MasterAI: Interdisciplinary Discovery of Nanomaterials Based on Convolutional Neural Networks

Anonymous Author(s)
Affiliation
Address
email

Abstract

The material science literature contains up-to-date and comprehensive scientific knowledge of materials. However, their content is unstructured and diverse, resulting in a significant gap in providing sufficient information for material design and synthesis. To this end, we used natural language processing (NLP) and computer vision (CV) techniques based on convolutional neural networks (CNN) to discover valuable experimental-based information about nanomaterials information and synthesis methods in energy-material-related publications. Our first system, TextMaster, extracts opinions from textual content and classifies them into challenges and opportunities, achieving 94% and 92% accuracy, respectively. Our second system, GraphMaster, realizes data extraction of tables and figures from publications with 98.3% classification accuracy and 4.3% data extraction mean square error on average. We prove that these systems could assess the suitability of materials for a certain application with detailed references by case analysis and evaluation of real nanomaterial synthesis and texture insights. This work offers a fresh perspective on mining and unifying knowledge from scientific literature, providing a wide swatch to accelerate nanomaterial research through CNN.

1 Introduction

The scientific literature is meant to communicate recent advances and keep researchers informed so they can continue to make breakthroughs and advance their fields. Since innovative materials have been regarded as one of the promising roles leading the fourth industrial revolution, the material science field has witnessed a drastic volume increase in publications. Therefore, comprehensive selection and reading of the material science scientific papers by hand are not realistic. A potential solution to this issue is to automatize the task using artificial intelligence (AI) technologies. However, the information in scientific literature is multimodal (text, figures, tables, etc.), which makes identifying and utilizing the latent knowledge in publications even more challenging.

In this paper, we resort to CNN to discover and summarize knowledge related to nanomaterials from scientific literature, with a novel perspective of combining the NLP and CV techniques to assimilate and integrate the multimodal literature knowledge. Firstly, we applied the classic NLP task opinion mining to the body content of energy-related publications. The system includes four modules: (i) text preparation, (ii) opinion extraction, (iii) opinion classification and (iv) opinion mining for information analysis, where CNN with attention mechanism is applied to modules (ii), (iii) and (iv). This system can assess the suitability of nanomaterials for an application from various aspects, such

Submitted to NeurIPS 2021 AI for Science Workshop.

as synthesis performance and natural properties. Secondly, we develop a system to assist in compiling a systematic and highly refined performance review of b-Si solar cells. The system includes (i) Content Retrieval, (ii)Graph Classification, (iii) data extraction. NLP techniques are applied to help screen out irrelevant articles in module (i). With modules (ii) using a CNN-based neural network, ResNet [1, 2], identifying the different elements, i.e., texts, tables, figures and images, then different NLP and CV models are further applied to extract corresponding data in module (iii). This system indicates that hierarchical textures and inverted-pyramidal b-Si nano-textures and micro-textures are promising as next-generation texturing techniques for Si solar cell applications.

2 Related Works

We identify related works for this paper in the area of NLP and CV methods. NLP methods are categorized into two domains, namely: NLP methods related to scientific discovery and opinion mining techniques. CV methods are mainly related to image recognition.

NLP methods related to scientific discovery NLP researchers have implemented different tasks on scientific text, such as pre-trained language model for scientific text [3], text classification [4], text summarization [5], named entity recognition (NER) [6] and knowledge graph [7]. In materials science, significant effort has been devoted to extracting material mentions [8, 9] or synthesis recipes [10, 11, 12]. These material science applications are based on NER, linking material names to their co-occurring entities, which help answer large-scale questions, such as "What are the most frequent methods for TiO_2 synthesis?" Recent attempts to predict suitable materials for certain applications have been made by word embeddings [13] or graph-based machine learning [14], which efficiently encodes materials science data into high-dimensional numeric representations.

Opinion mining techniques Opinion mining is an NLP task that analyzes people's opinions, sentiments, evaluations, attitudes, and emotions from written language [15]. It has widespread applications, such as categorizing product reviews to support consumers' decisions [16], analyzing social media texts to predict the stock market [17] or election results [18], and fostering effective interactions in smart cities [19]. This technique has been applied to scientific literature but mostly focuses on article reviews [20] or citations [21] instead of the article's body content. We creatively applied sentence-level opinion mining to the body content of energy material-related publications. Existing sentence-level opinion mining approaches can be divided into lexicon-based and corpus-based approaches [22]. Lexicon-based approaches are usually unsupervised, based on the measurement of word sentiment orientation using WordNet or machine learning models [23, 24]. Corpus-based approaches are usually supervised, and dominated by feature-based machine learning methods and deep learning methods. In comparison, deep learning methods show better performance on most of the benchmark datasets.

CV methods related to scientific graph classification The first graph type classification work was introduced by Zhou and Tan[25] with a modified probabilistic through transform methods. They achieved 84% accuracy of graph type classification in 1190 total images. After that, graph classification research began to attract attention. Scientists have applied different methods for graph classification problems, such as hidden Markov models, SVMs and CNNs. We summarize some of the existing graph classifications works in Table 1.

Work	Methods	Type of Labels	Dataset	Performance
[25]	Customized Algorithm	3	1190	84%
[26]	SVM	10	2601	75.00% - 86.00%
[27]	CNN	5	11174	99%
[28]	CNN	28	33000	88.96% - 92.90%

Table 1: Summary of the related graph type classification work

Techniques of data extraction from chart Previous research on data extraction from charts can be divided into two types, interactive and full automated approaches. Chartsense [2] was a semi-automated data extraction tool that provides the user with an interface to mark graph elements such as the x-axis and the origin of coordinates. However, interactive extraction was time-consuming and unsuitable for large data collection. Automated graph data extraction such as Revision [26], Chartreader [6], and Chartdecoder [27] used different Optical character recognition (OCR) engines and CNNs to realize Bar Chart data recovery. However, for material science publications, a lot of important material information was presented with line charts, especially for spectroscopy spectra. In this work, we demonstrated how to use CNN to fully automatically extract the data from the material science line charts.

