International Journal on Digital Libraries manuscript No.
(will be inserted by the editor)

Scientific Document Processing: Challenges for Modern Learning Methods

Abhinav Ramesh Kashyap* · Yajing Yang* · Min-Yen Kan

Received: 24 Jan 2022 / Accepted: 11 Jan 2023

Abstract Neural network models enjoy success on language tasks related to Web documents, including news and Wikipedia articles. However, the characteristics of scientific publications pose specific challenges that have yet to be satisfactorily addressed: the discourse structure of scientific documents crucial in scholarly document processing (SDP) tasks, the interconnected nature of scientific documents, and their multimodal nature. We survey modern neural network learning methods that tackle these challenges: those that can model discourse structure and their interconnectivity and use their multimodal nature. We also highlight efforts to collect large-scale datasets and tools developed to enable effective deep learning deployment for SDP. We conclude with a discussion on upcoming trends and recommend future directions for pursuing neural natural language processing approaches for SDP.

* – Corresponding authors, both contributed equally. This work was partially funded by the Singapore Ministry of Education's grant entitled "Scholarly Document Information Extraction" (MOE T1 251RES20237)

Abhinav Ramesh Kashyap
ASUS Intelligent Cloud Services (AICS), Singapore,
School of Computing, National University of Singapore, Computing 1, 13 Computing Drive, Singapore 11741
E-mail: abhinav_kashyap@asus.com,abhinav@comp.nus.edu.sg

Yajing Yang
School of Computing, National University of Singapore, Computing 1, 13 Computing Drive, Singapore 11741
E-mail: yajing.yang@u.nus.edu

Min-Yen Kan
School of Computing, National University of Singapore, Computing 1, 13 Computing Drive, Singapore 11741
E-mail: kanmy@comp.nus.edu.sg

Keywords Scientific Document Processing · Natural Language Processing

1 Introduction

A large number of scientific articles are published everyday, making it challenging for researchers to stay abreast of current developments in their fields. In biomedicine alone, researchers publish a new article every two minutes on average, resulting in more than a million publications per year [101]. This makes it difficult for researchers to find and read publications, synthesize and summarize them. Automated ways to help them in their daily activities are necessary. Automatic Scientific Document Processing (SDP) is such an avenue that it can enhance and simplify research tasks. For example, SDP-enabled digital libraries, such as Semantic Scholar[1] and Aminer[2], equip researchers with tools that search and filter papers, track citation counts, extract figures, tables, and equations, among other functions.

The current wave of modern neural network methods has enabled useful applications such as automatic summarization [16], the extraction of figures, tables and mathematical equations [31], and the recommendation of articles based on user interests [12]. It is natural to bring such advances to SDP, but so far the peculiarities of scientific publications have challenged conventional neural network models. For example, Long-Short-Term Memory Networks (LSTMs) [79], which are widely used, can process only a few hundred words at once, while scientific publications are much longer. This requires innovations in neural network architectures to enable the

[1] `https://semanticscholar.org`
[2] `https://www.aminer.org/`

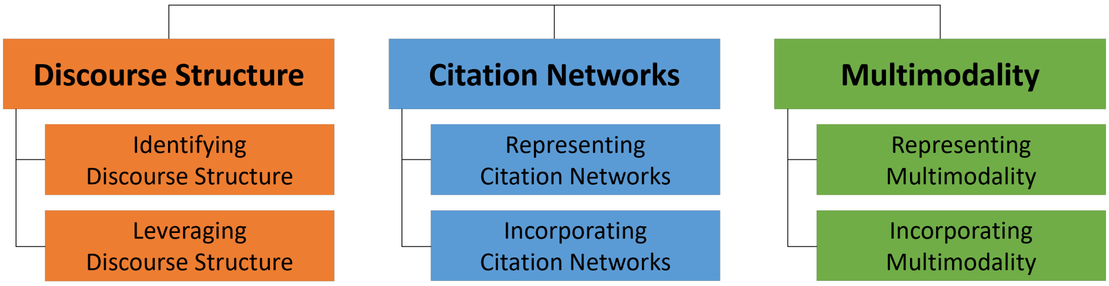
Discourse Structure
Identifying Discourse Structure
Leveraging Discourse Structure
Citation Networks
Representing Citation Networks
Incorporating Citation Networks
Multimodality
Representing Multimodality
Incorporating Multimodality

Fig. 1 We organize this article by the challenges offered by SDP to modern machine learning methods. The first challenge is in modeling and leveraging scientific discourse structure to improve the performance of models. A second challenge is in modeling and leveraging the inter-connected nature of scientific documents for neural network processing. Multimodality forms a final challenge, as neural methods currently handle non-textual modalities poorly.

processing of longer text [14]. Similar innovations are required to consider the hierarchical discourse structure of scientific documents (§ 2). Given these challenges offered by scientific document processing, can researchers reinvent modern methods to handle these peculiarities?

New methods and strategies to deal with the unconventional setting present in SDP have been on the rise. This warrants a literature survey to better understand the challenges and opportunities offered by SDP and the techniques developed to address them. Here, we synthesize the different challenges posed by SDP for neural networks, concluding each challenge with a representative rhetorical question.

First, scientific documents from STEM (Science, Technology, Engineering and Mathematics) fields follow a specific, conventionalized discourse structure [9]. In addition to identifying the different sections of the document, applications must effectively utilize this information for SDP tasks, such as summarizing and classifying citation intents, among others [32, 33]. For example, interpreting the purpose of equations is section-dependent; ones introduced in the evaluation section may explain how a work is quantatively evaluated, but ones introduced in a method section may describe key mathematical proofs. *How can we adapt vanilla neural architectures to deal with hierarchical document structure?*

Second, scientific work has the intrinsic characteristic of referencing prior work through citations. Citations serve many purposes. Citations are used to acknowledge the existence of closely related works, to refer to background knowledge beyond the scope of the current work, and to compare or contrast with other works, among others. The interconnected nature of scientific documents requires combining information from multiple documents to solve tasks such as citation recommendation [50], paper recommendation, and summarization. However, neural networks largely consider only one sentence or paragraph at a time. *How do we adapt neural networks to effectively incorporate information from multiple related documents?*

Third, scientific work is rich in its multimodal representations. In many subject areas, scientific work incorporates tables, figures, diagrams as embedded artifacts within the document itself. There are also auxilary artifacts related to the work, inclusive of data, computer code and other forms of attachments, that together with the manuscript provide a complete scientific package. *How can we represent and leverage such mulitmodal information to improve performance on key SDP tasks, and how do these multimodal artifacts enrich such tasks?*

We provide a review of the literature that answers these questions. Our contributions can be summarized as follows.

1. We identify three specific challenges (cf. Figure 1) that SDP poses to modern neural network learning models: discourse structure (§ 3), citation networks(§ 4), and multimodal data (§ 5). Then we outline the techniques to adapt such methods to overcome these challenges.
2. We collate and compare recent tools, datasets, and other resources that have been contributed by the SDP community (§ 6), which can serve as a starting point for parties in investigating possible solutions for SDP tasks.
3. We outline our vision for future challenges in SDP, especially considering how advances in neural network learning can be incorporated to forge meaningful progress for SDP (§ 7).

2 Background

We start by defining scientific document terminology and some representative tasks. These will feature throughout this article, although the tasks are certainly not exhaustive. Figure 2 illustrates the terms.

We conclude this background section by reviewing modern machine learning models that have spurred significant advances in the underlying methods for performing SDP tasks: modern neural networks. Although their mathematical underpinnings were present decades ago, only recently were such models sufficiently expressive to accurately capture detailed patterns from large amounts of textual and visual data. A basic review of such models helps frame the challenges we have identified.

2.1 Terminology

Discourse Structure: Scientific articles are divided into logical parts. Documents follow a conventionalized structure and typically contain an abstract (a summary of the paper), followed by the introduction, related work, methodology, and experimental results, often in this order. They are typically identified by their *section titles* or *headers.* We consider the logical organization of scientific publications as a discourse structure that aids in the discussion of scientific material. The logical parts have their own function and style. For example, the introduction provides the broader context of the research and mostly contains text, whereas the experimental section describes the experiments and may contain figures, tables, and mathematical expressions.

Citation: *Citing articles* (Figure 2, right) refers to older *cited articles* (left) using citations. Citations establish the claims made by authors, refer to methods and datasets, and credit the foundational work of other related papers. A *citation marker* — conventionally indicated with a number or an abbreviated form with the authors' name (indicated by the text "compared to [1]" in the figure) — marks the citation.

Citation Context: The text span around the citation contains contextual information, such as the reason for making the citation, information about the cited article. The text span (inclusive of the citation marker) may be limited just to the containing clause or sentence, but also may scope significantly beyond (both before and after) the immediate sentence. An example is marked in Figure 2), where the context continues to the following sentence and is marked in light blue.

Citation Intent / Citation Function: A citation can be made for various purposes: referring to background knowledge, comparing, and contrasting with another paper, and providing evidence to corroborate a statement. The citation intent provides the qualitative purpose of the citation, in contrast to *citation count* which merely provides the frequency of the citation of an article. The citation in Figure 2 *compares* with another paper.

Citation Provenance: An appropriate citation can refer to a specific text span in the cited article or, generally, to the entire work. The text spans in the cited article that are relevant to this citation is termed its citation provenance [162] (Figure 2, blue text in the cited article). Occasionally, authors do erroneously cite work – i.e., the cited work does not contain evidence supporting the citation context.

Citation String: The bibliography or the reference section of a scientific article contains a list of references, conventionally found as footnotes or endnotes (Figure 2, orange cross-hashed text in the citing article). Every item in the list — individually termed citation strings — contains necessary information to uniquely identify and locate the work: its authors, the publication venue and year, and other information.

2.2 Tasks

Scientific document processing encompasses many tasks for many stakeholders. Instead of reviewing all such tasks, our purpose is to highlight the challenges posed by scientific document processing for modern methods. As such, we focus our discussion on indicative exemplars that align with our three challenge areas, and the approaches that address them.

Keyphrase Extraction: Keyphrases are words and phrases that describe important aspects of an article: its main topic, materials or reagents, or methods [73]. The abstract section of Figure 2 shows the "cas-9 protein" — a protein associated with gene editing. Keyphrases aid different SDP tasks: indexing and searching documents by topics, clustering documents, recommendation, etc. Keyphrase extraction identifies and filters keyphrases from a publication.

Keyphrase Generation: Keyphrases may also generalize salient topics or be selected from a controlled vocabulary (keyphrase classification); and, as such, may not actually appear in the text. Topics can be described after reading and understanding a document. Generation differs from extraction, aiming to produce pertinent keyphrases including those that do not appear as-is.

Document Summarization: Summarization condenses a long document while still preserving key information. For scientific text, the summaries should contain background information, results given the context of related papers, the document's contributions, and their implications. Such salient information may appear in different parts of the scientific document. *Extractive* summarization extracts important sentences as-is from the document, while *abstractive* summarization does not draw

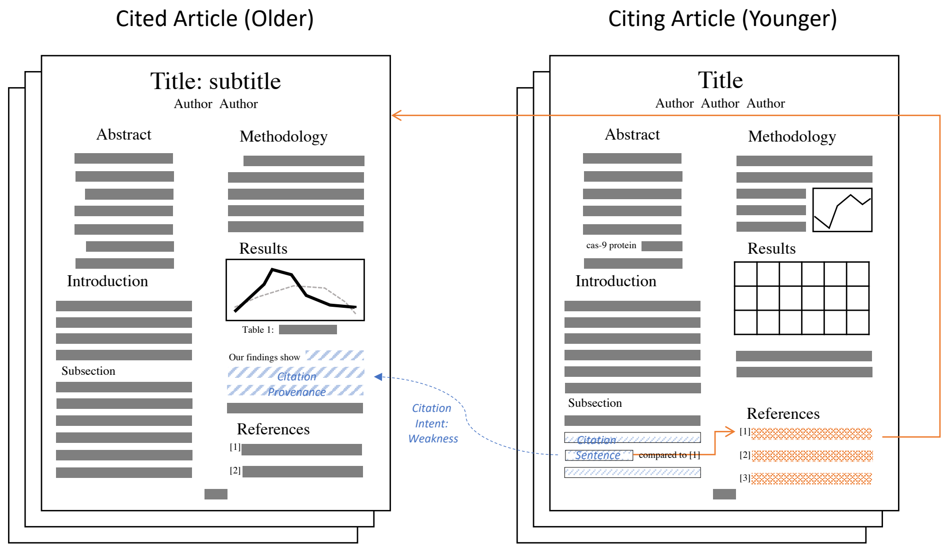

Fig. 2 (Best viewed in color) Pictorial representation of the structure of a scientific document and its related terms. Articles typically contain an abstract, and sectioned discourse such as introduction, methodology, etc. [169]. There are other discourse elements such as tables of results, figures, sections, etc. which play an important role in SDP. One article (*citing article*) may *cite* another article (*cited article*). The sentence which makes the citation is called the *citing sentence*. Relevant sentences before and after an anchoring *citing sentence* are called *citation text* and also plays a role for SDP tasks like citation intent classification. The text relevant to a citation in the cited article is called its *provenance*. The article may contain *keyphrases* useful to indexing and searching; in the citing article *cas-9* is a such keyphrase mentioned in the abstract.

sentences verbatim from the source. These forms of summarization are common to standard text corpora, such as news articles. However, the SDP summarization can capitalize on its unique structure of citing and cited papers. Citing papers provide the community's perspective of the paper, and can be considered a complement to the abstract. Summarization that consider citations is called citation-based summarization.

