# Can AI Tell More than the Available Abstracts?

Hong Zhou
Department of Mathematics and Computer Sciences
University of Saint Joseph
West Hartford, CT, USA
hzhou@usj.edu

Xiaoli Huan
Department of Computer Science
Troy University
Troy, Alabama, USA
xhuan@troy.edu

2024 International Conference on Sustainable Technology and Engineering (i-COSTE) | 979-8-3315-1733-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/i-COSTE63786.2024.11024734

***Abstract*—The word limitation of abstracts often prevents them from including many important scientific discoveries and technology applications that are fully explained in the corresponding full-text articles. This paper investigates whether some information embedded in the full-text articles can be deduced from the abstracts. Specifically, we explore the use of artificial intelligence (AI) to predict if certain experimental procedures or technologies, described in the Materials/Methods section of full-text articles, can be inferred based solely on the available abstracts. Using Pubmed Central (PMC) full-text articles as the data sample, we established a natural language processing (NLP) model and employed a pretrained distilled BERT model (DistilBERT). While the NLP model failed due to overfitting, the DistilBERT model achieved approximately 80% accuracy in predicting both xenograft procedures and lentiviral technology. Further work is underway to improve the DistilBERT model performance further.**

***Keywords—Artificial Intelligence, Machine Learning, PubMed, PMC, Abstract, NLP, Xenograft, Lentiviral.***

## I. INTRODUCTION

As the most accessed and complete collection of biomedical literature, today PubMed contains more than 37 million abstract citations from MEDLINE, life science journals, and online books. Additionally, PubMed accumulates thousands of abstracts on a daily average and the accumulation rates for certain subjects have increased exponentially [1], showcasing the huge amount of human intelligence invested in biomedical research. As the second largest component of PubMed, the PubMed Central (PMC) hosts over 10 million free full-text citations. Similar to the human genome, PubMed citations have become a highly valuable data source for bioinformatics research and biomedical text mining [2] [3] [4] [5] [6] [7] [8] [9], and there are a number of tools and platforms specifically developed to facilitate the knowledge discovery through PubMed [8] [10] [11] [12] [13] [14] [15] [16]. Over the past two decades, the technologies used in PubMed data mining have become increasingly powerful. Although keyword searching remains critical, machine learning technologies such as Natural Language Processing (NLP) and Convolutional Neural Network (CNN) have come into play. Along with the emergence of ChatGPT, there is PubMedGPT, an autoregressive large language model (LLM) pre-trained on PubMed abstracts and full-text papers [17] [18].

By far, most PubMed text mining or data mining focuses on bio-molecular interactions (such as gene-protein, protein-protein and protein-drug), gene pathways, and protein modifications, including phosphorylation and glycosylation, among others. Nevertheless, PubMed literature can be used for other purposes. For instance, making use of predefined positive, negative and neutral words, Vinkers et al. conducted a lexicographic analysis on PubMed abstracts and titles from 1 January 1974 to 31 December 2014 and found that abstracts were written in more and more positive tones [19].

Though more and more full-text articles are freely accessible through PMC or other open-access approaches, PubMed abstracts have always been the first choice for biomedical text mining [2, 3, 5-16]. This is probably due to several reasons. First, the majority of PubMed literature is still in the form of abstract citations. Second, the structural and content aspects of abstracts differ significantly from full-text articles [20]. Abstracts are usually written with succinct text sentences and only present the most valuable findings. However, full-text articles can include additional elements such as speculations, tables, figures, supplemental materials, and references. Third, some knowledge discoveries can be well achieved by mining PubMed abstracts alone. For example, research trend analysis. Fourth, the bundled PubMed abstracts have a better-defined XML format suitable for computational analysis than the bundled PMC full-text articles.

Due to the word limitation of abstracts, it is unsurprising that some scientific discoveries are only reported in full-text articles [21] [22]. For instance, Garten et al. reported that some important sentences showing pharmacogenomics associations were absent in abstracts but found in full-text articles [21]. Similarly, Blake found out that only a small portion of the scientific claims of full-text articles were reported in abstracts [23]. In 2018, a comprehensive comparison of text-mining in 15 million full-text articles versus their corresponding abstracts claimed that "text mining of full-text articles consistently outperforms using abstracts only" [24]. These findings motivated us to ask the question: can some information embedded in the full-text articles be deduced from the abstracts?

Since most PubMed abstracts do not have freely accessible full-text articles, being able to predict detailed information from the abstracts alone would significantly enhance their value for various purposes, including knowledge discovery and targeted marketing. For instance, in bio-product marketing, our analysis shows that only about 25% of PubMed citations include author email addresses, making it challenging for companies to directly reach out to researchers. However, by leveraging AI tools to analyze abstracts, companies can still identify potential customers.