3 Methods

Here we introduce NLP and CV methods that were used to analyze energy-related publications. Our python code (includes TextMaster, GraphMaster and SciCrawler) is available at three repositories in https://github.com/for-submission.

3.1 TextMaster

The schematic diagram of TextMaster system is shown in Figure 1.

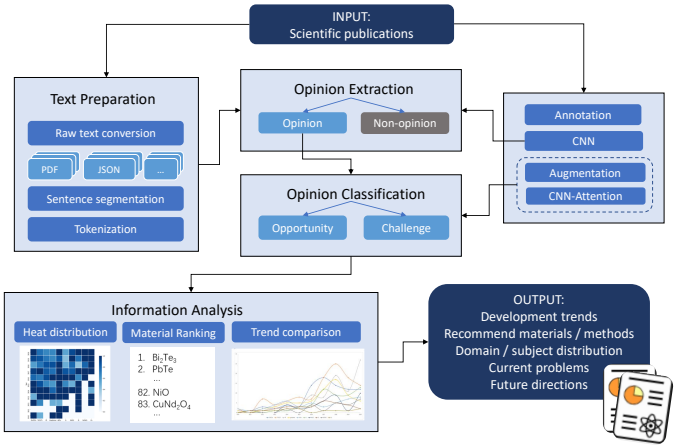


Figure 1: The schematic diagram of TextMaster

3.1.1 Text preparation

We used the Web of Science client to collect DOI lists for various nanomaterial scientific articles. Then we customized a web-scraper, SciCrawler, to automatically download a broad selection of nano materials-related papers published after 2000 from publishers' API or websites in PDF format. The publications on a certain topic were collected by searching keywords from databases such as Web of Science. We prepared a mixed-topic dataset formed by three datasets with end-of-life management (EoL), perovskite solar cells, and atomic layer deposition (ALD) themes, covering management, material and synthesis. The EoL dataset was formed by 77 full papers. Perovskite and ALD datasets were formed by 34,752 abstracts and 21,276 abstracts respectively, containing all meta fields provided by Web of Science. Plain texts were extracted, segmented into sentences, and tokenized. The sentences in EoL dataset were annotated as "opinion" and "non-opinion", and "opinion" sentences were further annotated as "opportunities" and "challenges".

3.1.2 Opinion extraction

The processed sentences were classified as opinions and non-opinions in this module. In the pilot study, we compared the performance of some typical lexicon-based methods and deep learning models on the EoL data set (see Appendix A). It can be seen that the CNN model exceeds other methods on distinguishing opinions from non-opinions. Due to the imbalance of opinion and non-opinion categories, we also used the Synthetic Minority Oversampling Technique (SMOTE) [29] to generate data for opinion category. To increase data size, the CNN-SMOTE (CNN with SMOTE technique) trained on the EoL dataset was used as raw models to predict the pseudo labels of sentences in abstracts of ALD- and perovskite-related publications. Each sentence was predicted as opinion or non-opinion. Then, annotators were invited to proofread the predictions. We trained the final CNN model on about 22 thousand samples, achieving an accuracy of 94%.

3.1.3 Opinion classification

The input opinions were classified as opportunities and challenges in this module. Compared with CNN, a CNN model with attention mechanism [30] had a better performance in the pilot study (see Appendix A). Similar to the last module, we used a raw model to increase data size and trained the final CNN-Attention model on about 9 thousand mixed-topic samples. Due to the imbalance of opportunity and challenge categories, we also used the SMOTE to generate data for the challenge category, which improved the performance of the CNN-Attention model from 89% to 92%.

3.1.4 Information analysis

With extracted and classified opinions, this module used embedded material names or elements for information analysis. For each opportunity or challenge sentence, ChemDataExtractor [8] was used to identify the material name in it. Based on the sentiments of opinions and year of publication, we can calculate each material's percentage of opportunities in opinions each year. This method quantified the sentiment towards elements as sentiment scores across time. Using opinions, we implemented a case study of ALD elements and a prediction of thermoelectric materials (see the Results and Discussion Section).

3.2 GraphMaster

Figure 2 presents the graph data recovery pipeline, which mainly consists of three components, i) content retrieval ii) graph classification and iii) data extraction.

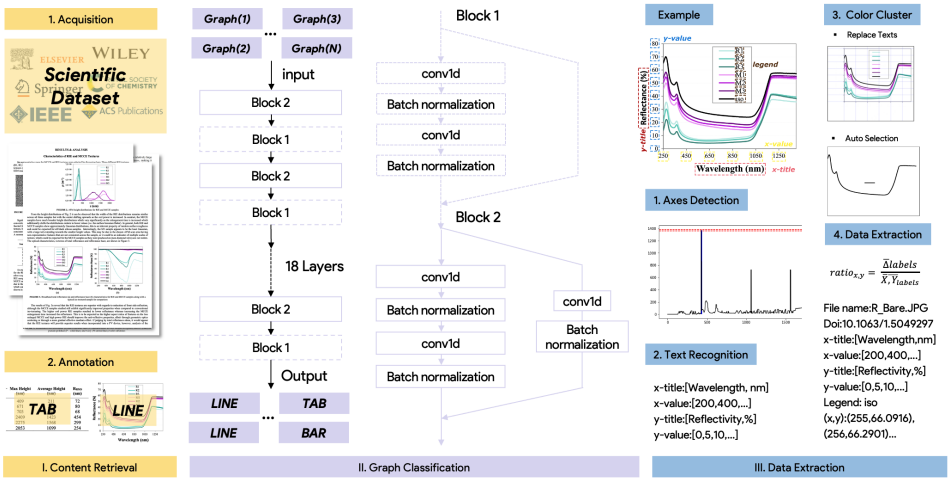


Figure 2: Schematic diagram of graph classification and data extraction pipeline

In this paper, we only proposed the data extraction solution for the line graph, since most of the material's characterization figures are line graphs including Raman spectroscopy, X-ray diffraction, external quantum efficiency graphs, etc. There are some existing solutions for data recovery of other graphs, including bar charts and tables [27, 8, 31].