Citation and Paper Recommendation: Researchers may use aids to find relevant publications to cite or to read. For example, given a sentence "adversarial autoencoders generate realistic images and show improved performance", Makhzani *et al.* [123] is an appropriate citation providing background information about adversarial auto-encoders. Citation recommendation aims to suggest appropriate citations considering the statement, the context around the statement, the aim of the publication, and the coverage of citations in prior sections.

Citation Intent Classification: Citations are not equivalent. Citations express different sentiment — positive, negative, no sentiment — about the cited article [216]. They may also reflect different purposes: referring to background knowledge; indicating weaknesses, similarities, differences or improvements with respect to other publications. This task's aim is to automatically identify such intents. The results can then be used to selectively read specific related literature (e.g., list papers that provide an improvement over a target paper or ones that create a benchmark evaluation metric).

2.3 Neural Networks

Supervised machine learning algorithms train a model using a set of examples – also termed as labeled data points. To predict outcomes, traditionally, features are extracted from raw data. However, in real-world applications, manually extracting features is difficult. Modern approaches solve this by automatically learning such features. Modern neural network approaches specialize in learning such complex features by means of using multiple layers that forms the network architecture. This form of learning in which the user specifies the text and its corresponding label — without engineering any features — is called *end-to-end learning*. It is this ease-of-use, coupled with its impressive performance gains, that has led to the rapid adoption of modern neural methods in many communities.

Recurrent Neural Networks (RNN): Standard neural networks process inputs from beginning to end in one pass: they pass information to subsequent layers but do not reuse intermediate information, although such intermediate states can be useful. The recurrent architecture addresses this by reusing these intermediate

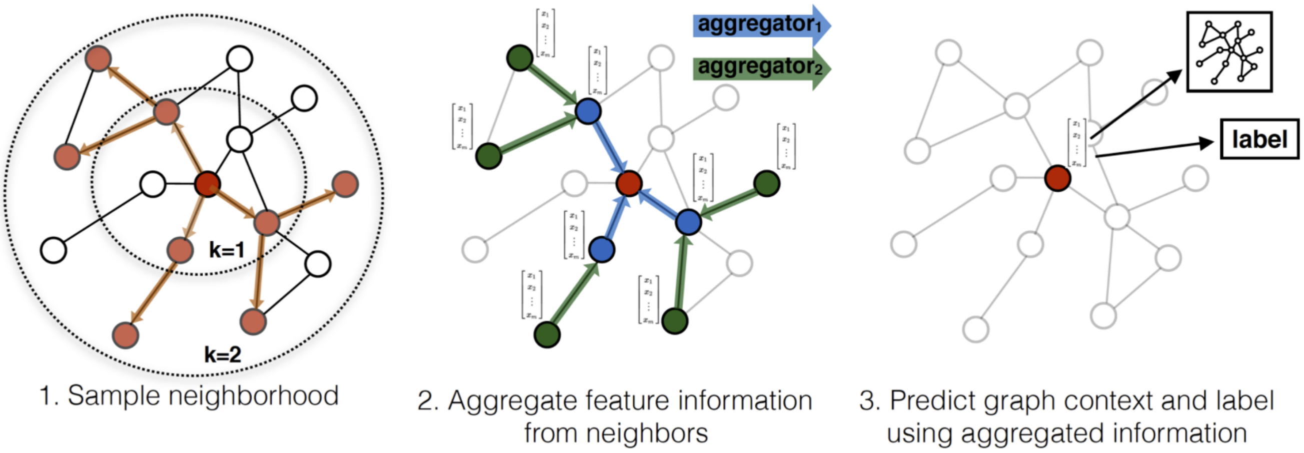
k=1
k=2
1. Sample neighborhood
aggregator1
aggregator2
2. Aggregate feature information from neighbors
label
3. Predict graph context and label using aggregated information

Fig. 3 An example of a graph neural network. It starts with sampling a set of neighborhood of the node. Within each layer of the graph, the NN-based aggregator aggregates information from a node's neighbors. The final representation of the node is used to predict graph context and label [71]

results from previous data points for future ones. RNNs are used to process sequences, where information from previous states plays an important role for the current state, such as text where previous words have predictive power in determining the next words as well as the overall meaning. In SDP, they are commonly used to process sequences of text from scientific articles for tasks like classification. While RNNs made significant progress in text processing, transformer models [44] have been prominent in the recent past. We refer the reader to an online post[3] for an inituitive introduction.

Convolutional Neural Networks (CNN): Another useful variation of general neural networks are Convolutional Neural Networks. CNNs specialize in understanding spatial inputs like images, 2D blocks of text, tables, or figures. They build representations by considering spatially local information, like a patch of image, and further compose these local representations to build representations of larger spatial blocks or volumes, until the entire input is considered. In SDP, CNNs find their application for tasks such as identifying discourse structure, where considering the entire article at different resolutions is critical. Other applications like automatic understanding of tables and figures also use CNNs, as these inputs also have locally-spatial regularities in the form of recursive, decomposable, top-down structure.

Graph Neural Networks (GNN): Unlike images or text, a graph consists of unordered nodes of no fixed form. Graph neural networks handle the complex topology of a graph by adopting the idea of convolution to look at only the local network neighborhoods of a node. Convolution builds representations using information from its local neighbors, as in CNNs; but in GNNs, the set of spatially local neighbors is dictated by the random graph and not fixed as in 2D images and tables. As shown in Figure 3, GNNs determine a computation graph for each node and learn each node's representation by aggregating its neighboring information. The resulting representation comprises both the node features and the graph topology. GNNs produce state-of-the-art performance in node classification, link prediction, and clustering tasks.

RNNs, CNNs and GNNs are all basic variants that specialize in handling regularities in the input; specifically, sequential, fixed spatial and random spatial regularities, respectively. All of these require sufficient data to set the weights of these models appropriately. We end by mentioning pre-training as a means to obtain good initial weights for many of these models.

Pretrained Language Models: Supervised learning requires large quantities of labeled data, often difficult to obtain. Can we first learn general information from the unlabeled, publicly-available text on the Web? Can we further use this information to learn appropriate tasks with limited labeled data? The answer to these two questions is "yes," where the first task is referred to as "pretraining", and the second as "finetuning". This pretraining revolution first uses unlabeled text to learn useful associative patterns, which address many shortcomings in language understanding and meaning interpretation. While pretraining in natural language research was first limited to the word level (Word2Vec), computer vision researchers trained deep CNNs in the ImageNet dataset for image recognition [42], learning generic, hierarchical representations of images such as edges, curves, and textures. They further finetuned these

[3] `https://lilianweng.github.io/lil-log/2020/04/07/the-transformer-family.html`

representations for advanced computer vision tasks such as object detection [154] and image captioning [89].

This pretrain-and-finetune task decomposition is applicable for text processing as well. Researchers trained deep neural networks to predict how likely the next word is given a piece of text (language modeling). The aim is to learn general patterns of language while performing this task. As of current, the most extensively-used pretrained language model is BERT [44]. After learning the general representation, finetuning the language model improved performance of other NLP tasks: sentiment analysis, semantic text similarity, assessing the linguistic acceptability of text, etc. [178]; all of such finetuning operations have been applied successfully to BERT.

3 Discourse Structure — Challenge #1

Scientific discourse structure divides the document into logical parts. We consider any such logical organization as *discourse structure*. Leveraging such a structure is crucial for SDP tasks. As an example, consider keyphrase extraction. Keyphrases are often found in particular sections (e.g., title, abstract, and introduction). However, neural networks do not natively incorporate discourse structure. Such a sequential, end-to-end processing largely disregards the hierarchical discourse structure of the scientific publications. Even the first step in identifying the structure of the discourse is challenging. We discuss methods to identify discourse sections (§ 3.1), then methods to take advantage of them (§ 3.2) to improve the performance of downstream SDP tasks.

3.1 Identifying Discourse Structure

The backbone of several SDP tasks requires the identification of logical sections in a scientific document, which is a necessary step to leverage discourse structure in downstream tasks. Here, we review neural network methods that analyze scientific publications, although there has been significant informed work on other structured documents such as receipts [43, 90, 91], web documents [135], business documents [185], examination papers [120], among others. The reader is invited to refer to [49, 166] for a comprehensive survey on other types of documents. Table 1 shows the comparison between neural methods to identify the structure of scientific discourse. We inventory such NN-based methods based on two dimensions of comparison: the discourse elements identified (Aim), and the NN approach used (Approach).

1. **Aim**: Kan et al. [88], a precursor non-NN method, associates every line with one of 23 line functions (i.e., section header, title, page number, figure caption, etc.) and further classifies sections (i.e., a section header accompanied by body text) into one of 13 generic logical types (introduction, related work, methods, etc.). Recent NN methods [199] have greatly improved the extraction of similar discourse structures, albeit only at the coarse-grained section level and without logical types (e.g., Allen AI's Science Parse; cf. Table 3).
   In addition to text, scientific documents incorporate visually-distinct figures, tables, and mathematical equations. They provide a summary of the methods and results which aid in finding related papers, among other uses. NN models also extract such multimodal objects, such as figures [160] and tables [161]. We limit our discussion to the discourse structure of text, leaving the details about the multi-modal elements to Section 5.
   All the papers discussed till now consider a high-level structure within the document — such as the abstract, introduction, and methodology — which we term *global*. Identifying global structure aids in document-level tasks such as recommendation, summarization, retrieval, etc.
   Finer, *local structure* within individual sections can help researchers assess the appropriate reading and writing strategies from other works. For example, Jin and Szolovits [85] identify the background, objectives, methods, results, conclusions, and order within the abstract section of scientific publications. They analyze the prevalent order in scientific publications, which can help researchers structure their own abstracts. Banerjee et al. [13] also identify the local structure of abstracts. They first train a model on biomedical data to identify categories before finetuning on a small set of computer science articles. Similarly, Dasigi et al. [41] identify the different components of the experiment section: the problem, the goal, and the results of the experiments. Analyzing fine-grained sections aids in automatic literature review by grouping works that use similar methods and outcomes.
2. **Approach**: Researchers use two types of neural network to identify discourse structure: Convolutional neural networks (CNN) and recurrent neural networks (RNN) (cf. § 2). Researchers choose CNNs or RNNs depending on the way they treat scientific publications: preferring RNNs when they treat them only as textual sequences [41, 85].

Recommendations: Neural methods are increasingly used to identify the discourse structure of the textual content. We find that the community is focused on identifying the global structure of scientific publications. But

Paper	Aim	Approach	Code Availability
Soto and Yoo [161]	Identify title, authors, abstract, body, etc ***(Global)***	Faster-RCNN (***Vision***)	
Science Parse	Identify title, authors, section headers, etc. ***(Global)***	Bi-LSTM (***Text***)	
Yang *et al.* [199]	Identify title, authors, abstract body, etc. ***(Global)***	CNN + text embeddings (***Hybrid***)	
Dasigi *et al.* [41]	Identify the problem, method, implications, etc. within the experiment section ***(Local)***	LSTM (***Text***)	✓
Jin and Szolovits [85]	Analyze the abstract of the scientific paper ***(Local)***	Hierarchical LSTMs (***Text***)	✓
Madisetty *et al.* [122]	Parse Mathematical Equations ***(Global)***	Bi-LSTM (***Hybrid***)	
Wang and Liu [184]	Generate Mathematical Latex Equations ***(Global)***	LSTM + CNN (***Hybrid***)	
Siegel *et al.* [160]	Extracting Figures ***(Global)***	CNN (***Vision***)	✓
Banerjee *et al.* [13]	Identify background, techniuque and observation from abstracts ***local***	Bi-LSTM ***Text***	✓

Table 1 Overview of different neural network based prior work that identify the discourse structure of scientific documents. **Aim** captures the diversity in the end goal of articles. The **Approach** identifies the neural network method used to identify discourse structure. **Code Availability** refers to availablity of the project code; where available, URLs to codebases are listed in the respective reference in the bibliography.

analyzing the local structure of publications can result in tools for efficient reading and writing for scholars [13].

3.2 Leveraging Discourse Structure

Modeling the discourse structure of scientific publications provides several advantages to downstream SDP tasks: division of long documents into small logical sections, providing prior information for certain tasks (citation intent classification), and allowing the comparison of publications based on sections. Here, we review NN methods that use discourse structure to benefit SDP tasks. Table 2 summarizes works based on their task, the modeling approach used to incorporate the discourse structure, and the NN architecture used.

1. **Task**: The modeling of discourse structure informs the downstream models in performing the key characteristic SDP tasks (i.e., document summarization, keyphrase extraction, and citation intent classification). Summarization approaches based on models of standard text [156, 157] underperform on scientific documents, due to their long and conventionalized document structure. Knowledge of discourse structure pinpoints where specific forms of knowledge lie (key aspect of the methodology in *Methods*, discoveries in the *Results* section), allowing summarization methods to model different functional aspects of the document. Both extractive [35, 188] and abstractive SDP summarization [16, 22, 33, 61] leverage discourse structure for this reason.
In addition to summarization, keyphrase extraction benefits from incorporating discourse structure. As keyphrases capture salient information about the paper, they concentrate within certain sections, such as in the methodology or the introduction. Similarly, other works [30, 100, 203] observe that the title of the document largely overlaps with keyphrases, and that modeling titles for keyphrase extraction improves performance.
The intent of the citations also depend on the section in which they appear. For example, most computer science papers compare and contrast with other works in the literature review section, while the citations in an introduction provide background knowledge. Therefore, it is important to incorporate discourse information for automatic classification of citation intent. [32, 164] are exemplars that incorporate the discourse structure in neural networks for citation intent classification.