AI models can predict whether a specific bio-product or technology is likely being used based on the information in the abstract. Even without access to the full article or the author's email, companies can compile a list of researchers who are likely to be interested in their products. They can then reach out to these researchers through other means, such as finding

their contact details in other databases, using professional networks, or targeting them with online advertising. This approach broadens the potential customer base and improves the effectiveness of marketing efforts, making PubMed abstracts a valuable resource for biotech companies.

## II. DATA AND MODEL PREPARATION

### A. Data Preparation

From the PMC FTP site, we downloaded the latest four oa_comm_xml (commercial use allowed) baseline files (PMC008xxxxxx to PMC011xxxxxx), which contain a total of 1,865,852 full-text articles in XML format. A typical PMC full-text article in XML format contains two major content nodes: <front> and <body>. The <front> node hosts author information and the abstract, while the <body> node is divided into multiple sections, such as Results, Discussion, References, and Materials/Methods. Using the Java packages org.w3c.dom and javax.xml.parsers, we parsed each XML file to extract the title and available abstract. To obtain the content of the Materials/Methods section, we first located the <body> node and then searched for a section title with no more than 21 characters that begin with one of the following words: Material, MATERIAL, Method, or METHOD. We manually examined 200 XML files to confirm that our approach was functioning correctly.

To investigate the xenograft procedure case, we further parsed the abstracts and Materials/Methods (MM) sections into individual words, then searched for one of the three keywords—XENOGRAFT, XENOGRAFTS, or XENOGRAFTED—without regard to case. We categorized the articles into four groups: (1) MM sections contain one of the keywords, but the abstracts do not (dataset 1); (2) both MM sections and abstracts contain one of the keywords (dataset 2); (3) neither MM sections nor abstracts contain any of the keywords (dataset 3); and (4) MM sections do not contain the keywords, but the abstracts do (dataset 4). The first two categories can be used as positive training data, either separately or combined, while the last two categories can be used as negative training data.

The same procedure was applied to the lentiviral technology, using the two keywords: lentiviral and lentivirus.

### B. Initial Preparation of an NLP Model

Initially, we tried to construct a natural language processing (NLP) model that was complex enough to meet the needs of our project. The model employed a word embedding layer, followed by five hidden layers and an output layer. The initial model is illustrated in Fig. 1.

```
model = nn.Sequential(
    nn.Embedding(vocab_size, embed_size),
    nn.AdaptiveAvgPool2d((1, embed_size)),
    nn.Flatten(), nn.Linear(embed_size, 64),
    nn.ReLU(),    nn.Linear(64, 256),
    nn.ReLU(),    nn.Linear(256, 1024),
    nn.ReLU(),    nn.Linear(1024, 256),
    nn.ReLU(),    nn.Linear(256, 64),
    nn.ReLU(),    nn.Linear(64, 2)
)
```

Fig. 1. The Python codes for the NLP model.

The computational platform used is Google Colaboratory, which provides unlimited CPU resources but limited GPU or TPU usage, free to the public. The model's performance was assessed using accuracy as the metric. The Fast.ai data loader was configured with the following conditions: 20% of the samples were allocated for validation, and the remaining 80% for training; the batch size was set to 64, sequence length (seq_length) to 400, and the random seed to 17.

To maintain balance between the label classes during model training and validation, we ensured that the model was fed an equal number of positive and negative training records. Since there were more negative training records, we kept all the positive records and randomly selected an equal number of negative records.

### C. The Distilled BERT Model

As one of the first large language models, BERT (Bidirectional Encoder Representations from Transformers) remains widely used because it is free and relatively easy to implement. However, to optimize BERT for the Google Colab environment, we employed DistilBERT, a smaller, faster, and lighter pretrained version of BERT [25]. DistilBERT is particularly well-suited for environments with limited computational resources, such as Google Colab, because it requires less memory and processing power while still maintaining much of the accuracy of the original BERT model. We froze all model weights except for the last classification layer, meaning that only the last classification layer of the DistilBERT model was trained with our data.

## III. RESULTS AND DISCUSSION

Since the label (target) of the employed AI models has only two classes (true or false), the evaluation of model performance should include two additional metrics besides the accuracy rate: sensitivity and specificity.

