5 PageRanks was set to classify papers:

$$\text{Classification} = \begin{cases} \text{Publishable,} & \text{if Avg Top 5 PageRanks} \geq 0.0562 \\ \text{Non-Publishable,} & \text{otherwise.} \end{cases}$$

Visualization and Analysis

Scatter plots and bar charts were generated to compare key metrics across datasets, providing further insights into the connectedness and coherence of publishable papers.

By integrating our novel text extraction method, zero-shot classification, and graph-based analysis, we developed a robust and explainable framework to tackle the challenges of evaluating research papers.

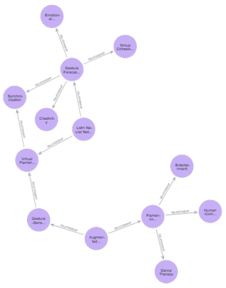


Figure 1: Knowledge graph illustrating the interconnected nodes of publishable research papers.

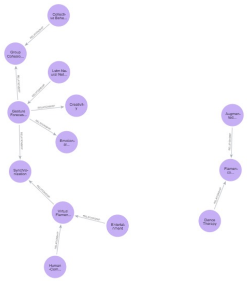


Figure 2: Knowledge graph illustrating the disconnected clusters of unpublishable research papers.

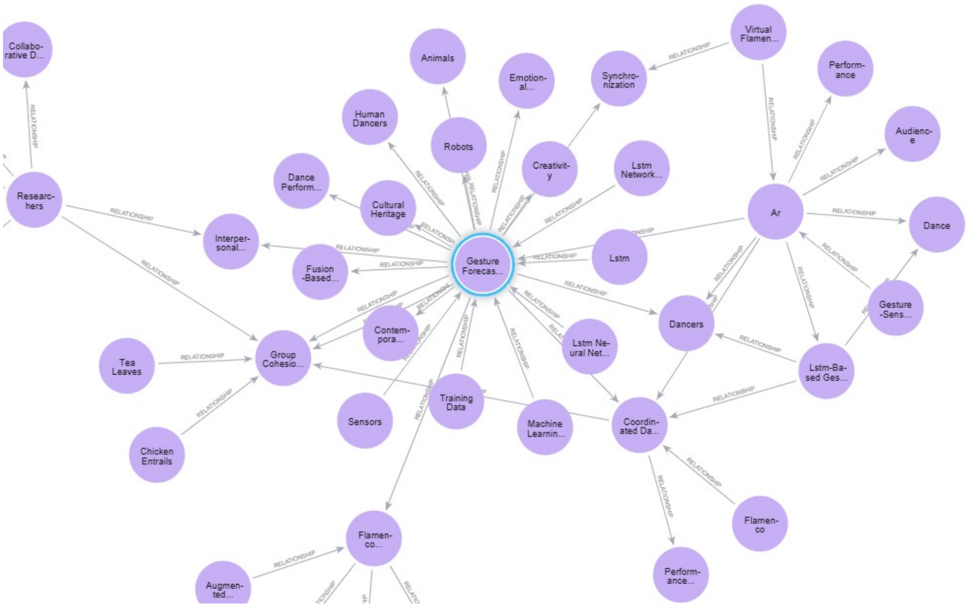


Figure 3: Comprehensive knowledge graph summarizing the relationships in publishable research papers using Neo4j. *Full image could not be captured due to Neo4j limitations.

Method for Task 2

Accesing Custom Dataset

This dataset is accessed through **Pathway Google Drive Connector** previously uploaded, which automates the ingestion of research paper PDFs from Google Drive, streamlining the pipeline for text extraction and processing.

Semantic Embedding using SciBERT

SciBERT is a variant of the BERT (Bidirectional Encoder Representations from Transformers) model, trained specifically on scientific text. The primary mathematical concepts underlying SciBERT involve:

Transformer Architecture

A transformer consists of an encoder-decoder architecture. SciBERT employs only the encoder stack, where each layer contains multi-head self-attention and feed-forward networks.