Paper	Approach	Task	Method Description	Code Availability
Cohan *et al.* [33]	End-to-End	Abstractive Summary	Hierarchical Attention Networks	✓
Xiao and Carenini [188]	End-to-End	Extractive Summary	Attention over word, sentence and document representations	✓
Collins *et al.* [35]	End-to-End	Extractive Summary	Categorical Feature added to neural network	✓
Gidiotis and Tsoumakas [61]	Divide-and-Conquer	Abstractive Summary	Pointer Generator RNNs per section	
Chaturvedi *et al.* [22]	Divide-and-Conquer	Extract + Abstractive Summary	BioBERT + Graph based extractive summaries + BART	✓
Kobayashi *et al.* [99]	Divide-and-Conquer	Citation Recommendation	Word Embeddings + Simple Classifier + Graph Neural Networks + Recommendation	
Chen *et al.* [30]	End-to-End	Keyphrase Generation	Attention based Encoder-Decoder RNNs	
Cohan *et al.*[32]	Multitask	Citation Intent Classification	LSTM networks	✓
Su *et al.* [163]	Multitask	Citation Provenance and Intent	LSTM networks	✓
Cachola *et al.* [16]	Multitask	Abstractive Summary	BART	✓
Ye and Wang [203]	Multitask	Keyphrase Generation	Encoder-Decoder RNNs	

Table 2 Overview of methods incorporating discourse structure for an end task in scientific document processing. **Approach**: We identify three main approaches to incorporate discourse structure into scientific articles. *End-to-End, Divide-and-Conquer, Multitask* learning. **Task**: Incorporating discourse structure benefits different SDP tasks, and we identify the different tasks tackled by different works. **Method Description**: A brief description of the neural networks used in the work. **Code Availability**: where available, a hyperlink to the codebase is given in paper's reference.

2. **Approach:** We identify three ways that neural networks incorporate discourse structure: *end-to-end, divide-and-conquer,* and *multitask.*

   End-to-end : End-to-end methods build continuous representations for a discourse section, starting from sequential text. Since forming representations for longer scientific documents is harder for neural networks, they compose representations of smaller elements like words [130, 146] and sentences [36, 98] to form representations of larger elements such as whole documents. Hierarchical attention networks [200] are one such framework that combine continuous word representations with sentence representations, further combining them to obtain section- or document-level representations using the attention mechanism [11]. Building document and section level representations using hierarchical networks have been shown to be useful for abstractive summarization[33], extractive summarization [188] and keyphrase generation [30]. Incorporating discourse structure to solve a SDP task using the popular end-to-end paradigm requires complex neural network architectural changes. Since scholarly documents are long, using neural networks to sequentially process documents is inefficient and ineffective, due to high computation and memory requirements.

   Divide-and-Conquer : To ease the burden of pure end-to-end learning, divide-and-conquer approaches are helpful. As the discourse structure of the documents naturally helps to divide the problem into smaller ones, smaller section-wise solutions can be solved and combined later. For example, a separate summary can be formed for different discourse sections and combined to form a final summary. Such an approach is popular for summarizing [22, 61], citation recommendation [99], and paper recommendation [143].

   Multitask : Multitask learning considers a main task and a complementary auxiliary task together.

The two tasks exploit the commonality and the differences between them to improve generalization [19]. It has been used in innovative ways to employ discourse structure in neural networks for various SDP tasks. Generally, an objective — related to the discourse section to be incorporated — is added as an auxiliary task to the main SDP task. For example, Cohan et al. [32] predict the title of the section as an auxiliary task for citation intent classification. Su et al. [163] show that the detection of citation intent and citation provenance can enhance each other's performance in a multi-task setting. Cachola et al. [16] use multi-task learning to generate extreme summaries of scientific documents that are a couple of sentences long; perfect as search result snippets. They use an auxiliary task of title generation, finding this multitask setup improved performance. Similarly, title generation has been used as an auxiliary task for keyphrase generation [203].

Multitask learning has multiple advantages: improving the generalization ability of a model's solution, and efficiently modeling a new task with minimal data. This is accomplished by incorporating information from discourse elements related to the task to obtain performance gains.

Commonalities and differences: Most end-to-end based approaches compose word and sentence representations to form discourse level representations using attention-based RNN networks [11]. Cohan et al. [33] uses RNN encoders to build sentence-level representations from word representations [33, 188]. Once the document has been encoded, such methods employ attention in the decoder to model differing levels of section importance when generating summaries. Similarly, Xiao and Carenini [189] uses the intuition that a decision to include a sentence in the extractive summary depends on not only the importance of the sentence within a section, but also on its importance within the entire document. To achieve this, they use an attention mechanism over the hierarchically-composed word, sentence, discourse section, and document representations to decide whether to include the sentence in the extractive summary.

On the other hand, instead of making complex architectural changes that become infeasible to handle long documents, the *divide-and-conquer* approaches use a pipeline-based approach. For example, Gidiotis and Tsoumakas [61] first classifies an annotated summary sentence into different discourse sections, creating a pseudo section-summary supervised pair. Then they use a neural network for abstractive summarization of every section before combining them.

The *divide-and-conquer* also has inspired a hybrid *extract-abstract* summarization approach that first extracts a sentence and then perform abstractive summarization. This method has become popular not only in non-scientific domains [105, 129, 190] but also in scientific documents [22]. *Divide-and-conquer* approach can combine multiple existing methods to achieve a better result [22, 61, 99] – allowing easy switch of components for more advanced ones, it handles longer documents effectively without complex changes to neural network architectures.

Multitask-based approaches share a few layers of a neural network like LSTMs [32, 163] and then use separate layers for each complementary task to capture task specificity. Although earlier works used multitask learning for mainly classification tasks [32, 163], it has found a resurgence in text generation [16, 203].

Recommendations: Incorporating discourse structure into neural network modeling is essential to improve SDP tasks. Although traditional methods can incorporate discourse structure by considering them as additional features, neural network methods best model discourse structure by making appropriate architectural changes to build hierarchical representations to incorporate such information. End-to-end methods for incorporating discourse representations are currently limited by the length of text that can be processed at once – ineffective in handling long paragraphs or documents. New methods such as [14] have attempted to alleviate this challenge recently. Our opinion is that the *divide-and-conquer* approach can leverage existing technologies creatively and provide more practical solutions.

4 Citation Networks — Challenge #2

Scientific documents link to each other using citations (Figure 4). Citations relate the scholarly work to the background and context of the works and relevant concepts. A collection of papers forms a citation network or graph, where edges model the citation relationship between two papers. The bibliographic details (i.e., author, year, publication venue, title) and the content of a paper are the node features in such a citation network. The citation context highlights the purpose of a citation or the edge type of citation network edge.

To properly utilize citation information under a neural scenario, it is essential to generate effective numerical representations for both the node and edge features of a citation network. In addition to component representations, an appropriate neural architecture needs to be chosen to represent such citation networks as a

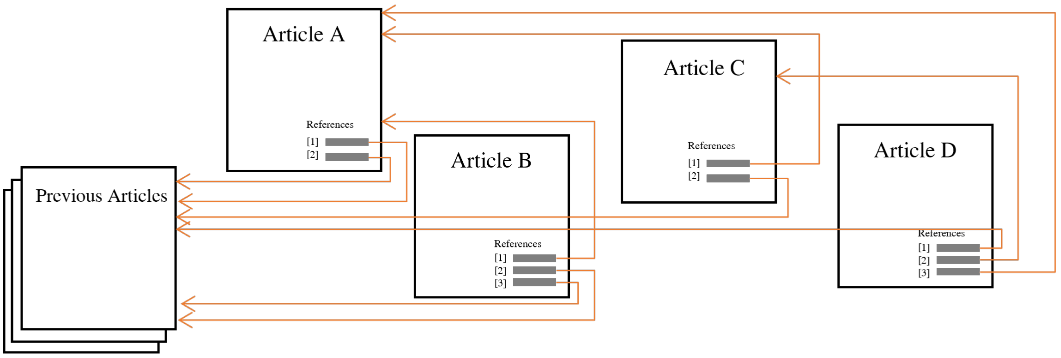

Fig. 4 Pictorial representation of a citation network. Articles cite the previous articles and are cited by future articles. The number of citation consisted of and received varies across different articles.

whole. Despite the success of CNNs and RNNs in handling computer vision and natural language processing tasks, they are designed to handle tabular (raster) or sequential data. However, citation networks have neither spatial logic nor fixed ordering of nodes. Graphical representation models built with NNs, such as Graph Convolutional Networks, yield superior performance in extracting the node features and graph structure automatically. We start by introducing the application of the NN-based graph representation methods on citation networks, then describe how these representations are utilized in SDP tasks.

4.1 Representing Citation Networks

Although NN-based methods achieve outstanding graph representation performance, naïve application of such methods on citation networks fall short in performing well. While some works have explored how to handle the challenges posed by the unique characteristics of citation networks, many areas remain to be explored.

Node characteristics. Bibliographical information includes nominal data (e.g., author, publication venue, and year) as well as textual strings (e.g., title, abstract). We discuss examples within these two classes of data in turn.

Author information can be utilized to assist in topic modeling, as authors preferentially publish on only certain topics. However, when a large number of co-authors are present on a publication, it becomes more difficult to characterize the work, making a low-dimensional representation of author essential. A common method is to aggregate all the titles of an author's publications [205]. Sugiyama and Kan [167] extend this representation method to the referenced paper and papers citing the scholar's work, while Bulut et al. [15] include the documents related to the scholar's research field. Ebesu and Fang [47] build two networks to learn the embedding of cited and citing authors separately, and incorporate the output during decoding. Holm et al. [80] models the author and venue with the total citations received.

In our view, the utilization of the venue and year information is still superficial in contrast to the many works that utilize author information. A venue representation aggregates all work grouped under the same research cluster could help to create a better representation for the target paper. Further, the prestige of a publication venue (e.g., impact factor of a journal or ranking of a conference) is important in deciding the influence of the paper. The publication year of a work can be used in tasks like citation recommendation, to account for its fading novelty as a publication ages [127].

Most works extract textual features from title and abstract using a recurrent network (specifically, LSTM) [180, 201] or pretrained language model [84] for graph representation learning. While titles and abstracts provide summaries of documents, the full text contains information not available in these two. However, the structured and lengthy nature of scientific documents poses challenges for standard RNN or transformer architectures to encode. Future works should also consider a faceted representation as introduced in Section 3, based on the needs of downstream SDPs.

Edge characteristics. The edge in a citation network represents the citation relation between two papers. Such edges are directed from the citing paper to the cited one and can be associated with a time lag and its citation context. Li et al. [110] build Graph Neural Networks (GNN) and handle the directions by using only the incoming nodes in identifying important information during aggregation. Jin and colleagues [86] model the citation directions using two separate networks. The time lags are the time difference between publication date of the cited paper and citing paper, which shows the diffusion speed of information from the cited paper. Fu et al. [56] model the time lags using Monte Carlo simulation. The resulting citation network is then deployed to improve the results of a citation recommendation task.

The citation function, or the reason a citation is included, can be extracted from citation context [173]. The citation context often implies the function and polarity (negative/neutral/positive) of a citation too [2, 60]. Future works will be able to produce better representation of citation relation by incorporating these derived features. Citation context is also the basis for certain SDP tasks. For example, a context-aware citation recommender aims to find papers that best match a given citation context [40, 84, 171, 180, 197, 198, 205]. And a stream of summarization models use citation context as the input for summary construction [111, 201, 208]. Citation context can also be used to generate document embedding by treating the citation context as sentence and citation as word [58, 72].

In summary, a node in a citation network is associated with rich information including metadata (author, year, publication venue, title, abstract) and a long document content. The features of a citation include date, function, and citation direction. Although research has been done to utilize these features, the incorporation of publication venue and year, the textual representation of the document content and its application of citation function/intention derived from citation context remain as challenges in building an effective and universal citation network representations.

4.2 Incorporating Citation Networks

Appropriate modeling of citation networks using the above methodologies significantly boosts performance on various SDP tasks. Although many applications such as citation forecasting [74] and extracting emerging concepts from scientific documents [97], we select two key SDP tasks — citation recommendation and citation-based summarization — where network representation is critical and forms the basis for their approach.

Citation Recommendation. Citation recommendation (and also § 2.2) is accomplished based on either the global or local context. A global citation recommendation model recommends papers according to their similarity with a given document. Gupta and Varma [68] randomly generate a set of short walks from the citation network with each node as the origin, and find the node representations to maximize the probability of the generated walks. This topology representation is combined with semantic features to yield document representations by maximizing the correlation between the two. Another method based on random walks generation proposed by Guo et al. [67] merges the citation relation and content similarity between papers as the input for node representation learning. Other works treat all bibliography entities as nodes and build bibliographic networks that contain nodes corresponding to different entities such as author, venue, and paper (Heterogeneous Bibliographic Networks or HBNs). Cai et al. [17] represent an HBN with an adjacency matrix to train a GNN-based recommender. Models proposed by Ma and colleagues [118, 119] generate node embedding by extracting topological features from HBNs using different meta-path proximity methods. Mu et al. [133] model an HBN as a multi-layered graph (for authors, papers and words, respectively), while incorporating user queries into the graph to yield a query-focused recommender.