Sensitivity measures the model's ability to correctly predict positive classes. It is calculated as:

$$Sensitivity = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

Specificity measures the model's ability to correctly predict negative classes. It is calculated as:

$$Specifity = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}}$$

Another commonly used metric is positive predictive value (PPV), which measures how accurately the model predicts the positive class. It is calculated as:

$$PPV = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

A good machine learning model should perform well in both sensitivity and specificity. However, in certain cases, the priority might be a high positive predictive value.

### A. NLP Model Performance on Xenograft Procedure

The NLP model achieved a high accuracy rate in classifying whether the MM section describes the xenograft procedure. Table 1 summarizes the performance results.

Table 1. NLP model performances on xenograft procedure

| Pos Data | Neg Data | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|
| 1-0 | 0-0 | 0.9383 | 0.9580 | 0.9190 |
| 1-0 | 0-0+0-1 | 0.8956 | 0.9376 | 0.8550 |
| 1-0+1-1 | 0-0 | 0.9427 | 0.9286 | 0.9583 |
| 1-0+1-1 | 0-0+0-1 | 0.9029 | 0.9463 | 0.8592 |

In Table 1, "1-0" represents the dataset 1 which serves as the positive training data. "1-0+1-1" stands for the positive dataset that has dataset 1 and dataset 2 merged. "0-0" represents dataset 3 which serves as the negative training data. "0-0+0-1" stands for the negative dataset that has dataset 3 and dataset 4 merged.

As shown in Table 1, the model achieves an accuracy rate of 90% or higher in all four cases. However, in two instances, the specificity is significantly lower than the corresponding sensitivity. In practical scenarios, when an abstract is randomly selected for the model to make a prediction, it is likely to fall into the case represented by "1-0+1-1 vs. 0-0+0-1" (case 4). Even in this scenario, the model's validation accuracy rate remains slightly over 90%.

### B. NLP Model Performance on Lentiviral Technology

We applied the model to lentiviral technology, which is a popular gene transfer technology. The two keywords used in data preparation are LENTIVIRAL and LENTIVIRUS. As expected, the NLP model achieved well in predicting whether the MM section has either keyword present. The data is summarized in Table 2. Overall, the model performed slightly better on lentiviral technology than on xenograft procedure.

We applied the model to lentiviral technology, a widely used method for gene transfer. The two keywords used in data preparation were "LENTIVIRAL" and "LENTIVIRUS." As expected, the NLP model performed well in predicting whether the MM section contained either keyword. The results are summarized in Table 2. Overall, the model performed slightly better on lentiviral technology than on the xenograft procedure.

Table 2. NLP model performances on lentiviral technology

| Pos Data | Neg Data | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|
| 1-0 | 0-0 | 0.9809 | 0.9879 | 0.9739 |
| 1-0 | 0-0+0-1 | 0.9188 | 0.9432 | 0.8936 |
| 1-0+1-1 | 0-0 | 0.9270 | 0.9335 | 0.9206 |
| 1-0+1-1 | 0-0+0-1 | 0.9238 | 0.9405 | 0.9071 |

### C. Unexpected Discovery

At this point, the model seemed promising. To further evaluate its effectiveness, we tested the NLP model on an independent xenograft testing dataset. However, the model's performance on this new dataset was unexpectedly poor, as shown in Table 3. The results were no better than what would be expected from a random model.

Table 3. The NLP Model failed with the xenograft test dataset

| Pos Data | Neg Data | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|
| 1-0+1-1 | 0-0+0-1 | 0.594 | 0.681 | 0.531 |

Certainly, the NLP model encountered an overfitting problem. We applied several techniques to address this, including adding dropout layers and reducing model complexity. However, none of these approaches were able to overcome the overfitting issue. We suspect that the text data loader from Fast.ai may have a serious bug. During the model training and validation process, the data loader repeatedly allocated 20% of the training data as the validation dataset, which likely contaminated the validation set. We are currently conducting further research to investigate this issue.

### D. DistilBERT Performance

While we continued investigating the overfitting issue with the NLP model, we began to employ the DistilBERT model. Although the DistilBERT model did not achieve a high accuracy rate, it performed well on the independent testing dataset. The results are summarized in Table 4.

Table 4. The DistilBERT model performances

| | Pos Data | Neg Data | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|---|
| X-1 | 1-0+1-1 | 0-0+0-1 | 0.862 | 0.944 | 0.780 |
| X-2 | 1-0+1-1 | 0-0+0-1 | 0.812 | 0.961 | 0.714 |
| Lenti-viral | 1-0+1-1 | 0-0+0-1 | 0.776 | 0.818 | 0735 |

In Table 4, "X-1" represents the validation xenograft dataset, while "X-2" stands for the independent testing xenograft dataset. The significantly higher sensitivity values suggest that the model is optimizing for positive data, indicating another type of overfitting—specifically, overfitting to positive data.