- **Self-Attention Mechanism**: The self-attention mechanism computes the attention scores for each word in the input sequence. Given input embeddings $X = \{x_1, x_2, \ldots, x_n\}$, the attention scores are calculated as:

  $$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

  where:

  - $Q = XW_Q$: Query matrix
  - $K = XW_K$: Key matrix
  - $V = XW_V$: Value matrix
  - d_k: Dimensionality of K

- **Multi-Head Attention**: Multiple self-attention heads enhance model capacity:

  $$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W_O \tag{2}$$

  where $\text{head}_i = \text{Attention}(QW_{Q_i}, KW_{K_i}, VW_{V_i})$.

- **Feed-Forward Network**: Each transformer layer applies a feed-forward network (FFN):

  $$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \tag{3}$$

The final output embeddings are contextualized representations of the input text, optimized for scientific domains in SciBERT.

Semantic Chunking using SciBERT

We again ustilise SciBERT's semantic chunking capiblities for this task.

Vector Similarity Search

The embeddings generated by SciBERT are stored in **Pathway VectorStore** for similarity-based retrieval, which facilitates efficient storage and retrieval of semantic embeddings, enabling rapid similarity searches for document classification and clustering. For a query embedding q and a set of document embeddings $E = \{e_1, e_2, \ldots, e_m\}$, similarity is computed using cosine similarity:

$$\text{Similarity}(q, e_i) = \frac{q \cdot e_i}{\|q\|\|e_i\|}, \quad \forall i \in \{1, 2, \ldots, m\} \tag{4}$$

The most similar embedding is retrieved as:

$$e^* = \arg\max_{e_i \in E} \text{Similarity}(q, e_i) \tag{5}$$

Sequence Classification

The classification of chunks into specific conferences is modeled as a supervised learning problem using a neural network. Given input embeddings X, the sequence classification model predicts the probability distribution over K classes (e.g., NeurIPS, CVPR):

$$\hat{y} = \text{softmax}(XW + b) \tag{6}$$

where:

- W: Weight matrix of the classifier
- b: Bias vector

The model is trained using cross-entropy loss:

$$\mathcal{L} = -\sum_{i=1}^{K} y_i \log(\hat{y}_i) \tag{7}$$

where y_i and $\hat{y}_i$ represent the true and predicted probabilities for class i.

Handling Unlabeled Data with Hybrid Techniques

Unlabeled data is addressed through a combination of vector similarity and supervised classification. For a new document embedding q, classification proceeds as follows:

1. Compute similarity scores with labeled embeddings in the vector store.
2. Assign a preliminary label based on the nearest neighbor.
3. Use the classification model to refine the prediction.

Structured Insights through Clustering

The embeddings are clustered to identify trends and relationships. Using k-means clustering, embeddings are grouped into k clusters:

$$\mu_j = \frac{1}{|C_j|} \sum_{x \in C_j} x, \quad j = 1, 2, \ldots, k \tag{8}$$

where C_j represents the set of embeddings in cluster j, and μ_j is the cluster centroid. The objective is to minimize intra-cluster variance:

$$\text{Objective} = \sum_{j=1}^{k} \sum_{x \in C_j} \|x - \mu_j\|^2 \tag{9}$$

Text Preprocessing

Abstracts are truncated to a maximum of $k = 512$ tokens:

$$\text{processed\_text} = \begin{cases} \text{abstract}[:k], & \text{if len(abstract)} > k, \\ \\ \text{abstract}, & \text{otherwise.} \end{cases}$$

Prompt Design for LLM

A structured prompt guides rationale generation:

```
Abstract: [preprocessed_text]
Conference: [predicted_conference]
Conference Info: [conference_theme]
Explain why this paper aligns with [predicted_conference].
```

Rationale Generation

Using **FLAN-T5**, the rationale y is generated as:

$$y = \arg\max_{y'} P(y'|x;\theta),$$

where x is the prompt and θ are the model parameters.