In contrast, local (or context-aware) citation recommendation produces recommendations for a specified text span where a citation is needed. Such applications can benefit writing assistance and can incorporate citation impact prediction as well [139]. These recommenders generate a representation for both the citation context and each candidate paper, and make recommendation based on the similarity score between the two. Jeong et al. [84] predicts the probability of recommending a paper based on the concatenation of citation context embedding encoding generated using BERT, and a document embedding produced with GNN. Medić and Snajder [126] also tackle the local citation recommendation problem, incorporating article title and abstract information – hinting at the productivity of incorporating discourse structural information (c.f. § 3).

Other works compare the paper content and the given citation context directly. Neural NLP methods are commonly applied to create the text representation [40, 142, 171, 180, 197, 198, 205]. In contrast to the aforementioned works, which generate one representation for each paper, [51, 72] build two vectors per paper: one for each case of citing and being cited, respectively.

Citation-based summarization. Citation graphs enable improved contextual modeling of a paper, and hence can improve summarization performance. As an example, certain scientific concepts may not be explicitly explained, where the document authors refer the reader to other documents through citations to explain the necessary concepts. The citation graph-based summarization model proposed by [7] incorporate this information in the referenced paper using GAT. The input features of the nodes are the textual features extracted from the abstracts of the referenced paper, inferred from an LSTM. The node representations are then fed to a LSTM-based decoder to yield attention weights, which are then applied to choose content for document summaries.

Citation sentences are usually in the form of a summary of the aim, method, or results of the cited work. A set of such sentences for the cited documents become useful data sources for generating a summary. Qazvinian and Radev [149] proposes citation summary networks where each node represents a citation sentence referencing the document of interest, and where similarity scores between nodes determine edge weights. Sentences extracted from a clustering of the weighted graph form the summary. Chen and Zhuge [26] design a multi-document summarizer employing citation sentences. They cluster the citation sentences to identify common facts and them as weights (in the same guise as attention weights) to select sentences from input documents to form a summary.

However, such citation sentences are written from the referrer's perspective, hence tend to be subjective. Recent works solve this issue by also utilizing citation provenance. Here, citation provenance refers to the most similar to the citation sentences found in the target document, which serve as the proxy for the provenance or reason for the citation. Classifiers built with pretrained BERT [208] or CNN [111] find citation provenance by treating it as a sentence pair classification problem. Yasunaga et al. [201] uses the abstract and the citation provenance as input to build a GNN. The node representation output from GNNs serves as a salience estimation in selecting summary sentences.

Recommendations: Incorporating citation information boosts the results of many SDP tasks. As discussed, a node representation of a document in the citation network provides a community view of the document; hence, it is useful in identifying related papers for citation recommendation. The citation context serves as a query to indicate the users' interests in citation recommendation. As for summarization, appropriate modeling of citation networks helps in understanding scientific concepts. Citation contexts are used directly as candidate summary sentences, or indirectly as the references for document sentence extraction.

In this review, we have only concentrated on core tasks in SDP in which citation network modeling is essential. Our opinion finds that citation modeling can assist in tasks where the contextual positioning of how a publication fits in the community's understanding is needed. Such applications extend beyond summarization and recommendation and could include trend analysis, survey paper construction, field and topic characterization, expert finding, author and institution reputation tracking, among many other future creative uses. However, the current studies apply task-agnostic graph representation, resulting in suboptimal performance. Further enhancements should be made by incorporating or attending to document information for the task of representing the citation networks. Besides, while each discipline demands its own metadata in the references [46], identifying the optimal set of metadata entities in effective citation networks construction requires further study.

5 Multimodality — Challenge #3

Scholarly documents are richly formatted documents, in the sense that they are not exclusively corpora of running text. Tables, figures, line art, and workflow diagrams are among some of the artifacts that are used to communicate scholarly work. Such artifacts are often embedded within the document itself (we term these *internal* artifacts) that allow a reader to peruse data to validate the findings highlighted by the authors. These internal artifacts are often accompanied by a textual *caption*, allowing these independent artifacts to be better understood and contextualized within the parent scholarly work. At the one end of the spectrum of multimodality, scholarly documents also have other *inlined* textual artifacts, usually symbolic notation — such as ones for music, chemistry and mathematics — or domain-specific entities that are used to encode and transmit domain-specific information. At the other end of the spectrum, datasets, program code, posters, presentation slides and videos, are satellite, complementary *external* artifacts that serve to complete a scholarly work, often facilitating secondary uses of the work, inclusive of replication, communication and use.

However, many tasks requiring modality considerations appeal to methods that concern just a single modality. As such, specific, single-modality methods are utilized. For example, techniques in computer vision utilizing deep learned CNNs are used for representing and modeling internal artifacts such as figures and tables.

5.1 Representing Multimodality

To focus on the challenges in *multi*modality, we constrain our discussion to the representation of more than one modality. We summarize multimodal representation methods in different combinations of modalities, constraining our discussion to modalities common in scientific documents. As the textual modality is central in scientific document processing, we examine multimodal representations in which text is combined with other modalities.

Text + Figures/Tables. Figures and tables are internal artifacts commonly seen in a scholarly document, containing information not covered by the text. To generate a joint representation of texts and figures of a document, a general framework is to extract the unimodal high-level features separately and then combine the two sets of features. Anastasopoulos et al. [8] apply concatenation to fuse the textual and visual features extracted using two pre-training models. The models proposed by Lu et al. [117] and Tan and Bansal [170] are of similar structure, but fuse features differently. They consist of two single-modal networks to separately encode input text sequences and images, and a cross-modal transformer to combine modalities. Zhang et al. [209] project features from one modality to the other as cross-modal attention, and design a cross-modal transformer that models both self-attention and cross-modal attention. Li et al. [113] and Yu et al. [206] include the object tags, attributes, and relations extracted from the source image as input to ease the text-image alignment. Although they have only tested on neural images, similar methods can be applied to scientific document representation by extracting features of scientific charts, such as the data's trend (rise and fall), minimum and maximum.

Bilinear pooling, which originated in the computer vision community, is another common method for combining features of different modalities [172]. It combines every pair of multimodal input channels. However, while powerful, this results in an explosion in feature dimensionality and can result in overfitting and poor performance. To address this, multimodal low-rank bilinear pooling [57, 96] targets to solve this dimensionality problem by means of a low-dimensional approximation. Furthermore, multiple attempts have been made to integrate the attention mechanism with multimodal bilinear pooling to improve representation effectiveness. For example, Kim et al. [95] introduces bilinear attention networks to find bilinear attention distributions of a bimodal input before proceeding with low-rank bilinear pooling.

Tables contain text content as cells, thus can be linearized and concatenated with the textual inputs [191] for multimodal representation. However, although such a method requires no feature fusion, it neglects the 2-D structure of tables. Table parsers [78] solve this problem by including the cell locations in the modeling process, but how to fuse the extracted table features and textual features effectively becomes a challenge.

Text + Layouts. Layout is an important visual component of a scientific document. As the relative positions of document components (text blocks/figures/tables) significantly contribute to the document's semantics, incorporating layout information improves the multimodal representation of a visually-rich document. Layout information is included as a 2-D position of a document component to build its representation. For example, Xu et al. [194] build a layout-aware transformer with text tokens and their 2-D positions as input. To utilize the visual information, Li et al. [112] slice the document images into rows, representing text and images with their textual/visual features and positional encodings, and then combine the two modalities using a cross-modality encoder. Wu et al. [187] divide the documents into blocks to accommodate the various size of document components, and represent the location of the block with its top-left and bottom-right coordinates. They design a two-level structure to encode each block, then aggregate block-level representations, and pretrain using hierarchical objectives. For scientific documents that contains multiple pages, Pramanik et al. [147] utilize a special transfomer [14] network architecture, in which the attention mechanism scales linearly with the sequence length, to encode the multimodal information. The model uses page images and page numbers as inputs, in addition to token features. The system proposed by Huang et al. [83] is an extension to [194], with an emphasis on alignment between word and image patches. It projects the patches and word tokens linearly to generate contextualized vector representations. The model is pretrained to learn cross-modal alignment between words and patches.

Recommendations: The commonality in such representation works is that both image and visual input content can be represented by extracting their features first separately and then to apply some fusion means them to generate a joint representation. However, because of the nature of the images (natural images vs. scientific figures) and length of the document (short paragraphs vs. long documents), practitioners should make modeling adjustments based on the characteristics of the target scientific documents. Possible adjustments include replacing the generic CNN image encoder with encoders specifically designed for scientific figures [23, 24, 25], and using transformers designed to represent long documents [147]. Representing the table with a pretrained table parser has yet to be tested in representing the tabular and textual content of a scientific document. While including layout information in the modeling process improves the representation results, additionally incorporating discourse structure (§ 3.2) to better represent multimodal documents remains a fertile area to be explored, in our view. Similarly, the scope of what constitutes multimodality — in terms of both inlined and external multimodal artifacts — can be expanded to more holistically represent scientific discourse for downstream tasks that may benefit from this auxiliary pathways for computing relevance and similarities.

5.2 Incorporating Multimodality

Multimodal SDP tasks can operate either in a trans-modality setting (where information in single input modality is transferred or translated into another) or in a true multimodal setting (where multiple modalities are represented as input). We examine two representative multimodal SDP tasks — captioning and summarization — along these lines and conclude with our perspective of the logical future directions in multimodal SDP.

Captioning. Internal artifacts such as figures and tables include text captions to emphasize aspects of the data [94, 102]. For visual information contained in charts, captioning improves the recall and comprehension of the data by drawing attention to some aspect of the underlying information [75]. Due to the importance of captions in understanding scholarly document content and the presence of title-only captions, automated caption generators can assist in paper writing. Generating captions is largely a trans-modality task, with the task of appropriate encoding of the figure or table to decode an appropriate output textual caption.

Trans-modality figure captioning. To automatically generate captions for figures, a common process first parses the figure with a CNN decoder, and then generates caption text based on the figure content using an RNN decoder [23, 24, 25]. Attending to the labels during decoding enables the model to utilize the text contained in the figures [23]. The sequence-level training with reinforcement learning improves the generation of long captions by directly optimizing over the evaluation metrics [25]. Such end-to-end models rely largely on the quality and quantity of the training data to automatically learn the caption generation process. To improve the understanding of the caption generation problem, Qian et al. [150]'s corpus-based study collected and analyzed human-written captions, finding that a caption often consists of a set of *caption units*, which refer to an enumerated set of clause types (e.g., number and labels of items, pairwise comparisons) describing specific types of information in the figure. As such, to automatically generate figure captions, a system should first generate caption units according to the figures, then stitch the units together using diverse patterns to form captions. Specifically, Qian et al. [151]'s later model attends to both the visual information and the metadata information of the input figure to generate accurate caption units.

Trans-modality table captioning. To generate table captions, in contrast, prior work first serializes table data, then processes them with a sequence encoder using pre-trained language model [132]. Such models are fine-tuned for numerical reasoning, but currently are still unable to generate high-fidelity text. Therefore, Suadaa et al. [165] introduce the use of the copy mechanism during fine-tuning to reduce the occurrence of generated phrases that are irrelevant to the table.

Multimodal figure and table captioning. Recent work on captioning in the SDP context consider text beyond the input figure and table. This form of the task upgrades the task to a true multimodal task, as the input includes relevant body text as well. Besides being based on the target figure or table, Yang et al. [196] consider texts relevant to the figure for caption generation. The proposed system comprises of a figure parsing module and a module to identify the sentences in the body text related to the figure, fusing both together using a rule-based algorithm.

Xu et al. [191] tackles a similar task, but for tables instead of figures. They use a BM25-based retrieval model to find sentences related to the text of the table to augment the table input in a concentative method. They feed the resultant text sequence to pretrained language models to output a caption, or extend a partially generated one in an auto-regressive manner.

Multimodal Summarization. Unlike the captioning task, multimodal summarization must incorporate multimodal sources, such as image and video as input [175] alongside the text, so is a true multimodal application.

Accounting for tables and figures contained in a scholarly document improves the quality of textual summaries[175]. As figures and tables highlight important messages of a paper, end-to-end multimodal summarization focuses on critical aspects during text summary generation. For example, Li et al. [108] and Chen and Zhuge [27] design a modality-based attention mechanism to summarize a sentence–image pair by attending to different part of the images and text units simultaneously. Similarly, the multimodal selective gate network proposed by Li et al. [109] considers both textual input as well as visual features at the global, grid and object levels.

Accounting for multimodal input also enables multimodal summary output; i.e., summaries containing both text and non-text modalities. A multimodal summary containing both text and figures provides the user with additional visual information and ease the comprehension process as compared to a text-only summary [213]. In addition to text, summarization incorporating multimodal output system includes an image selector to include images into the summary based on evidence from both the input text and images. Zhu et al. [213] propose a multimodal attention model to jointly generate summary text and select the image that has the highest visual coverage [108]. Chen and Zhuge [27, 28] follow a similar framework in their two works, also considering both the visual features and the ordering of those components in generating visual representations. They accomplish this by encoding the visual features with a CNN model, followed by an RNN model for the ordering representation. To further improve the multimodal representation, Zhang et al. [211] proposed a unified approach to handle multimodal input by incorporating both multimodality and knowledge distillation in representation.