In some cases, overfitting occurs when the model is too complex for the data. We then explored whether adding the article title to the abstract could increase data complexity and, in turn, improve the model's performance. As expected, adding the article title to the abstract did not significantly impact model performance. However, when we tested the model's ability to make predictions based on article titles alone, an unexpected discovery emerged.

It was surprising to find that the model could achieve nearly the same accuracy rates using only the article titles. Based on our understanding, this should not happen, as article titles typically do not contain enough information for an AI model to make accurate predictions on a specific topic. This likely indicates that the DistilBERT model also has a serious overfitting issue. Further experiments revealed that this overfitting problem is present in the NLP model as well. Table 5 presents the results obtained from experiments with the positive dataset 1-0+1-1 and the negative dataset 0-0+0-1.

Table 5. Model performances with titles only

| | Model | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|
| Xenograft | NLP | 0.876 | 0.920 | 0.876 |
| | BERT | 0.783 | 0.921 | 0.682 |
| Lentiviral | NLP | 0.913 | 0.929 | 0.900 |
| | BERT | 0.848 | 0.827 | 0.870 |

Ideally, when there is not enough information for the AI model to make a prediction, its performance should be close to a 50% accuracy rate. Occasionally, some article titles may contain relevant information indicating the use of either the xenograft procedure or lentiviral technology. In these cases, the model might make a correct prediction. However, even considering these situations, we suspect that the model's performance should not exceed a 60% accuracy rate. Therefore, the data in Table 5 suggests that neither the DistilBERT nor the NLP models should be used to make predictions until their overfitting issues are significantly reduced.

## IV. Conclusion

Our current NLP model and the DistilBERT model have produced some exciting results, but they also raise serious concerns. In future work, we plan to address the overfitting issues observed in both the NLP and DistilBERT models by exploring several strategies. First, we will refine our model selection process by experimenting with simpler models and alternative architectures that may better balance complexity and performance, potentially mitigating overfitting. Specifically, we will consider experimenting with simpler models or reducing the number of layers or parameters in DistilBERT to enhance its suitability for our dataset. Additionally, we intend to implement advanced regularization techniques, such as L2 regularization and dropout with varying rates, to improve the model's generalization capabilities. Furthermore, we plan to evaluate the models on a broader range of biomedical domains to assess their generalizability beyond the current dataset. We will also refine our data processing pipeline by incorporating cross-validation and using alternative data loading libraries to prevent validation set contamination. Lastly, we will investigate the potential of ensemble learning approaches, combining predictions from multiple models to reduce the impact of overfitting and improve overall performance. These efforts aim to develop more robust and reliable AI models for predicting detailed information from PubMed abstracts.

## References


[1] W. W. M. Fleuren and W. Alkema, "Application of text mining in the biomedical domain," *Methods,* vol. 74, pp. 97-106, 2015.

[2] H. Chen and B. M. Sharp, "Content-rich biological network constructed by mining PubMed abstracts," *BMC Bioinformatics,* vol. 5, p. 147, 2004.

[3] S. Zaremba, M. Ramos-Santacruz, T. Hampt, P. Shetty, J. Fedorko, J. Whitmore, J. M. Greene, N. T. Perna, J. D. Glasner, G. Plunkett, M. Shak and D. Pot, "Text-mining of PubMed abstracts by natural language processing to create a public knowledge base on molecular mechanisms of bacterial enteropathogens," *BMC Bioinformatics,* vol. 10, p. 177, 2009.

[4] M.-T. Pandi, P. . J. v. d. Spek, M. Koromina and G. P. Patrinos, "A Novel Text-Mining Approach for Retrieving Pharmacogenomics Associations From the Literature," *Frontiers in Pharmacology,* vol. 11, 2020.

[5] S. Anand, O. R. Iyyappan, S. Manoharan, D. Anand, M. A. Jose and R. R. Shanker, "Text Mining Protocol to Retrieve Significant Drug-Gene Interactions from PubMed Abstracts," *Methods in Molecular Biology,* vol. 2496, pp. 17-39, 2022.

[6] Y. Luo, G. Riedlinger and P. Szolovits, "Text Mining in Cancer Gene and Pathway Prioritization," *Cancer Informatics,* vol. 13, no. S1, p. 69–79, 2014.

[7] K. Arumugam, M. Sellappan, D. Anand, S. Anand and S. V. Radhakrishnan, "A Text Mining and Machine Learning Protocol for Extracting Posttranslational Modifications of Proteins from PubMed: A Special Focus on Glycosylation, Acetylation, Methylation, Hydroxylation, and Ubiquitination," *Methods in Molecular Biology,* vol. 2496, pp. 179-202, 2022.