Visualization and Analysis

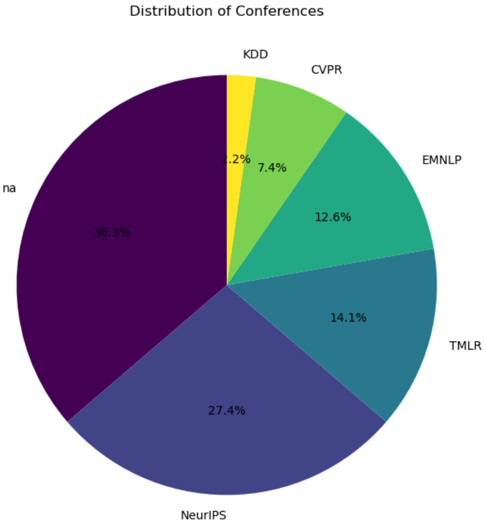
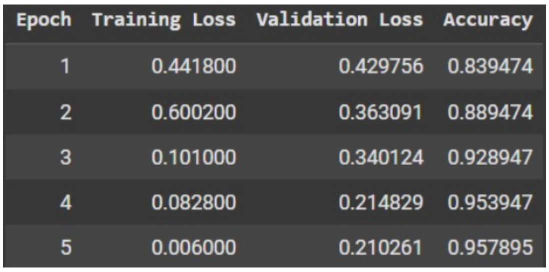
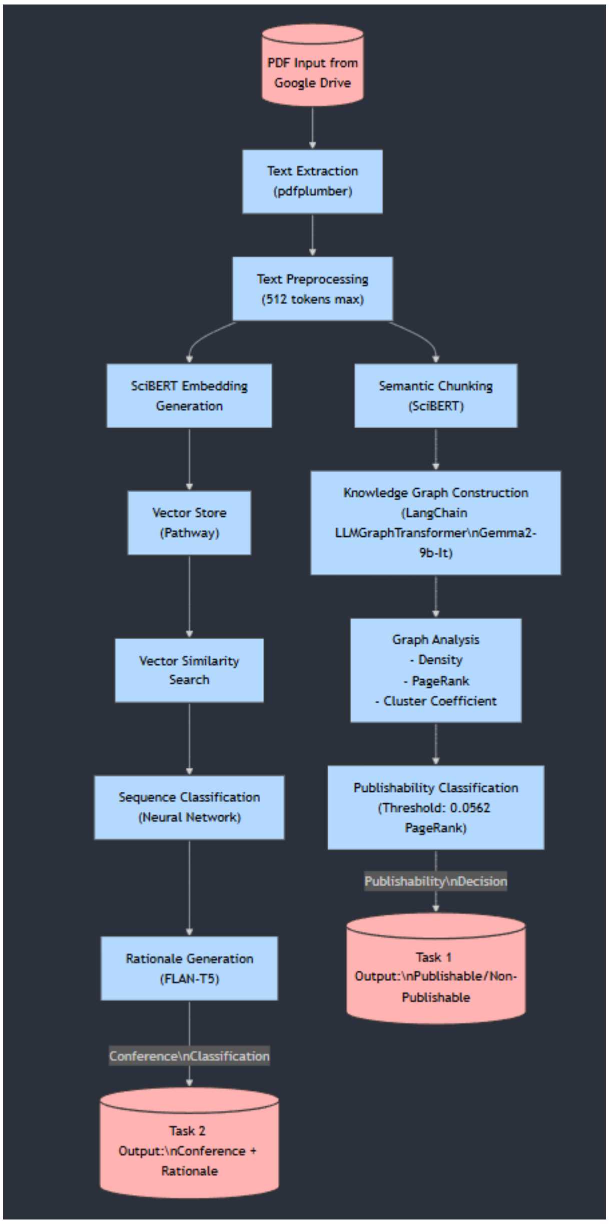


Figure 4: The distribution of Conference types of all the provided unlabelled papers.



Epoch	Training Loss	Validation Loss	Accuracy
1	0.441800	0.429756	0.839474
2	0.600200	0.363091	0.889474
3	0.101000	0.340124	0.928947
4	0.082800	0.214829	0.953947
5	0.006000	0.210261	0.957895

Figure 5: This is the table determined by running the model on the training data.