Yet with these advances, there is still much room for improvement. Performance of multimodal summarizers degrade in the face of two connected problems: modality bias and insufficient data (resources). The problem of modality bias occurs when models are optimized to generate good text summaries but ignore the quality of the selected images during training [215]. Zhu et al. [215] design a multimodal objective function to include the image selection loss in the training objective. Ye et al. [204] further show that adding a residual connection to the model effectively alleviates modality bias. Methods to improve the performance in low-resource settings include utilizing cross-domain dataset and incorporating unsupervised training. One way to incorporate pre-trained models proposed by Yamamoto et al. [195] is to build visual-based image selector and language-based text generator as two individual modules. While this setup allows both modules to be trained with other datasets, the improvement brought by fusing the other modality in each task is neglected. To address this, Zhu et al. [214] project the embeddings of the textual and visual information to a common semantic space and estimate the text–image similarity using a pre-trained captioning model. They then build a graph-based extractive summarizer where the similarity (as edges) between different information units (as nodes) is unsupervisedly trained.

Summaries can also take special forms in scholarly documents: as posters and presentation slides, which are themselves multimodal views of a scholarly document. A poster contains several panels, each covering a section of the source document and containing both text and figures. A poster generator needs to identify the important sections and extract the salient content for the panel. The poster generation model by Xu and Wan [192, 193] learns to predict important text and figures simultaneously based on their section-aware representations. The extracted panels are then used to fill the predefined template to form the poster. Similarly, a presentation slide deck can inherit the structure of the source document. It involves multimodal summary extraction and paraphrasing of the source content to be more concise [55]. Furthermore, the identified information units need to be arranged into the layout of presentation slides. The system proposed by Fu et al. [55] contains modularized components for each of these sub-tasks, and trains the model end-to-end using a multi-objective loss function covering both content selection and layout. In contrast, Sun et al. [168] features a human-in-the-loop, leting users input slide titles to retrieve the most relevant sentences and figures to those titles. They use a question answering paradigm, where the slide titles function as questions and retrieve sentences as the source passage for paraphrasing into concise forms.

Recommendations: In reviewing the prior work in multimodal SDP, we notice an alignment gap. The practical work in multimodal SDP applications we reviewed here (§ 5.2) have largely neglected the importance of appropriate multimodal representation (§ 5.1). This suggests a significant pathway forward: where the performance of multimodal applications — such as our reviewed ones of captioning and summarization — improve by careful and appropriate choice of their multimodal representation. We note that in work related to both applications, there has also been a focus on appropriate selection: body text is long and laden with discourse signals that help localize relevant text. Summarization is an application also featured in Challenges #1 and #2, such that both discourse structure and citation networks can be thought of as modalities themselves. This suggests that fusing representations across all three challenges may benefit summarization performance.

In the larger context, we believe that the scope of what multimodality is defined as in current scientific document processing is still limiting. As SDP matures, a broadened scope that includes the entire spectrum of modalities — *inlined, internal* and *external* artefacts — will afford new and interesting artefact-centric (i.e., equation, dataset, and grant funding mention indexing) functionalities.

6 Resources

The proliferation of neural methods in SDP has spurred the creation of many new large-scale datasets, and tools to train and deploy modern neural network models and make them accessible to downstream practitioners.

Tool	SDP tasks	Neural support	PDF processing	Experimentation	End-user application	Code availability
CERMINE [174]	Extract reference string, Citation string parsing		✓			✓
GROBID [116]	Header parsing, Reference extraction and parsing, Citation context recognition, Citation string parsing, Logical structure recovery		✓			
ParsCit [37]	Logical structure recovery, Header parsing, Citation string parsing		✓		✓	✓
Neural-ParsCit [148]	Citation string parsing	✓				✓
Scienceparse[4]	Logical structure recovery	✓	✓			✓
SciSpacy [140]	Biomedical named entity recognition, Biomedical named entity linking				✓	✓
SciWING [153]	Logical structure recovery, Header parsing, Citation string parsing, Citation intent classification, Clinical notes parsing	✓	✓	✓	✓	✓

Table 3 SDP related tools and frameworks, characterized based on a) **SDP tasks**: which tasks are directly supported, b) **Neural support**: whether they include neural network based models, c) **PDF processing**: whether they support processing PDF pipelines, d) **Experimentation**: whether they allow researchers to experiment with the machine learning models, e) **End-user application**: whether they provide mechanisms to deploy models, and f) **Code Availability**: The availablity of the project code. The url to the code is listed in the reference for those available.

6.1 Tools and Frameworks

Many recent generic NLP tools, like AllenNLP [59], FLAIR [5], Texar [81], Spacy[5] have been built on top of deep learning frameworks like TensorFlow [1] and Pytorch [145]. However, they do not cater to the specific needs of SDP. We consider tools and frameworks specific to SDP and use the following dimensions to compare them as shown in the columns of Table 3: applicable SDP tasks, neural support, PDF support, experimentation support and end-user applications. We make the following observations:

1. **SDP tasks**: To be useful in the real world, we need to combine solutions from different SDP tasks. An ideal package for practitioners would provision coverage for a large number of SDP tasks. From the survey, we find that GROBID and SciWING [116, 153] lead on this front, while others like Neural-ParsCit[148] and Science Parse deal with specific tasks like citation string parsing and logical structure recovery. Future tools and frameworks should be engineered to faciltate the easy addition of SDP tasks to provide better coverage.
2. **Neural support**: There still is a significant gap in making recent neural methodologies easy to apply for downstream domains such as SDP. We call for the community to continue to add to or support frameworks to help researchers and practitioners obtain the fruits of these benefits. Some teams have chosen to reinvent their frameworks anew. Neural-ParsCit [148], Science Parse and SciWING [153] provide native access to neural network pre-trained models for end-users. Others, such as GROBID have taken the path of retrofitting neural-network methods into their frameworks. GROBID [116] also provides end-users with neural network based methods.
3. **PDF processing**: Most of the scientific documents are available in the Portable Document Format (PDF). Tools that provide end-to-end PDF processing pipelines improves ease of use. Otherwise, preprocessing to obtain the needed input representation from the PDF is first required. Given the integral nature of processing PDFs for SDP tasks — for example, to identify discourse structure — tools like CERMINE [174], GROBID [116], ParsCit [37] and SciWING [153] include mechanisms to directly ingest PDFs. However, most of these tools treat PDFs as text-only documents, and do not consider the computer vision and multi-modal methods that have shown improved

[5] https://www.spacy.io

performance in other application domains. This current weakness is a key area we feel for improvement that the community should prioritize.

4. **Experimentation**: Many neural network models are a precise combination of different modules and embeddings; even applications to related SDP tasks and domains often exhibit lower performance due to the need for extensive tuning. Both SDP researchers and practitioners would benefit from allowing experimentation of the model, to tune the embeddings, and to train and fine-tune the models on their own application domain's data. Except for SciWING [153], most tools in our list and frameworks do not allow for such experimentation.
5. **End-user application**: Most SDP practitioners are interested in obtaining results for downstream use, treating the software as off-the-shelf solutions with easy to use, and may not be concerned in tuning the models. Providing end-user applications are essential to achieve this. Most of the tools like Science Parse and Neural-ParsCit [148] are the result of research efforts, so do not focus on providing end-user applications. Even trivial interfaces provided by ParsCit [37], SciSpacy [140] and SciWING [153] are useful. Further, tools like scite.ai[6] and Semantic Scholar that integrate the end results of research into a user-friendly website is already proving beneficial for scholar–practitioners.

6.2 Datasets

We also summarize the recent efforts to provide large-scale datasets addressing challenges in SDP. Again, while there are limitless possible SDP tasks, it is instructive to limit our discussion to tasks (and their associated resources) centrally related to our three challenges. For these reasons, in this survey, we examine SDP datasets targeting the two tasks of **summarization** and **citation intent classification**.

6.2.1 Summarization

Progress in practical summarization has taken large strides in the past few years, spurred by the aforementioned techniques to train large-scale models and the availability of data [44, 156]. Neural network approaches have particularly targeted abstractive forms of summarization, but require large amounts of data. For news article summarization, corpora such as CNN/Daily Mail [76] and NewsRoom [65] are examples. However, providing large-scale human annotated summaries datasets for scientific documents is expensive. They are long documents and require human subjects to have a complete understanding of the document before creating a good summary. In recent years, the SDP community has invested efforts to create larger datasets [20, 21], for which we provide an overview here:

1. **SciTLDR** [16]: introduces a TLDR ("Too long; didn't read") dataset for 1–2 sentence summaries of scientific publications, suitable for presenting as search snippets. They obtain TLDRs from peer reviews culled from OpenReview[7] — a platform for authors and independent reviewers. Such extreme forms of summarization for scientific publications are reminiscent of their news article counterparts in XSUM [138].
2. **SciSummNet** [202]: creates a human-annotated scientific summarization dataset by asking the annotators to read the abstract of a paper and all the citing sentences and form a summary. This dataset contains 1,000 paper–summary pairs.
3. **BIGPATENT** [159]: Like scientific documents, patents are long-form, conventionally structured documents. BIGPATENT uses the abstract of a patent as its summary. It includes more than a million document–summary pairs.
4. **TALKSUMM** [106]: considers the video recordings of paper presentations in NLP and machine learning based conferences. They align the sentences in the video transcripts with sentences from the paper using Hidden Markov Models and use these sentences as extractive summaries. Models trained on the automatically extracted dataset are as performant as on human annotated data. The dataset contains 1,700 paper–summary pairs.
5. **CSPubSum** [35]: use the highlight statements provided by authors of ScienceDirect[8] publications as human annotated extractive summaries. Further, they also extend the gold summary sentences, by considering the top sentences that have a high ROUGE-L [114] scores. The dataset consists of summaries for more than 10,000 papers.
6. **MS^2** [45]: Medical studies are also a form of scientific documents that have started to garner attention. The Multi-Document Summarization of Medical Studies (MS^2) dataset features medical articles and summaries to investigate the summarization and sense-making of possibly contradictory biomedical articles. It also provides annotation of key clinical medical metadata in the form of patient, intervention, comparison and outcome (PICO [82]) keyphrases. Such

[6] www.scite.ai

[7] `https://openreview.net/`

[8] `www.sciencedirect.com`

Dataset Name	Long/Short	Size (# summaries)	Single/Multiple	Abstractive/Extractive	Dataset Availability
SciTLDR [16]	Short	1700	Single	Abstractive	✓
SciSummNet [202]	Long	1K	Single	Abstractive	✓
BIGPATENT [159]	Long	1M	Single	Abstractive	✓
TALKSUMM [106]	Long	1.7K	Single	Extractive	✓
CSPubSum [35]	Long	10K	Single	Extractive	✓
MS^2 [45]	Long	470K	Multi	Abstractive	✓
arXiv [33]	Long	215K	Single	Abstractive	✓
PubMed [33]	Long	133K	Single	Abstractive	✓
LaySumm [20]	Long	572	Single	Abstractive	✓
LongSumm [20]	Long	2.2K	Single	both	✓

Table 4 Large-scale SDP summarization datasets. We compare them based on: a) **Long/Short**: we consider any summary greater than a mean length of 50 words a *long* summary, b) **Size**: The number of document summary pairs, c) **Single/Multiple**, d) **Abstractive/Extractive**: Whether the summaries are extractive or abstractive. **Dataset Availability** refers to availablity of the paper's dataset; where available, URLs are listed in the respective reference in the bibliography.

datasets represent the recent wave towards multitask and joint learning, where two SDP tasks can profitably benefit each other.

7. **ArXiv and Pubmed datasets** [33]: introduces the ArXiv and the PubMed datasets for summarization. They consider the abstracts as the summaries and the entire scientific article as the source. Since the abstracts are written by humans, the summaries are considered abstractive. The ArXiv dataset contains more than 200,000 articles while the PubMed dataset – contains more than 133,000 articles, making these datasets some of the largest available.
8. **LongSumm** [20]: Most of the summarization datasets include summaries that are a few hundred words. A longer summary that enables one to explore the research article — such as research weblogs — are lacking. LongSumm aims to tackle this challenge by contributing 1,705 extractive summaries from the previous [106] dataset, also accompanied by abstractive summaries originating from research blogs that contain on average of 30 or more sentences. It formed one of the SDP shared tasks in 2021, chalking up 18 submissions to its three tasks.
9. **LaySumm** [20]: To make science more accessible to non-technical readers, LaySumm aims to produce summaries that explain the overarching goal and impact of a scientific document. This dataset contains around 570 human written lay summaries of scientific documents and the corresponding abstract and full text are made available.

Table 4 compares the summarization datasets among a few salient dimensions. 1) ***Long/Short***: The SDP community aims to provide long summaries compared to other domains such as news articles. This is because scientific articles need to include multiple facets to facilitate reasonable comprehension. For example, the summary should help readers understand the context, the problems and gaps in the literature, and the scope of the current article in solving it. 2) ***Size***: News article summaries contain millions of document–summary pairs, compared to a few thousand supervised pairs for scientific articles. Large transformer model that are in vogue for summarization [107, 152, 210] require large-scale data for training. Although recent efforts have been directed in curating large-scale datasets for summarization [16], continued efforts will benefit scientific document summarization. 3) ***Single/Multiple***: Most works consider single documents for summarization, and do not consider the citations or the citing article for summarization. With the increasingly large network of scientific articles and accompanying citations, capturing salient information from multiple related documents provides an alternative form of summarization that is unique to scientific documents. Such summaries will place the scientific document in an appropriate context with respect to other works. 4) ***Extractive/Abstractive***: While the recent application of neural models have improved abstractive summarization, key issues for scientific article summarization remain unaddressed. For example, there is no guarantee of the factuality of the generated summaries. We note that recent methods take

Dataset Name	Label Space	Size	Labeled Citation Context	Dataset Availability
Abu-Jbara *et al.* [3]	Criticizing, Comparison, Using, Substantiating, Bias, Other	3.5K	✓	
Cohan *et al.* [32]	Background, Method, Comparison	11K		✓
Jurgens *et al.* [87]	Background, Motivation, Using, Extending, Compare and Contrast, Future	1.9K		✓
Lauscher *et al.* [103]	Background, Motivation, Using, Extending, Similarities, Difference, Future	12.6K	✓	✓
Nambanoor Kunnath et al. [136]	Background, Motivation, Using, Extending, Compare and Contrast, Future	4K		✓
Su *et al.* [164]	Weakness, Comparison and Contrast, Positive, Negative	1.4K		✓
Valenzuela *et al.* [176]	Related Work, Comparison, Using, Extending	465		

Table 5 Datasets for Citation Intent Classification. We compare them based on a) **Label Space**: The set of labels used to classify the citation, b) **Size**: The number of citations annotated with the citation intent, c) **Labeled Citation Context**: Indicates whether the dataset also provides annotation for the context of the citation. **Dataset Availability**: Where the dataset is publically available, its hyperlink is listed in its bibliographic reference.

steps to address this [29, 137] and going forward, a summary's fidelity will remain an important criterion in evaluating summaries.