3.2.1 Content Retrival

The paper collection process is similar to the TextMaster (See 3.1.1). PDFFigure 2 [32, 33] , an existing algorithm, was used to extract figures from scientific papers. In material science publications, one figure commonly includes several sub-figures. However, it is hard for PDFFigure 2.0 to capture each subfigure. We decomposed the figure into subfigures through valid axis-aligned splitting recognition[34]. Finally, we manually assigned them category labels, including line charts, bar charts, and tables. The figure and metadata like the graphs, journal, and title were stored in the Neo4j dataset for convenient access (https://neo4j.com/). After the selection of samples related to the analyzed materials, we used 6,000 figures from 3,600 scientific papers.

3.2.2 Graph Classification

We used a 20-layer ResNet [35] with 9 bottleneck blocks to do binary classification on the collected graphs. The purpose of the classification was to extract the specific type of charts that were ready to be fed to the information detection step in the pipeline to extract the data of the plot. In this paper, the extraction step focused on the line charts. However, we tried to make every module as general and extensible as possible. Thus, the classification module is not only able to classify line charts, but also other three types of charts and images that were prevalent in academic literature. The four types were line charts, images of tables, bar charts and images of pseudo code of algorithms. The model was trained on about 20 thousand charts and images with labels using the cross entropy loss. We achieved very high accuracy on the classification task in all four categories, 97.9% accuracy on line charts, 97.33% accuracy on table images, 99.19% on bar charts and 98.78% on algorithm images. All the accuracies were obtained on a set-aside test dataset. This high accuracy on the test dataset suggests that our graph classification module has a very strong generalization performance. This solidly indicates that it would not only perform well on our own dataset but will also perform well on any unseen dataset of the same types, which means that our model would be practical and useful in real-world using scenarios.

3.2.3 Data Extraction

Axes Detection Inspired by chartreader [36], we convert the line charts to the grey-scale image to identify the location of the x-axis and y-axis, by replacing all pixels with luminance greater than 200 with value '1' (black) and all other pixels with value '0' (white). We experimented with different values for grey-scale thresholds and 200 was found to be best for grey-scale image conversion. For x-axis identification, we scanned the matrix horizontally the recorded the black pixels' continuity within surrounding rows. Then we selected all the continuous black pixels with less than ten white pixels in the row. Finally, we chose the column with the lowest fluctuation by the highest continuously of the adjacent rows. The reason was that that axis commonly consisted of 2-3 pixels in publication for clear reading. With a similar process, we can also detect the y-axis column.

Text Detection AWS-Rekognition DetectText API was used to detect text within an image. DetectText also provided rectangular bounding boxes of the detected text. We selected the detected text with the highest confidence score. A high threshold of the confidence score improved precision. However, it would also reduce the recall rate. Therefore, in order to mitigate recall rate reduction, we added a dual layer for the second pass to release high-quality recognition. To this end, we set the confidence score threshold to 82 empirically. The bonding boxed centre coordinates and height and width were stored in a matrix, $\{t_x, t_y, t_w, t_h\}$, and they were replaced with white colour.

Text Extraction A line chart contains a lot of different texts, such as figure titles, axes titles, labels, and legends. The five text types that we classified were listed below: *X-axis value* (blue): x-axis text is the bounding boxes detected below the x-axis, and their centroids are parallel to the x-axis.

X-axis title (red): x-axis text is the bounding boxes detected below the x-axis title, which is near the bottom of the figure in the relative coordination.

Y-axis value (yellow): y-axis text is the bounding boxes detected left to the y-axis, and their centroids are parallel to the y-axis.

Y-axis title (pink): y-axis text is the bounding boxes detected left to the y-axis, and their centroids are parrel to the y-axis.

Legend (brown): legends are the bounding boxes detected in other locations.

Colour Cluster Since the line charts usually contained many lines, the authors used different features to distinguish, such as colours and dash lines. We applied the DBSCAN [37] to classify different figures based on the RGB value of each colour. We examined the classification results on a set of randomly selected groups, and 16 groups had the best colour classification results. We selected the figures with the most pixels, and the threshold depended on the number of legends bounding boxes detected. (See Appendix C for an example diagram detailed process)

Data Extraction Based on the clustering results, we extracted data from each line consecutively. Specifically, for each line, we scanned from left to right horizontally. At each iteration (at each x value), we examined the vertical column of pixels. There was always one coloured pixel that belongs to the line while all other pixels belong to the white background. Besides, we can easily locate the position of that coloured pixel in that column. By calculating the ratio between the coloured pixel's height and the total height of the column, we can recover the y value for this specific x value. By repeating the same process for all x values in the scan and for each line in the plot, we can recover all the x-y value pairs of the plot. (See Appendix D for an example detailed process)

4 Results and Discussion

Here we display the results of two systems on different publication datasets and discuss how they can render the latent knowledge and accelerate the energy material research.