[8] A. Barbosa-Silva, J.-F. Fontaine, E. R. Donnard, F. Stussi, J. M. Ortega and M. A. Andrade-Navarro, "PESCADOR, a web-based tool to assist textmining of biointeractions extracted from PubMed queries," *BMC Bioinformatics,* vol. 12, p. 435, 2011.

[9] P. Monsarrat and J.-N. Vergnes, "Data mining of effect sizes from PubMed abstracts: a cross-study conceptual replication," *Bioinformatics,* vol. 34, no. 15, p. 2698–2700, 2018.

[10] J. Rani, A. B. R. Shah and S. Ramachandr, "pubmed.mineR: An R package with text-mining algorithms to analyse PubMed abstracts," *Journal of Biosciences,* vol. 40, pp. 671-682, 2015.

[11] C. Simon, K. Davidsen, C. Hansen, E. Seymour, M. B. Barnkob and L. R. Olsen, "BioReader: a text mining tool for performing classification of biomedical literature," *BMC Bioinformatics,* vol. 19, no. Suppl 13, p. 57, 2019.

[12] D. Fantini, "easyPubmed," 2019. [Online]. Available: https://www.data-pulse.com/dev_site/easypubmed/. [Accessed 29 07 2024].

[13] D. Winter, S. Chamberlai and H. Guangchun, "rentrez," 10 11 2020. [Online]. Available: https://docs.ropensci.org/rentrez/. [Accessed 29 07 2024].

[14] C.-H. Wei, B. R. Harris, D. Li, T. Z. Berardini, E. Huala, H.-Y. Kao and Z. Lu, "Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts," *Database,* p. bas041, 2012.

[15] M. H. Gunturkun, E. Flashner, T. Wang, M. K. Mulligan, R. W. Williams, P. Prins and H. Chen, "RatsPub: a webservice aided by deep learning to mine

PubMed for addiction-related genes," *bioRxiv Preprint,* p. 297358, 2020.

[16] N. R. Smalheiser, D. P. Fragnito and E. E. Tirk, "Anne O'Tate: Value-added PubMed search engine for analysis and text mining," *PLoS ONE,* vol. 16, no. 3, p. e0248335, 2021.

[17] A. Venigalla, J. Frankle and M. Carbi, "BioMedLM: a Domain-Specific Large Language Model for Biomedical Text," Mosaicml.com, 15 12 2022. [Online]. Available: https://www.mosaicml.com/blog/introducing-pubmed-gpt. [Accessed 29 07 2024].

[18] E. Bolton, D. Hall,, M. Yasunaga, T. Lee, C. Manning and P. Liang, "Stanford CRFM Introduces PubMedGPT 2.7B," 15 12 2022. [Online]. Available: https://hai.stanford.edu/news/stanford-crfm-introduces-pubmedgpt-27b. [Accessed 29 07 2024].

[19] C. H. Vinkers, J. K. Tijdink and W. M. Otte, "Use of positive and negative words in scientific PubMed abstracts between 1974 and 2014: retrospective analysis," *BMJ,* vol. 351, p. h6467, 2015.

[20] B. K. Cohen, H. L. Johnson, K. Verspoor, C. Roeder and L. E. Hunter, "The structural and content aspects of abstracts versus bodies of full text journal articles are different," *BMC Bioinformatics,* vol. 11, p. 492, 2010.

[21] Y. Garten and R. B. Altman, "Pharmspresso: a text mining tool for extraction of pharmacogenomic concepts and relationships from full text," *BMC Bioinformatics,* vol. 10 , no. Suppl 2, p. S6, 2009.

[22] J. Samuel, X. Yuan, X. Yuan and B. Walton, "Mining online full-text literature for novel protein interaction discovery," in *2010 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*, Hongkong, China, 2010.

[23] C. Blake, "Beyond genes, proteins, and abstracts: Identifying scientific claims from full-text biomedical articles," *Journal of Biomedical Informatics,* vol. 43, no. 2, pp. 173-189, 2010.

[24] D. Westergaard, H.-H. Stñrfeldt, C. Tønsberg, L. J. Jensen and S. Brunak, "A comprehensive and quantitative comparison of text-mining in 15 million fulltext articles versus their corresponding abstracts," *PLoS Comput Biology,* vol. 14, no. 2, p. e1005962, 2018.

[25] V. Sanh, L. Debut, J. Chaumond and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv,* p. arXiv:1910.01108v4, 2020.