6.2.2 Citation Intent Classification

Analyzing the citations made for a scientific publication can help researchers understand how the scientific community perceives a scientific article. Online scientific platforms such as Semantic Scholar and scite.ai[6] have deployed such analyses to aid researchers. Table 5 details the efforts to curate datasets for such citation analysis. We compare them with respect to their a) **Label Space**: Intents annotated by the dataset, b) **Size**, and c) **Labeled Citation Context**: whether the citation context is also annotated.

1. **Label Space**: Datasets use disparate labels, and some feature a hierarchical taxonomy. Building upon previous works, some datasets break a label into a more fine-grained label. For example, Lauscher et al. [103] breaks down compare and contrast further into *Similarities* and *Differences*. Cohan et al. [32] compose many categories defined by Jurgens et al. [87] into the *Background* section. The common reasons cited by authors are ease of use or observations without any evidence. The recent C3 shared tasks also labeled citation influence (importance), appealing to solutions featuring joint predictions of both tasks Nambanoor Kunnath et al. [136].
Unfortunately, many of these works do not build upon others, fragmenting the datasets and making fair comparison difficult. We suggest that the community rally around a common, simple label space, but which can be extended for discipline-specific needs.
2. **Size**: The largest of datasets has close to 13,000 labeled citation contexts. Compared to well-known text classification datasets, this scale is at least a magnitude smaller, insufficient for building high-performing neural models. This highlights the challenge to employ neural models for this task. Techniques that require special treatment to handle lack of data, are yet to be applied for citation intent classification. Provided that annotating large-scale datasets has been difficult up to now, we see an outlook where such problems are addressed not with additiona data, but with data-efficient techniques.
3. **Labeled Citation Context**: While it has been repeatedly shown that citation context improves citation intent classification, currently only two datasets also annotate the citation context [3, 103]. Abu-Jbara and Radev [4] propose to identify citation context automatically, which is not tackled by current neural network methods. Apart from curation efforts to label citation context, automatically identifying the

context should be part of the pipeline for citation intent classification.

7 Conclusion

We have given an overview of the challenges offered by scientific document processing (SDP). In addition to these key challenges, we conclude by discussing the limitations of our survey and our view of future trends and outlook for scholarly document processing.

Survey scope limitations: Our view of SDP, as envisioned in this article, is still limited towards work related more closely to the natural language processing (NLP) community: with intrinsic document and citation processing. This is consistent with the vision of a large subset within the digital library community; e.g., [124, 134]. Our intent was to provide a comprehensive viewpoint on this scope; accordingly, our discussion of the tasks, terminology, and datasets is limited to the scenarios mentioned here.

However, SDP can be construed more broadly to account for relevance to any textual processing involved in scholarly documents, including its multimodality — visual and aural [144]), its auxiliary artifacts (that is, data and software [18, 77] or controlled metadata [125, 141]) — and its archiving and preservation [54]. A key limitation of our work is that we have purposefully omitted discussion of issues related to these other areas, and leave the generalization to future scholars. Importantly, we believe that the three challenges we have identified are still entirely relevant to all such research and application areas.

Future Outlook: Are there other key issues and contexts that the SDP community needs to consider in the upcoming years of development, within the scope of the three challenges described here? Emphatically, yes!

To conclude, we offer our point of view on five critical issues that the community should address.

1. **Lack of Deep Learning Tools for SDP**: The proliferation of modern learning methods within the NLP community has had a deep and lasting impact. However, the use of such advances within the SDP community has been difficult [64]. To facilitate sharing the advances of deep learning on SDP tasks, there is a clear need for easy-to-use tools. In § 6.1 we saw that neural network methods are integrated in a few frameworks. But they are siloed, address only a limited number of tasks, and have a steep learning curve. To enable researchers to adopt modern methods in SDP, there is a dire need for tools that provide pretrained models and allow easy experimentation with minimal changes, parallel to general NLP open-source projects.
2. **Minimal Supervised Data**: Abundant data is one of the reasons for the success of large-scale neural networks. Annotating data to obtain large-scale supervised data is an expensive venture that requires domain expertise, money, and effort. Therefore, researchers continue to work in makingneural networks effective in low-data scenarios. Pre-training and fine-tuning domain-specific transformer models is currently the most effective and popular way to make modern neural methods work in new domains [70]. With pre-training becoming ubitiquous in NLP applications, more studies such as Gupta et al. [69] that examine its effects of SDP tasks are needed. Data augmentation is another popular technique to increase the size of the data set [6, 62, 128, 155]. Alternate learning mechanisms such as active learning [92, 186] that reduces annotation costs and multitask learning that results in more generalized models can also alleviate data scarcity problems [38, 63, 163, 183, 217]. On the other hand, large language models can be used as a tool to alleviate the burden of annotating by producing annotations in a semi-automatic manner [48]. Working with minimal supervised data is an important endeavor for machine learning in general. Solutions developed to this problem should be adopted by SDP, and generally help the scientific community.
3. **Knowledge Driven Methods**: Although advances in deep learning generative methods produce fluent language, it suffers from hallucinations and other text degeneration problems [179]. Also, it does not ensure that the generated text is factual and that the important terms from the source document are not missed — critical for scientific documents. Summarization has especially seen an influx of work that ensures that the generated summaries are factual (in the non-SDP context) and ensures that important facts are not omitted from the summaries [121]. Another important area where factuality is important is question answering [158] and fake science detection [104]. Knowledge bases, which are mostly manually-curated concise representations of real-world knowledge, can help ensure that neural networks outputs are factually correct. Many modern neural network methods inject side information from knowledge bases into their architectures [115]; for example, for summarization systems [66, 212] and for question answering [158]. Integrating knowledge graphs into neural network representations is an interesting recent endeavor, which will continue

to gain importance in the future and is especially important for scientific document processing.

4. **Understanding Long and Multiple Documents**: Scientific documents are long and complex documents that pose major challenges to the current neural network architecture. Recent efforts have taken different approaches to improve the number of tokens analyzed and produced by models [14, 207]. Additionally, to assist in understanding a scientific field and automating literature reviews, the SDP community needs to research work that goes beyond single documents. Neural network representations can consider other related documents. In this vein, Cohan et al. [34] introduce the SPECTER model, which uses contrastive learning to learn similar representations for closely related documents. This concept presents multiple challenges to the current neural network paradigm, such as increasing computational time and cost. These tasks serve as an appropriate testbed to understand the advances on these fronts, as with even longer documents, such as work on theses and dissertations [53].
5. **For Humans, By Humans**: SDP aims to make science faster and better. The importance of SDP has increased with the collaborative work on COVID-19 [93, 131]. The community can use these urgent necessities to motivate work to further streamline common research goals so that researchers can spend more quality research time working on difficult cognitive tasks. One way to achieve this is to help researchers perform literature review [182], write scientific papers efficiently [181], recommend papers [52], produce automated summaries [16], understand the context of a problem, and write critiques of a paper. Progress in SDP is of little use if such human-assistive technologies are not adopted outside research. Digital libraries need to deploy such works to enable researchers to be more efficient.
   Automation allows efficiency, but the SDP community also needs to engineer suitable work and evaluation processes for check and balances of the quality of automation. Advanced automation poses difficulty for evaluation as tasks become harder to judge, especially with respect to recall (missing key work or insights). Authors of productionized SDP tools have an advantage for understanding and creating insights that further their own research, possibly creating imbalances that unfairly discriminate against junior and non-native researchers. This is a recognized problem in general language technology deployments and is being actively addressed in the NLP community through a series of workshops [10, 39, 177]. We need to address these fundamental of diversity and inclusion issues before they become endemic problems in scientific research.

Acknowledgements

The authors would like to thank the editorial team of the *International Journal on Digital Libraries* for their tireless work in helping shepherd and carefully proofread the manuscript. We thank them for their patience with our revision process and deeply thank them in their critical involvement in shaping this article.

The team authors would also like to thank the many group members of our research team whose readings and insights also helped to shape the manuscript's direction.

This work was partially funded by the Singapore Ministry of Education's grant entitled "Scholarly Document Information Extraction" (MOE T1 251RES20237).

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
2. Abu-Jbara, A., Ezra, J., and Radev, D. (2013a). Purpose and polarity of citation: Towards nlp-based bibliometrics. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 596–606.
3. Abu-Jbara, A., Ezra, J., and Radev, D. (2013b). Purpose and polarity of citation: Towards NLP-based bibliometrics. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 596–606, Atlanta, Georgia. Association for Computational Linguistics.
4. Abu-Jbara, A. and Radev, D. R. (2012). Reference scope identification in citing sentences. In *HLT-NAACL*.
5. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., and Vollgraf, R. (2019). FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.
6. Amin-Nejad, A., Ive, J., and Velupillai, S. (2020). Exploring transformer text generation for medical dataset augmentation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4699–4708, Marseille, France. European Language Resources Association.
7. An, C., Zhong, M., Chen, Y., Wang, D., Qiu, X., and Huang, X. (2021). Enhancing scientific papers summarization with citation graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12498–12506.
8. Anastasopoulos, A., Kumar, S., and Liao, H. (2019). Neural language modeling with visual features. *arXiv preprint arXiv:1903.02930.*
9. André, J., Furuta, R., Furuta, R. K., and Quint, V. (1989). *Structured documents*, volume 2. Cambridge University Press.
10. Axelrod, A., Yang, D., Cunha, R., Shaikh, S., and Waseem, Z., editors (2019). *Proceedings of the 2019 Workshop on Widening NLP*, Florence, Italy. Association for Computational Linguistics.
11. Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.
12. Bai, X., Wang, M., Lee, I., Yang, Z., Kong, X., and Xia, F. (2019). Scientific paper recommendation: A survey. *IEEE Access*, 7:9324–9339.
13. Banerjee, S., Sanyal, D. K., Chattopadhyay, S., Bhowmick, P. K., and Das, P. P. (2020). Segmenting scientific abstracts into discourse categories: A deep learning-based approach for sparse labeled data. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, JCDL '20, page 429–432, New York, NY, USA. Association for Computing Machinery.
14. Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv:2004.05150.*
15. Bulut, B., Gündoğan, E., Kaya, B., Alhajj, R., and Kaya, M. (2020). User's research interests based paper recommendation system: A deep learning approach. In *Putting Social Media and Networking Data in Practice for Education, Planning, Prediction and Recommendation*, pages 117–130. Springer.
16. Cachola, I., Lo, K., Cohan, A., and Weld, D. (2020). TLDR: Extreme summarization of scientific documents. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777, Online. Association for Computational Linguistics. Dataset available at `https://github.com/allenai/scitldr`.
17. Cai, X., Han, J., and Yang, L. (2018). Generative adversarial network based heterogeneous bibliographic network representation for personalized citation recommendation. In *Thirty-second AAAI conference on artificial intelligence.*
18. Candela, L., Castelli, D., Manghi, P., and Callaghan, S. (2017). On research data publishing. *International Journal on Digital Libraries*, 18(2):73–75.
19. Caruana, R. (1997). Multitask learning. *Mach. Learn.*, 28(1):41–75.
20. Chandrasekaran, M. K., Feigenblat, G., Hovy, E., Ravichander, A., Shmueli-Scheuer, M., and de Waard, A. (2020). Overview and insights from the shared tasks at scholarly document processing 2020: CL-SciSumm, LaySumm and LongSumm. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 214–224, Online. Association for Computational Linguistics.

21. Chandrasekaran, M. K., Yasunaga, M., Radev, D., Freitag, D., and Kan, M.-Y. (2019). Overview and results: Cl-scisumm shared task 2019. In *In Proceedings of Joint Workshop on Bibliometric-enhanced Information Retrieval and NLP for Digital Libraries (BIRNDL 2019)*.

22. Chaturvedi, R., ., S., Dhani, J. S., Joshi, A., Khanna, A., Tomar, N., Duari, S., Khurana, A., and Bhatnagar, V. (2020). Divide and conquer: From complexity to simplicity for lay summarization. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 344–355, Online. Association for Computational Linguistics. Code available at `https://github.com/anuragjoshi3519/laysumm20`.

23. Chen, C., Zhang, R., Kim, S., Cohen, S., Yu, T., Rossi, R., and Bunescu, R. (2019a). Neural caption generation over figures. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, pages 482–485.

24. Chen, C., Zhang, R., Koh, E., Kim, S., Cohen, S., and Rossi, R. (2020). Figure captioning with relation maps for reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1537–1545.