4.1 CNN assisted trends analysis and insights discovery for nanomaterial synthesis

In this paper, we showed the preliminary findings of ALD, which was a popular nano-film fabrication technique. The ALD dataset had over 2,100 opinions with material names, in which nearly 70% of their sentiments are opportunities. To avoid data fragmentation, we clustered opinions to corresponding periodic table elements of materials. We visualized scientific categories of the top 10 elements used in ALD in Figure 3. It can be summarized that popular chemical elements used in ALD are widely employed in material science, chemistry, nanoscience and physics, and to a lesser extent, energy and fuels. In addition, categories like crystallography and pharmacy are unique to the ALD of specific elements, namely, titanium (Ti). The gap between disciplines may hinder the extraction and utilization of shared knowledge about materials. This figure implies the great potential to transfer knowledge in opinions from one field to another.

We visualized the frequency of popular elements used by opinions from 2014 to 2022 in Figure 4(a). To quantify the sentiment towards elements, we calculated each element's percentage of opportunities in opinions each year based on the sentiments and year of publication. As shown in Figure 4(b), each cell represents the average sentiment value of the element fabricated with ALD that year. We observed that there is often a drop in frequency after the year when the element experienced a rather negative sentiment and vice versa. For example, sentiment scores of Zn (dark blue), Ti (orange), Mo (pink), and Si (grey) had low levels in 2018, and their frequency fell in the following years. In contrast, the sentiment score of Mo (pink) in 2016, Ni (yellow) in 2017 and Al (purple) in 2019 were

all high levels, and their frequency went up after experiencing positive sentiments. Although this relationship cannot be guaranteed, it helps us to predict the future trends of elements: Al, Si and Ni go rather cold; Pt, C and Cu emerge more; Zn, Ti, Li and Mo have relative stable attention. We also visualized opinion mining results of elements or materials in other domains such as perovskite and thermoelectric (see Appendix B).

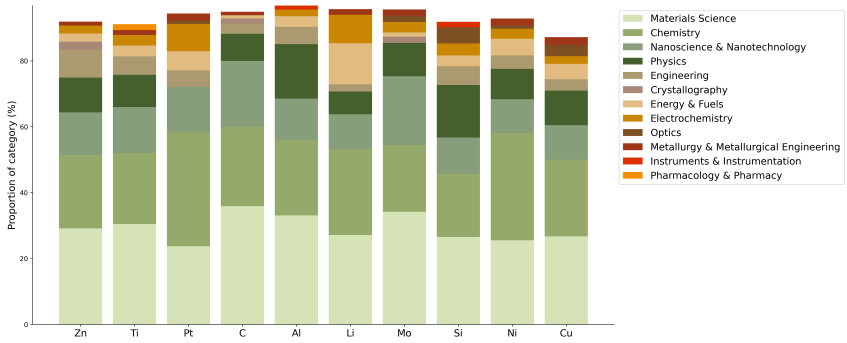
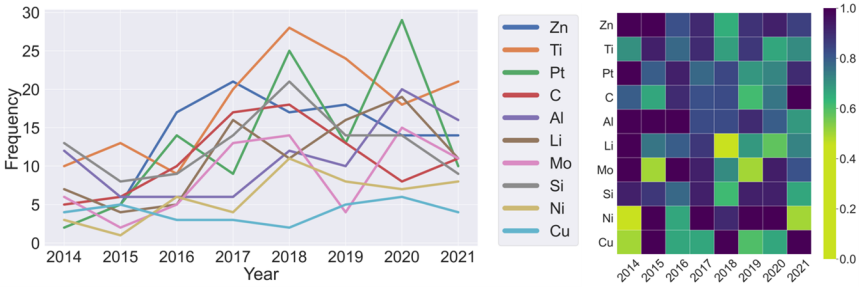


Figure 3: Proportion of publication category of the popular ALD elements



(a) (b)

Figure 4: (a) Frequency and (b) sentiment heatmap of popular ALD elements.

Case Study on Li We examined the scientific facts in extracted opinions of lithium, an element with marked changes in sentiment score. As we can see in Figure 4(b), lithium experienced a prominent negative sentiment in 2018 due to intrinsic properties, including thermal instability [38, 39], shuttling effects [40], and electrochemical instability [41]. Fortunately, several improvement measures of these drawbacks were found in opinions of the following years. In 2019, Li [42] provided the possible solution for the $LiMn_1 \cdot 5\,Ni_0 \cdot 4\,O_4$ (LMNO) nanomaterial cycle time through selectively controlled methods, integrating ALD with annealing at the nano level and process controlling and optimization at the macro level. After that, Lou et al. [43] showed that the addition of Ti to Li-compounds can improve its cycling stability which can easily be achieved using ALD. These drawbacks and improvements were from diversified categories, such as energy fuels, nanoscience, and metallurgy. This case study shows that our system can further provide the reasons behind rising or falling sentiments.

4.2 CNN assisted investigation of potential nanomaterial textures for silicon solar cell

The wide range of complex nanotextured surface morphologies made a comprehensive and systematic empirical investigation impossible. An unambiguous performance review for reported empirical results was necessary to determine the promising materials' nanotextures. For generating textual and graphical information for nanotextures, we implemented GraphMaster to help the data recovery from

material nanotextures characterization results, which were usually present as figures, such as Raman spectroscopy spectra, and wavelength-dependent External Quantum Efficiency(EQE).

Case Study of black Si Nanotexture for solar cells we used NLP, CV and ML to generate a highly refined database of reported b-Si solar cell results from the literature. The procedure was rapid and straightforward, and the generated database provided a comprehensive insight into b-Si nanotextures for solar cell applications. For example, the solar cell current improvement spectrum in Figure 5 was generated by 45 reported EQE spectrums from 17 independent research groups under certain criteria. It should be noted that the b-Si nanotextures used in the solar cells of these research works were highly distinct from each other. The figure shows that there is a solar cell current increase for the 300-600 nm wavelength range when replacing the conventional textures with the b-Si nanotextures. We can draw a solid conclusion that b-Si nanotextures are promising for solar cell applications as such textures will contribute to superior optical and electrical performance in the short-wavelength spectral region.