PDF Input from Google Drive
Text Extraction (pdfplumber)
Text Preprocessing (512 tokens max)
SciBERT Embedding Generation
Semantic Chunking (SciBERT)
Vector Store (Pathway)
Knowledge Graph Construction (LangChain LLMGraphTransformer\nGemma2-9b-It)
Vector Similarity Search
Graph Analysis
- Density
- PageRank
- Cluster Coefficient
Sequence Classification (Neural Network)
Publishability Classification (Threshold: 0.0562 PageRank)
Publishability\nDecision
Rationale Generation (FLAN-T5)
Task 1 Output:\nPublishable/Non-Publishable
Conference\nClassification
Task 2 Output:\nConference + Rationale

Overall System Architecture

Results

- **Task 1:** Achieved classification accuracy of **100%** accuracy on the labeled data provided, and **96.7%** on custom made dataset

- **Task 2:** Successfully identified key issues affecting publishability, getting an accuracy of **94.3%** on the test files we got after splitting the data into test and train (used for training model).

Experimentation

Custom Dataset Generation

To create a custom dataset, we curated a balanced set of research papers aimed at capturing both publishable and non-publishable categories. For the publishable papers, **30 top papers** from each target conference and an additional **30 randomly selected papers** corresponding to each conference were collected. This ensured diversity in quality and thematic representation across conferences.

For the non-publishable papers, we used **OpenReview, IEEE Xplore, Rejecta Mathematica** among many other resources to curate a diverse set of papers for this category.

This process resulted in a custom dataset of **600 research papers**, providing a robust foundation for classification and rationale generation.

Other Attempts

- **Baseline Model:** Implemented a basic classification model using TF-IDF & logistic regression which wasn't giving stable and reliable results. Also tried BERT+LSTM and ZeroGPT API approach for task 1.
- **Large Context Models:** Implemented Meta's LCM architecture, which had a very high inference time.
- **Other Approaches:** Explored advanced architectures such as transformers for improved performance.

Conclusion and Future Work

The framework achieved satisfactory results in classifying research papers based on publishability. Future work will focus on improving the model's generalizability and scalability to other domains of scientific papers.

Appendix

1. Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (pp. 3615-3620). Association for Computational Linguistics.

2. Wang, K., Shen, Z., Huang, C., Wu, C. H., Dong, Y., & Kanakia, A. (2020). Microsoft academic graph: When experts are not enough. *Quantitative Science Studies*, *1*(1), 396-413.

3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *In Advances in Neural Information Processing Systems*, *30*, 5998-6008.

4. Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., ... & Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144.*

5. Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *In Proceedings of EMNLP-IJCNLP 2019* (pp. 3982-3992).

6. Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, *7*(3), 535-547.

7. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., & Zhou, M. (2020). LayoutLM: Pre-training of text and layout for document image understanding. *In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 1192-1200).

8. Kang, D., Ammar, W., Dalvi, B., van Zuylen, M., Kohlmeier, S., Hovy, E., & Schwartz, R. (2018). A dataset of peer reviews (PeerRead): Collection insights and NLP applications. *In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1647-1661).

9. Chen, S., Ma, K., & Zheng, Y. (2019). Assessing scientific research papers with knowledge graphs. *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)* (pp. 5427-5437).

10. Goldberg, Y., Huang, Y., Tay, Y., & Wang, A. (2023). Large Concept Models: Towards Conceptual Abstraction in Large Language Models. *arXiv preprint arXiv:2308.05930.*

11. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., ... & Gebru, T. (2019). Model cards for model reporting. *In Proceedings of the Conference on Fairness Accountability and Transparency* (pp. 220-229).

12. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135-1144).

Citations

[1] https://www.scribbr.com/apa-examples/journal-article/

https://www.semanticscholar.org/paper/SciBERT:-A-Pretrained-Language-Model-for-Scient
156d217b0a911af97fa1b5a71dc909ccef7a8028

https://essaypro.com/blog/research-paper-format

https://kyleclo.com/assets/pdf/scibert-a-pretrained-language-model-for-scientific-tex
pdf

https://www.bibguru.com/blog/citation-styles-for-science/

https://ar5iv.labs.arxiv.org/html/1903.10676 https://www.enago.com/academy/write-research-paper-apa-format/

https://d-nb.info/1330363833/34

https://www.indeed.com/career-advice/career-development/how-to-cite-a-research-paper

https://pmc.ncbi.nlm.nih.gov/articles/PMC10148354/