25. Chen, C., Zhang, R., Koh, E., Kim, S., Cohen, S., Yu, T., Rossi, R., and Bunescu, R. (2019b). Figure captioning with reasoning and sequence-level training. *arXiv preprint arXiv:1906.02850*.

26. Chen, J. and Zhuge, H. (2014). Summarization of scientific documents by detecting common facts in citations. *Future Generation Computer Systems*, 32:246–252.

27. Chen, J. and Zhuge, H. (2018). Abstractive text-image summarization using multi-modal attentional hierarchical rnn. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4046–4056.

28. Chen, J. and Zhuge, H. (2019). Extractive summarization of documents with images based on multi-modal rnn. *Future Generation Computer Systems*, 99:186–196.

29. Chen, S., Zhang, F., Sone, K., and Roth, D. (2021). Improving faithfulness in abstractive summarization with contrast candidate generation and selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5935–5941, Online. Association for Computational Linguistics.

30. Chen, W., Gao, Y., Zhang, J., King, I., and Lyu, M. R. (2019c). Title-guided encoding for keyphrase generation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6268–6275. AAAI Press.

31. Clark, C. and Divvala, S. (2016). Pdffigures 2.0: Mining figures from research papers. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pages 143–152.

32. Cohan, A., Ammar, W., van Zuylen, M., and Cady, F. (2019). Structural scaffolds for citation intent classification in scientific publications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Dataset available at `https://github.com/allenai/scicite`.

33. Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., and Goharian, N. (2018). A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics. Dataset available at `https://github.com/acohan/long-summarization`.

34. Cohan, A., Feldman, S., Beltagy, I., Downey, D., and Weld, D. (2020). SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

35. Collins, E., Augenstein, I., and Riedel, S. (2017). A supervised approach to extractive summarisation of scientific papers. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 195–205, Vancouver, Canada. Association for Computational Linguistics. Dataset available at `https://github.com/EdCo95/scientific-paper-summarisation`.

36. Conneau, A., Kruszewski, G., Lample, G., Barrault, L., and Baroni, M. (2018). What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

37. Councill, I., Giles, C. L., and Kan, M.-Y. (2008). ParsCit: an open-source CRF reference string parsing package. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA). Code available at `http://wing.comp.nus.edu.sg/parsCit/`.
38. Crichton, G., Pyysalo, S., Chiu, B., and Korhonen, A. (2017). A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*, 18(1).
39. Cunha, R., Shaikh, S., Varis, E., Georgi, R., Tsai, A., Anastasopoulos, A., and Chandu, K. R., editors (2020). *Proceedings of the The Fourth Widening Natural Language Processing Workshop*, Seattle, USA. Association for Computational Linguistics.
40. Dai, T., Zhu, L., Wang, Y., and Carley, K. M. (2019). Attentive stacked denoising autoencoder with bi-lstm for personalized context-aware citation recommendation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:553–568.
41. Dasigi, P., Burns, G. A., Hovy, E., and de Waard, A. (2017). Experiment segmentation in scientific discourse as clause-level structured prediction using recurrent neural networks. *arXiv preprint arXiv:1702.05398*. Code available at `https://github.com/edvisees/sciDT`.
42. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.
43. Denk, T. I. and Reisswig, C. (2019). Bertgrid: Contextualized embedding for 2d document representation and understanding. *CoRR*, abs/1909.04948.
44. Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.
45. DeYoung, J., Beltagy, I., van Zuylen, M., Kuehl, B., and Wang, L. L. (2021). MSˆ2: Multi-document summarization of medical studies. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7494–7513, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. Dataset and code available at `https://github.com/allenai/ms2`.
46. dos Santos, E. A., Peroni, S., and Mucheroni, M. L. (2022). The way we cite: Common metadata used across disciplines for defining bibliographic references. In *International Conference on Theory and Practice of Digital Libraries*, pages 120–132. Springer.
47. Ebesu, T. and Fang, Y. (2017). Neural citation network for context-aware citation recommendation. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*, pages 1093–1096.
48. El-Ebshihy, A. (2021). Semi-automatic labelling of scientific articles using deep learning to enlarge benchmark data for scientific summarization. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2707, New York, NY, USA. Association for Computing Machinery.
49. Eskenazi, S., Gomez-Krämer, P., and Ogier, J.-M. (2017). A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognition*, 64:1 – 14.
50. Färber, M. and Jatowt, A. (2020). Citation recommendation: approaches and datasets. *International Journal on Digital Libraries.*
51. Färber, M. and Sampath, A. (2020). Hybridcite: A hybrid model for context-aware citation recommendation. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, pages 117–126.
52. Färber, M., Thiemann, A., and Jatowt, A. (2018). Citewerts: A system combining cite-worthiness with citation recommendation. In *European conference on information retrieval*, pages 815–819. Springer.
53. Fox, E. A., Eaton, J. L., McMillan, G., Kipp, N. A., Weiss, L., Arce, E., and Guyer, S. (1996). National digital library of theses and dissertations. *D-Lib Magazine, September.*
54. Fox, E. A., Klein, M., and Xie, Z. (2018). Guest editors' introduction to the special issue on web archiving. *International Journal on Digital Libraries*, 19(1):1–2.
55. Fu, T.-J., Wang, W. Y., McDuff, D., and Song, Y. (2022). Doc2ppt: Automatic presentation slides generation from scientific documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 634–642.
56. Fu, T.-Y., Lei, Z., and Lee, W.-C. (2016). Modeling time lags in citation networks. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 865–870. IEEE.
57. Fukui, A., Park, D. H., Yang, D., Rohrbach, A., Darrell, T., and Rohrbach, M. (2016). Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*.
58. Ganguly, S. and Pudi, V. (2017). Paper2vec: Combining graph and text information for scientific paper representation. In *European Conference on Information Retrieval*, pages 383–395. Springer.
59. Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N. F., Peters, M., Schmitz, M., and

Zettlemoyer, L. (2018). AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

60. Ghosh, S., Das, D., and Chakraborty, T. (2016). Determining sentiment in citation text and analyzing its impact on the proposed ranking index. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 292–306. Springer.
61. Gidiotis, A. and Tsoumakas, G. (2020). A divide-and-conquer approach to the summarization of academic articles. *ArXiv*, abs/2004.06190.
62. Giorgi, J. M. and Bader, G. D. (2018). Transfer learning for biomedical named entity recognition with neural networks. *Bioinformatics*, 34(23):4087–4094.
63. Giorgi, J. M. and Bader, G. D. (2019). Towards reliable named entity recognition in the biomedical domain. *Bioinformatics*, 36(1):280–286.
64. Grennan, M. and Beel, J. (2020). Synthetic vs. real reference strings for citation parsing, and the importance of re-training and out-of-sample data for meaningful evaluations: Experiments with grobid, giant and cora. *ArXiv*, abs/2004.10410.
65. Grusky, M., Naaman, M., and Artzi, Y. (2018). Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.
66. Gunel, B., Zhu, C., Zeng, M., and Huang, X. (2020). Mind the facts: Knowledge-boosted coherent abstractive text summarization. *CoRR*, abs/2006.15435.
67. Guo, L., Cai, X., Qin, H., Guo, Y., Li, F., and Tian, G. (2019). Citation recommendation with a content-sensitive deepwalk based approach. In *2019 international conference on data mining workshops (ICDMW)*, pages 538–543. IEEE.
68. Gupta, S. and Varma, V. (2017). Scientific article recommendation by using distributed representations of text and graph. In *Proceedings of the 26th international conference on world wide web companion*, pages 1267–1268.
69. Gupta, Y., Ammanamanchi, P. S., Bordia, S., Manoharan, A., Mittal, D., Pasunuru, R., Shrivastava, M., Singh, M., Bansal, M., and Jyothi, P. (2021). The effect of pretraining on extractive summarization for scientific documents. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 73–82, Online. Association for Computational Linguistics.
70. Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.
71. Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1025–1035.
72. Han, J., Song, Y., Zhao, W. X., Shi, S., and Zhang, H. (2018). hyperdoc2vec: Distributed representations of hypertext documents. *arXiv preprint arXiv:1805.03793*.
73. Hasan, K. S. and Ng, V. (2014). Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland. Association for Computational Linguistics.
74. He, F., Lee, W.-C., Fu, T.-Y., and Lei, Z. (2021). Cines: Explore citation network and event sequences for citation forecasting. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 798–807, New York, NY, USA. Association for Computing Machinery.
75. Hegarty, M. and Just, M.-A. (1993). Constructing mental models of machines from text and diagrams. *Journal of memory and language*, 32(6):717–742.
76. Hermann, K. M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1693–1701, Cambridge, MA, USA. MIT Press.
77. Hermon, S. and Niccolucci, F. (2021). Fair data and cultural heritage special issue editorial note. *International Journal on Digital Libraries*, 22(3):251–255.
78. Herzig, J., Nowak, P. K., Müller, T., Piccinno, F., and Eisenschlos, J. M. (2020). Tapas: Weakly supervised table parsing via pre-training. *arXiv preprint arXiv:2004.02349*.
79. Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.
80. Holm, A. N., Plank, B., Wright, D., and Augenstein, I. (2020). Longitudinal citation prediction using temporal graph neural networks. *arXiv preprint arXiv:2012.05742*.

81. Hu, Z., Shi, H., Tan, B., Wang, W., Yang, Z., Zhao, T., He, J., Qin, L., Wang, D., Ma, X., Liu, Z., Liang, X., Zhu, W., Sachan, D., and Xing, E. (2019). Texar: A modularized, versatile, and extensible toolkit for text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 159–164, Florence, Italy. Association for Computational Linguistics.

82. Huang, X., Lin, J. J., and Demner-Fushman, D. (2006). Evaluation of pico as a knowledge representation for clinical questions. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, pages 359–63.

83. Huang, Y., Lv, T., Cui, L., Lu, Y., and Wei, F. (2022). Layoutlmv3: Pre-training for document ai with unified text and image masking. *arXiv preprint arXiv:2204.08387*.

84. Jeong, C., Jang, S., Park, E., and Choi, S. (2020). A context-aware citation recommendation model with bert and graph convolutional networks. *Scientometrics*, 124(3):1907–1922.

85. Jin, D. and Szolovits, P. (2018). Hierarchical neural networks for sequential sentence classification in medical scientific abstracts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Code available at `https://github.com/jind11/HSLN-Joint-Sentence-Classification`.

86. Jin, M., Chang, H., Zhu, W., and Sojoudi, S. (2021). Power up! robust graph convolutional network via graph powering. In *35th AAAI Conference on Artificial Intelligence*.

87. Jurgens, D., Kumar, S., Hoover, R., McFarland, D., and Jurafsky, D. (2018). Measuring the evolution of a scientific field through citation frames. *Transactions of the Association for Computational Linguistics*, 6. Dataset available at `https://github.com/davidjurgens/citation-function`.

88. Kan, M.-Y., Luong, M.-T., and Nguyen, T. D. (2010). Logical structure recovery in scholarly articles with rich document features. *Int. J. Digit. Library Syst.*, 1(4):1–23.

89. Karpathy, A. and Fei-Fei, L. (2017). Deep visual-semantic alignments for generating image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):664–676.

90. Katti, A. R., Reisswig, C., Guder, C., Brarda, S., Bickel, S., Höhne, J., and Faddoul, J. B. (2018). Chargrid: Towards understanding 2D documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4459–4469, Brussels, Belgium. Association for Computational Linguistics.

91. Kerroumi, M., Sayem, O., and Shabou, A. (2020). Visualwordgrid: Information extraction from scanned documents using A multimodal approach. *CoRR*, abs/2010.02358.

92. Kholghi, M., Sitbon, L., Zuccon, G., and Nguyen, A. (2016). Active learning: a step towards automating medical concept extraction. *Journal of the American Medical Informatics Association : JAMIA*, 23 2:289–96.

93. Kieuvongngam, V., Tan, B., and Niu, Y. (2020). Automatic text summarization of covid-19 medical research articles using bert and gpt-2.

94. Kim, D. H., Setlur, V., and Agrawala, M. (2021). Towards understanding how readers integrate charts and captions: A case study with line charts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–11.

95. Kim, J.-H., Jun, J., and Zhang, B.-T. (2018). Bilinear attention networks. *Advances in neural information processing systems*, 31.

96. Kim, J.-H., On, K.-W., Lim, W., Kim, J., Ha, J.-W., and Zhang, B.-T. (2016). Hadamard product for low-rank bilinear pooling. *arXiv preprint arXiv:1610.04325*.

97. King, D., Downey, D., and Weld, D. S. (2020). High-precision extraction of emerging concepts from scientific literature. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1549–1552, New York, NY, USA. Association for Computing Machinery.

98. Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3294–3302.

99. Kobayashi, Y., Shindo, H., and Matsumoto, Y. (2019). Scientific article search system based on discourse facet representation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:9859–9860.

100. Kontoulis, C. G., Papagiannopoulou, E., and Tsoumakas, G. (2021). Keyphrase extraction from scientific articles via extractive summarization. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 49–55, Online. Association for Computational Linguistics. Code available at `https://github.com/intelligence-csd-auth-gr/keyphrase-extraction-via-summarization`.

101. Landhuis, E. (2016). Scientific literature: Information overload. *Nature*, 535(7612):457–458.

102. Large, A., Beheshti, J., Breuleux, A., and Renaud, A. (1995). Multimedia and comprehension: The rela-

tionship among text, animation, and captions. *Journal of the American society for information science*, 46(5):340–347.