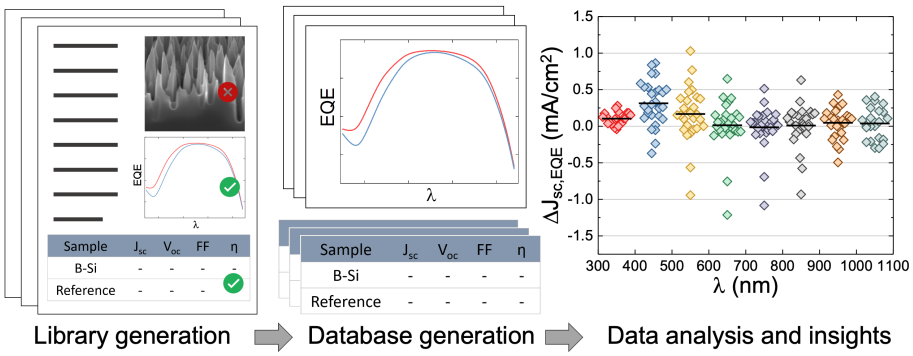


Figure 5: Schematic diagram of b-Si Nanotexture analysis pipeline.[]

5 Conclusions and Future work

This paper prepared a large nanomaterial literature collection with about 381,000 sentences and 34,000 graphs for model training. Then we developed two CNN models to extract scientific information from unstructured texts and graphs. Additionally, the deep learning model helped us to predict new thermoelectrical materials and provided valuable solutions to the problems of existing nanomaterial synthesis. Furthermore, we also use the graph information combined with the numerical simulation to compare the different nanotextures' performance. Our future work will address data recovery for more kinds of material characterization graphs. In summary, the results of both systems show that a CNN pre-classification process can improve scientific information extraction accuracy. The linkage between the text and graph information will provide a foundation to test and verify existing nanomaterial performance, which will be considered for future work. Our automated information extraction pipeline will boost nanomaterial data mining research by replacing the heavy work of manual information extraction. And we are also looking forward to building more downstream applications other than designing optimized material composition and synthesis methods.

References

[1] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[2] Daekyoung Jung, Wonjae Kim, Hyunjoo Song, Jeong-in Hwang, Bongshin Lee, Bohyoung Kim, and Jinwook Seo. Chartsense: Interactive data extraction from chart images. In *Proceedings of the 2017 chi conference on human factors in computing systems*, pages 6706–6717, 2017.

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.

[4] Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440, 2016.

[5] Mengqian Wang, Manhua Wang, Fei Yu, Yue Yang, Jennifer Walker, and Javed Mostafa. A systematic review of automatic text summarization for biomedical literature and ehrs. *Journal of the American Medical Informatics Association*, 28(10):2287–2297, 2021.

[6] Nadeesha Perera, Matthias Dehmer, and Frank Emmert-Streib. Named entity recognition and relation detection for biomedical information extraction. *Frontiers in cell and developmental biology*, page 673, 2020.

[7] Lucy Lu Wang and Kyle Lo. Text mining approaches for dealing with the rapidly expanding literature on covid-19. *Briefings in Bioinformatics*, 22(2):781–799, 2021.

[8] Matthew C Swain and Jacqueline M Cole. Chemdataextractor: a toolkit for automated extraction of chemical information from the scientific literature. *Journal of chemical information and modeling*, 56(10):1894–1904, 2016.

[9] Leigh Weston, Vahe Tshitoyan, John Dagdelen, Olga Kononova, Amalie Trewartha, Kristin A Persson, Gerbrand Ceder, and Anubhav Jain. Named entity recognition and normalization applied to large-scale information extraction from the materials science literature. *Journal of chemical information and modeling*, 59(9):3692–3702, 2019.

[10] Edward Kim, Kevin Huang, Stefanie Jegelka, and Elsa Olivetti. Virtual screening of inorganic materials synthesis parameters with deep learning. *npj Computational Materials*, 3(1):1–9, 2017.

[11] Haoyan Huo, Ziqin Rong, Olga Kononova, Wenhao Sun, Tiago Botari, Tanjin He, Vahe Tshitoyan, and Gerbrand Ceder. Semi-supervised machine-learning classification of materials synthesis procedures. *npj Computational Materials*, 5(1):1–7, 2019.

[12] Zach Jensen, Edward Kim, Soonhyoung Kwon, Terry ZH Gani, Yuriy Román-Leshkov, Manuel Moliner, Avelino Corma, and Elsa Olivetti. A machine learning approach to zeolite synthesis enabled by automatic literature data extraction. *ACS central science*, 5(5):892–899, 2019.

[13] Vahe Tshitoyan, John Dagdelen, Leigh Weston, Alexander Dunn, Ziqin Rong, Olga Kononova, Kristin A Persson, Gerbrand Ceder, and Anubhav Jain. Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature*, 571(7763):95–98, 2019.

[14] Kan Hatakeyama-Sato and Kenichi Oyaizu. Integrating multiple materials science projects in a single neural network. *Communications Materials*, 1(1):1–10, 2020.

[15] Bing Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012.

[16] Tanjim Ul Haque, Nudrat Nawal Saber, and Faisal Muhammad Shah. Sentiment analysis on large scale amazon product reviews. In *2018 IEEE international conference on innovative research and development (ICIRD)*, pages 1–6. IEEE, 2018.

[17] Venkata Sasank Pagolu, Kamal Nayan Reddy, Ganapati Panda, and Babita Majhi. Sentiment analysis of twitter data for predicting stock market movements. In *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*, pages 1345–1350. IEEE, 2016.

[18] Hao Wang, Doğan Can, Abe Kazemzadeh, François Bar, and Shrikanth Narayanan. A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In *Proceedings of the ACL 2012 system demonstrations*, pages 115–120, 2012.

[19] Kaoutar Ben Ahmed, Atanas Radenski, Mohammed Bouhorma, and Mohamed Ben Ahmed. Sentiment analysis for smart cities: state of the art and opportunities. In *Proceedings on the international conference on internet computing (ICOMP)*, page 55. The Steering Committee of The World Congress in Computer Science, Computer . . . , 2016.

[20] Brian Keith Norambuena, Exequiel Fuentes Lettura, and Claudio Meneses Villegas. Sentiment analysis and opinion mining applied to scientific paper reviews. *Intelligent data analysis*, 23(1):191–214, 2019.

[21] Abdallah Yousif, Zhendong Niu, John K Tarus, and Arshad Ahmad. A survey on sentiment analysis of scientific citations. *Artificial Intelligence Review*, 52(3):1805–1838, 2019.

[22] Parth Mehta, Gaurav Arora, and Prasenjit Majumder. Attention based sentence extraction from scientific articles using pseudo-labeled data. *arXiv preprint arXiv:1802.04675*, 2018.

[23] Vasileios Hatzivassiloglou and Kathleen McKeown. Predicting the semantic orientation of adjectives. In *35th annual meeting of the association for computational linguistics and 8th conference of the european chapter of the association for computational linguistics*, pages 174–181, 1997.

[24] Jaap Kamps, Maarten Marx, Robert J Mokken, Maarten De Rijke, et al. Using wordnet to measure semantic orientations of adjectives. In *LREC*, volume 4, pages 1115–1118. Citeseer, 2004.

[25] YP Zhou and Chew Lim Tan. Learning-based scientific chart recognition. In *4th IAPR International workshop on graphics recognition, GREC*, volume 7, pages 482–492. Citeseer, 2001.

[26] Manolis Savva, Nicholas Kong, Arti Chhajta, Li Fei-Fei, Maneesh Agrawala, and Jeffrey Heer. Revision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 393–402, 2011.

[27] Wenjing Dai, Meng Wang, Zhibin Niu, and Jiawan Zhang. Chart decoder: Generating textual and numeric information from chart images automatically. *Journal of Visual Languages & Computing*, 48:101–109, 2018.

[28] KV Jobin, Ajoy Mondal, and CV Jawahar. Docfigure: A dataset for scientific document figure classification. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 1, pages 74–79. IEEE, 2019.

[29] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[31] Juraj Mavracic, Callum J Court, Taketomo Isazawa, Stephen R Elliott, and Jacqueline M Cole. Chemdataextractor 2.0: Autopopulated ontologies for materials science. *Journal of Chemical Information and Modeling*, 61(9):4280–4289, 2021.

[32] Harold Mouchère, Richard Zanibbi, Utpal Garain, and Christian Viard-Gaudin. Advancing the state of the art for handwritten math recognition: the crohme competitions, 2011–2014. *International Journal on Document Analysis and Recognition (IJDAR)*, 19(2):173–189, 2016.

[33] Mahshad Mahdavi, Richard Zanibbi, Harold Mouchere, Christian Viard-Gaudin, and Utpal Garain. Icdar 2019 crohme+ tfd: Competition on recognition of handwritten mathematical expressions and typeset formula detection. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1533–1538. IEEE, 2019.

[34] Noah Siegel, Zachary Horvitz, Roie Levin, Santosh Divvala, and Ali Farhadi. Figureseer: Parsing result-figures in research papers. In *European Conference on Computer Vision*, pages 664–680. Springer, 2016.

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[36] Chinmayee Rane, Seshasayee Mahadevan Subramanya, Devi Sandeep Endluri, Jian Wu, and C Lee Giles. Chartreader: Automatic parsing of bar-plots. In *2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)*, pages 318–325. IEEE, 2021.

[37] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[38] Congcong Zhang, Junming Su, Tao Wang, Kaiping Yuan, Chunguang Chen, Siyang Liu, Tao Huang, Jianhua Wu, Hongliang Lu, and Aishui Yu. Significant improvement on electrochemical performance of limn2o4 at elevated temperature by atomic layer deposition of tio2 nanocoating. *ACS Sustainable Chemistry & Engineering*, 6(6):7890–7901, 2018.

[39] Qiuhong Yu, Rongjie Luo, Xianlin Bai, Wenchao Yang, Yang Lu, Xiaoyi Hou, Tao Peng, Xianming Liu, Jang-Kyo Kim, and Yongsong Luo. Rational design of double-confined mn2o3/s@al2o3 nanocube cathodes for lithium-sulfur batteries. *Journal of Solid State Electrochemistry*, 22(3):849–858, 2018.

[40] Brecht Put, Philippe M Vereecken, and Andre Stesmans. On the chemistry and electrochemistry of lipon breakdown. *Journal of Materials Chemistry A*, 6(11):4848–4859, 2018.

[41] Biwei Xiao and Xueliang Sun. Surface and subsurface reactions of lithium transition metal oxide cathode materials: an overview of the fundamental origins and remedying approaches. *Advanced Energy Materials*, 8(29):1802057, 2018.

[42] Jie Li, Yan Gao, Xinhua Liang, and Jonghyun Park. Ultra-thin coating and three-dimensional electrode structures to boosted thick electrode lithium-ion battery performance. *Batteries & Supercaps*, 2(2):139–143, 2019.

[43] Shuaifeng Lou, Yang Zhao, Jiajun Wang, Geping Yin, Chunyu Du, and Xueliang Sun. Ti-based oxide anode materials for advanced electrochemical energy storage: lithium/sodium ion batteries and hybrid pseudocapacitors. *Small*, 15(52):1904740, 2019.

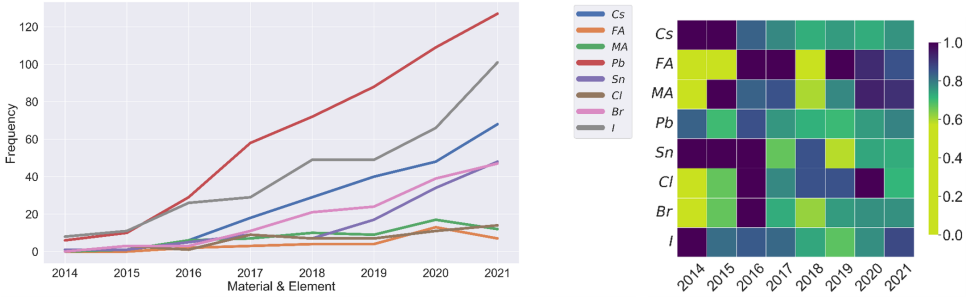
A Textmaster experiment details

Task	Methods	Precision	Recall	F1-score	Accuracy
Opinion extraction	TextBlob	0.2460	0.3007	0.2706	0.6432
	Unsupervised-lexicon	0.3525	0.3203	0.3356	0.7209
	Corpus comparison	0.4451	0.5033	0.4724	0.7525
	CNN	0.5614	0.6275	0.5926	0.8101
	CNN-Attention	0.5856	0.4248	0.4924	0.8072
	BiLSTM	0.5625	0.4706	0.5125	0.8029
	CNN-LSTM	0.5115	0.4379	0.4718	0.7842
	BiGRU	0.5319	0.4902	0.5102	0.7928
	CNN-SMOTE	0.6082	0.7712	0.6801	0.8403
Opinion classification	TextBlob	0.8211	0.8632	0.8416	0.7516
	Unsupervised-lexicon	0.7257	0.7008	0.7130	0.5686
	Corpus comparison	0.8977	0.6752	0.7707	0.6928
	CNN	0.8947	0.8718	0.8831	0.8235
	CNN-Attention	0.9266	0.8632	0.8938	0.8431
	BiLSTM	0.8707	0.8632	0.8670	0.7974
	CNN-LSTM	0.9035	0.8803	0.8918	0.8366
	BiGRU	0.8807	0.8205	0.8496	0.7778

Table 2: Performance of different methods on test set of EoL dataset

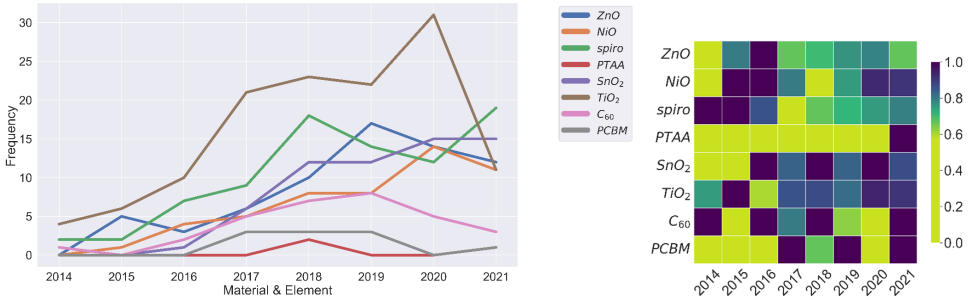
In opinion extraction task, precision here presents the fraction of correct predictions of opinion among the retrieved opinions, while recall is the fraction of correct predictions of opinion among all true opinions (because we focus more on opinions instead of non-opinions). In opinion classification task, precision here presents the fraction of correct predictions of opportunity among retrieved opportunities, while recall is the fraction of correct predictions of opportunity among all true opportunities (because we focus more on opportunities instead of challenges). F1-score is defined as the harmonic mean of precision and recall. The accuracy is the fraction of correct predictions among all predictions made. The SMOTE method was used to double the size of opinion category to further alleviate the data imbalance problem of opinion and non-opinion categories in the EoL dataset

B Opinion mining results of perovskite and thermoelectric materials



(a) (b)

Figure 6: (a) Frequency and (b) sentiment heatmap of popular perovskite elements



(a) (b)

Figure 7: (a) Frequency and (b) sentiment heatmap of popular perovskite materials

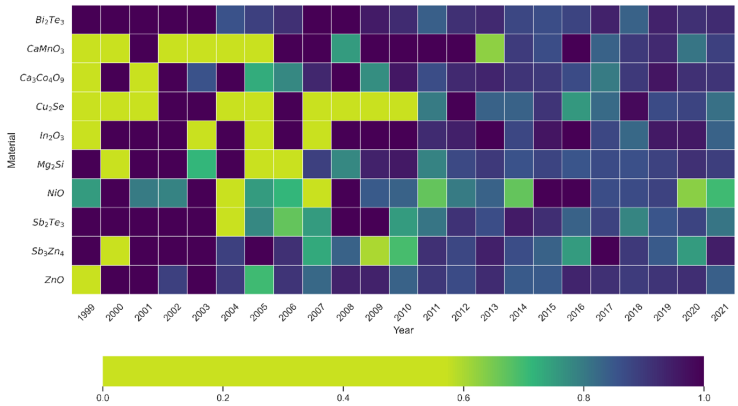


Figure 8: Sentiment heatmap of popular thermoelectric materials

C Graphmaster Classifcation Example

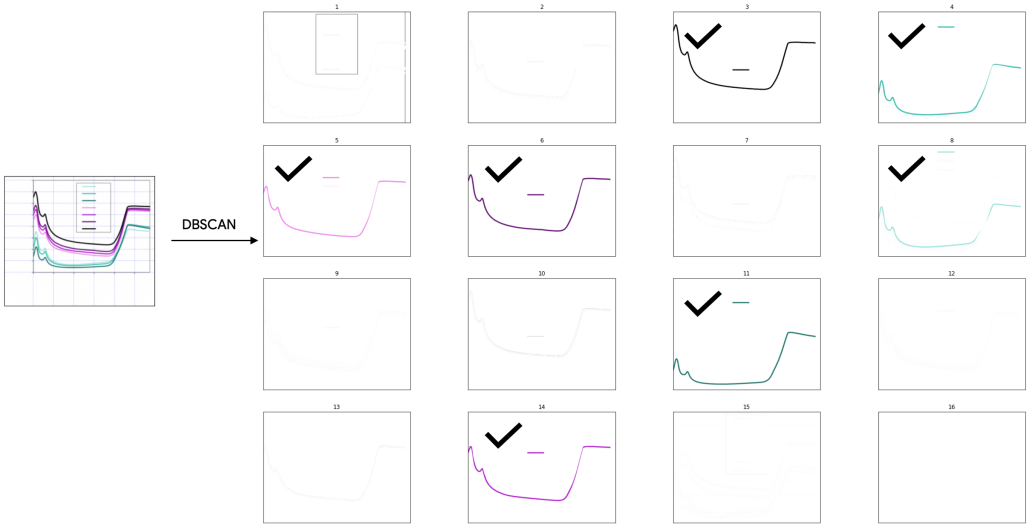


Figure 9: Process flow of Graphmaster Classification in example line charts

D Data Exatraction Results

Here we showed the detailed result of the example in fig. 2. Since we clustered the figure based on different RGB values. Therefore, the figure contains not only the data line but also the legends and some edges of other lines. For example, in figure 2, the author used green colours from dark to light to annotate the R1-R3 and purple colours from dark to light to annotate M1 to M3. Therefore, The R3 line's edge colour was the same as the R1. To solve that problem, we scanned the figure from left to right and only selected the data with a minimum distance from the previous y-value when there were two more pixels in one column.

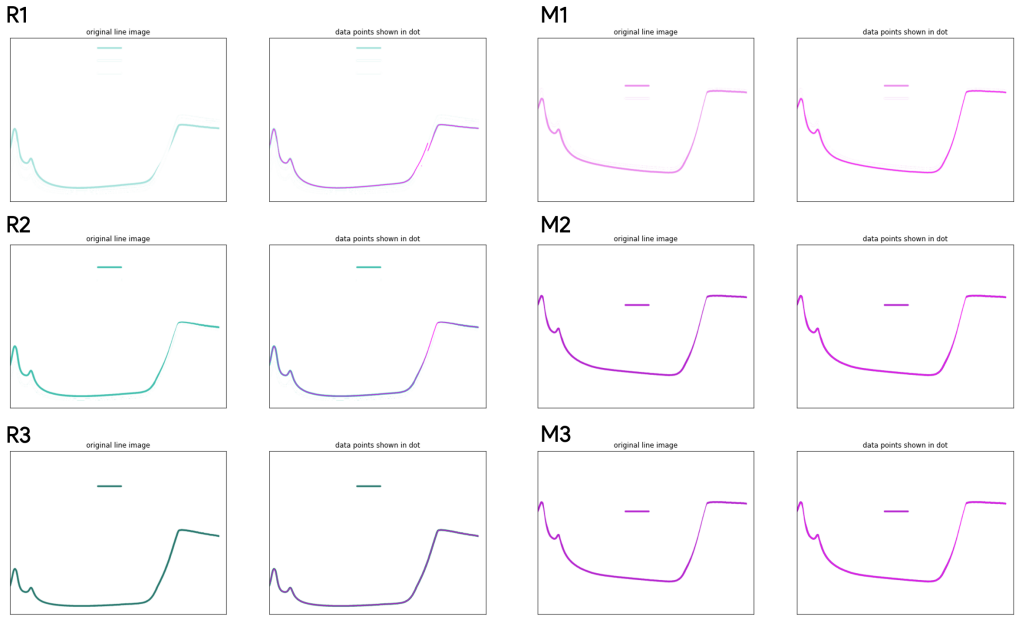


Figure 10: Details of data extraction in example line charts. We presented them with different legend labels. The left figure is the result after the colour cluster and the purple line in the right figure is plotted by the extracted data.

E Graphmaster Classification Experiment Result

The classification step is trained on the DocFigure dataset with 21K various types of charts and images. The ResNet model is trained on 1 Nvidia Tesla V100-SXM2 GPU, with 32GB memory size.

Table 3: Accuracies of different models on test dataset

		Accuracy			
Task	Methods	Line chart	Table image	Bar chart	Algorithm image
Graph clissification	ResNet	97.9%	97.33%	99.19%	98.78%
	DenseNet	96.9%	69.73%	71.88%	74.35%

The text detection process was tested on 120 wavelength-dependent EQE graphs.

Table 4: Accuracies of text detection

	Accuracy				
Task	X-axis title	X-axis value	Y-axis title	Y-axis value	Legends
Text Detection	98.3%	96.8%	98.4%	93.4%	56.4%

We showed the mean square error of the example image presented in Fig.2 below:

Table 5: Mean square errors of example figure

	Mean Square Error						
Task	iso	M3	M2	M1	R3	R1	R2
Data Extaction	5.6%	2.5%	3.5%	3.5%	7.1%	5.8%	2.1%