103. Lauscher, A., Ko, B., Kuehl, B., Johnson, S., Jurgens, D., Cohan, A., and Lo, K. (2021). MultiCite: Modeling realistic citations requires moving beyond the single-sentence single-label setting. Dataset available at `https://github.com/allenai/multicite`.
104. Lay, P., Lentschat, M., and Labbe, C. (2022). Investigating the detection of tortured phrases in scientific literature. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 32–36, Gyeongju, Republic of Korea. Association for Computational Linguistics.
105. Lebanoff, L., Song, K., Dernoncourt, F., Kim, D. S., Kim, S., Chang, W., and Liu, F. (2019). Scoring sentence singletons and pairs for abstractive summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2175–2189, Florence, Italy. Association for Computational Linguistics.
106. Lev, G., Shmueli-Scheuer, M., Herzig, J., Jerbi, A., and Konopnicki, D. (2019). TalkSumm: A dataset and scalable annotation method for scientific paper summarization based on conference talks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2125–2131, Florence, Italy. Association for Computational Linguistics. Dataset available at `https://github.com/levguy/talksumm`.
107. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.
108. Li, H., Zhu, J., Liu, T., Zhang, J., Zong, C., et al. (2018). Multi-modal sentence summarization with modality attention and image filtering. In *IJCAI*, pages 4152–4158.
109. Li, H., Zhu, J., Zhang, J., He, X., and Zong, C. (2020a). Multimodal sentence summarization via multimodal selective encoding. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5655–5667.
110. Li, J., Peng, J., Liu, S., Weng, L., and Li, C. (2020b). Tsam: Temporal link prediction in directed networks based on self-attention mechanism. *arXiv preprint arXiv:2008.10021*.
111. Li, L., Zhu, Y., Xie, Y., Huang, Z., Liu, W., Li, X., and Liu, Y. (2019). Cist@ clscisumm-19: Automatic scientific paper summarization with citances and facets. In *BIRNDL@ SIGIR*, pages 196–207.
112. Li, P., Gu, J., Kuen, J., Morariu, V. I., Zhao, H., Jain, R., Manjunatha, V., and Liu, H. (2021). Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660.
113. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al. (2020c). Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer.
114. Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
115. Logan, R., Liu, N. F., Peters, M. E., Gardner, M., and Singh, S. (2019). Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971, Florence, Italy. Association for Computational Linguistics.
116. Lopez, P. (2009). GROBID: combining automatic bibliographic data recognition and term extraction for scholarship publications. In Agosti, M., Borbinha, J. L., Kapidakis, S., Papatheodorou, C., and Tsakonas, G., editors, *Research and Advanced Technology for Digital Libraries, 13th European Conference, ECDL 2009, Corfu, Greece, September 27 - October 2, 2009. Proceedings*, volume 5714 of *Lecture Notes in Computer Science*, pages 473–474. Springer.
117. Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.
118. Ma, X. and Wang, R. (2019). Personalized scientific paper recommendation based on heterogeneous graph representation. *IEEE Access*, 7:79887–79894.
119. Ma, X., Zhang, Y., and Zeng, J. (2019). Newly published scientific papers recommendation in heterogeneous information networks. *Mobile Networks and Applications*, 24(1):69–79.
120. Ma, Y., Tong, S., Liu, Y., Wu, L., Liu, Q., Chen, E., Tong, W., and Yan, Z. (2021). Enhanced representation learning for examination papers with hierarchical document structure. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2156–2160, New York, NY, USA. Association for Computing Machinery.

121. MacAvaney, S., Sotudeh, S., Cohan, A., Goharian, N., Talati, I. A., and Filice, R. W. (2019). Ontology-aware clinical abstractive summarization. *ArXiv*, abs/1905.05818.

122. Madisetty, S., Maurya, K. K., Aizawa, A., and Desarkar, M. S. (2020). A neural approach for detecting inline mathematical expressions from scientific documents. *Expert Systems*.

123. Makhzani, A., Shlens, J., Jaitly, N., and Goodfellow, I. J. (2015). Adversarial autoencoders. *CoRR*, abs/1511.05644.

124. Mayr, P., Frommholz, I., Cabanac, G., Chandrasekaran, M. K., Jaidka, K., Kan, M.-Y., and Wolfram, D. (2017). Introduction to the special issue on bibliometric-enhanced information retrieval and natural language processing for digital libraries (BIRNDL). *International Journal on Digital Libraries*, 19(2-3):107–111.

125. Mayr, P., Tudhope, D., Clarke, S. D., Zeng, M. L., and Lin, X. (2016). Recent applications of knowledge organization systems: introduction to a special issue. *International Journal on Digital Libraries*, 17(1):1–4.

126. Medić, Z. and Snajder, J. (2020). Improved local citation recommendation based on context enhanced with global information. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 97–103, Online. Association for Computational Linguistics. Code available at `https://github.com/zoranmedic/duallcr`.

127. Medo, M., Cimini, G., and Gualdi, S. (2011). Temporal effects in the growth of networks. *Physical review letters*, 107(23):238701.

128. Melamud, O. and Shivade, C. (2019). Towards automatic generation of shareable synthetic clinical notes using neural language models. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 35–45, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

129. Mendes, A., Narayan, S., Miranda, S., Marinho, Z., Martins, A. F. T., and Cohen, S. B. (2019). Jointly extracting and compressing documents with summary state representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3955–3966, Minneapolis, Minnesota. Association for Computational Linguistics.

130. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Burges, C. J. C., Bottou, L., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

131. Möller, T., Reina, A., Jayakumar, R., and Pietsch, M. (2020). COVID-QA: A question answering dataset for COVID-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

132. Moosavi, N. S., Rücklé, A., Roth, D., and Gurevych, I. (2021). Scigen: a dataset for reasoning-aware text generation from scientific tables. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

133. Mu, D., Guo, L., Cai, X., and Hao, F. (2017). Query-focused personalized citation recommendation with mutually reinforced ranking. *IEEE Access*, 6:3107–3119.

134. Mutschke, P., Scharnhorst, A., Belkin, N. J., Skupin, A., and Mayr, P. (2017). Guest editors' introduction to the special issue on knowledge maps and information retrieval (KMIR). *International Journal on Digital Libraries*, 18(1):1–3.

135. Mysore Gopinath, A. A., Wilson, S., and Sadeh, N. (2018). Supervised and unsupervised methods for robust separation of section titles and prose text in web documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 850–855, Brussels, Belgium. Association for Computational Linguistics.

136. Nambanoor Kunnath, S., Stauber, V., Wu, R., Pride, D., Botev, V., and Knoth, P. (2022). ACT2: A multi-disciplinary semi-structured dataset for importance and purpose classification of citations. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3398–3406, Marseille, France. European Language Resources Association. Dataset available at `https://github.com/oacore/ACT2`.

137. Nan, F., Nogueira dos Santos, C., Zhu, H., Ng, P., McKeown, K., Nallapati, R., Zhang, D., Wang, Z., Arnold, A. O., and Xiang, B. (2021). Improving factual consistency of abstractive summarization via question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6881–6894, Online. Association for Computational Linguistics.

138. Narayan, S., Cohen, S. B., and Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*,

pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

139. Narimatsu, H., Koyama, K., Dohsaka, K., Higashinaka, R., Minami, Y., and Taira, H. (2021). Task definition and integration for scientific-document writing support. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 18–26, Online. Association for Computational Linguistics. Code available at `https://github.com/citation-minami-lab/citation-dataset`.

140. Neumann, M., King, D., Beltagy, I., and Ammar, W. (2019). Scispacy: Fast and robust models for biomedical natural language processing. Code available at `https://allenai.github.io/scispacy/`.

141. Niccolucci, F. (2017). Extending, mapping, and focusing the CIDOC CRM. *International Journal on Digital Libraries*, 18(4):251–252.

142. Ohagi, M. and Aizawa, A. (2022). Pre-trained transformer-based citation context-aware citation network embeddings. In *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries*, JCDL '22, New York, NY, USA. Association for Computing Machinery.

143. Ostendorff, M., Blume, T., Ruas, T., Gipp, B., and Rehm, G. (2022). Specialized document embeddings for aspect-based similarity of research papers. In *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries*, JCDL '22, New York, NY, USA. Association for Computing Machinery.

144. Page, K. R. and Downie, J. S. (2019). Guest editors' introduction to the special issue on digital libraries for musicology. *International Journal on Digital Libraries*, 20(1):1–2.

145. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

146. Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

147. Pramanik, S., Mujumdar, S., and Patel, H. (2020). Towards a multi-modal, multi-task learning based pre-training framework for document representation learning. *arXiv preprint arXiv:2009.14457*.

148. Prasad, A., Kaur, M., and Kan, M.-Y. (2018). Neural parscit: a deep learning-based reference string parser. *International Journal on Digital Libraries*, 19:323–337. Code available at `https://github.com/WING-NUS/Neural-ParsCit`.

149. Qazvinian, V. and Radev, D. R. (2008). Scientific paper summarization using citation summary networks. *arXiv preprint arXiv:0807.1560*.

150. Qian, X., Koh, E., Du, F., Kim, S., and Chan, J. (2020). A formative study on designing accurate and natural figure captioning systems. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–8.

151. Qian, X., Koh, E., Du, F., Kim, S., Chan, J., Rossi, R. A., Malik, S., and Lee, T. Y. (2021). Generating accurate caption units for figure captioning. In *Proceedings of the Web Conference 2021*, pages 2792–2804.

152. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

153. Ramesh Kashyap, A. and Kan, M.-Y. (2020). SciWING– a software toolkit for scientific document processing. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 113–120, Online. Association for Computational Linguistics. Code available at `http://sciwing.io/`.

154. Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.

155. Romanov, A. and Shivade, C. (2018). Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium. Association for Computational Linguistics.

156. Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

157. See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume*

1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

158. Serban, I. V., García-Durán, A., Gulcehre, C., Ahn, S., Chandar, S., Courville, A., and Bengio, Y. (2016). Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 588–598, Berlin, Germany. Association for Computational Linguistics.
159. Sharma, E., Li, C., and Wang, L. (2019). BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics. Dataset available at `https://evasharma.github.io/bigpatent/`.
160. Siegel, N., Lourie, N., Power, R., and Ammar, W. (2018). Extracting scientific figures with distantly supervised neural networks. In Chen, J., Gonçalves, M. A., Allen, J. M., Fox, E. A., Kan, M., and Petras, V., editors, *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries, JCDL 2018, Fort Worth, TX, USA, June 03-07, 2018*, pages 223–232. ACM. Code available at `https://github.com/allenai/deepfigures-open`.
161. Soto, C. and Yoo, S. (2019). Visual detection with context for document layout analysis. 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)*, pages 3464–3470, Hong Kong, China. Association for Computational Linguistics.
162. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., and Dai, J. (2019a). Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530.*
163. Su, X., Prasad, A., Kan, M., and Sugiyama, K. (2019b). Neural multi-task learning for citation function and provenance. In Bonn, M., Wu, D., Downie, J. S., and Martaus, A., editors, *19th ACM/IEEE Joint Conference on Digital Libraries, JCDL 2019, Champaign, IL, USA, June 2-6, 2019*, pages 394–395. IEEE. Code available at `https://github.com/WING-NUS/citation_func_n_prov`.
164. Su, X., Prasad, A., Kan, M.-Y., and Sugiyama, K. (2018). Neural multi-task learning for citation function and provenance. *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 394–395. Dataset available at `https://github.com/WING-NUS/citation_func_n_prov`.
165. Suadaa, L. H., Kamigaito, H., Funakoshi, K., Okumura, M., and Takamura, H. (2021). Towards table-to-text generation with numerical reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1451–1465.
166. Subramani, N., Matton, A., Greaves, M., and Lam, A. (2020). A survey of deep learning approaches for ocr and document understanding.
167. Sugiyama, K. and Kan, M.-Y. (2010). Scholarly paper recommendation via user's recent research interests. In *Proceedings of the 10th annual joint conference on Digital libraries*, pages 29–38.
168. Sun, E., Hou, Y., Wang, D., Zhang, Y., and Wang, N. X. (2021). D2s: Document-to-slide generation via query-based text summarization. *arXiv preprint arXiv:2105.03664.*
169. Suppe, F. (1998). The structure of a scientific paper. *Philosophy of Science*, 65(3):381–405.
170. Tan, H. and Bansal, M. (2019). Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490.*
171. Tao, S., Shen, C., Zhu, L., and Dai, T. (2020). Svd-cnn: a convolutional neural network model with orthogonal constraints based on svd for context-aware citation recommendation. *Computational Intelligence and Neuroscience*, 2020.
172. Tenenbaum, J. B. and Freeman, W. T. (2000). Separating style and content with bilinear models. *Neural computation*, 12(6):1247–1283.
173. Teufel, S., Siddharthan, A., and Tidhar, D. (2006). Automatic classification of citation function. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 103–110.
174. Tkaczyk, D., Szostek, P., Fedoryszak, M., Dendek, P. J., and Bolikowski, L. (2015). Cermine: Automatic extraction of structured metadata from scientific literature. *Int. J. Doc. Anal. Recognit.*, 18(4):317–335. Code available at `https://github.com/CeON/CERMINE`.
175. UzZaman, N., Bigham, J. P., and Allen, J. F. (2011). Multimodal summarization of complex sentences. In *Proceedings of the 16th international conference on Intelligent user interfaces*, pages 43–52.
176. Valenzuela, M., Ha, V., and Etzioni, O. (2015). Identifying meaningful citations. In *AAAI Workshop: Scholarly Big Data.*
177. Varis, E., Georgi, R., Tsai, A., Anastasopoulos, A., Chandu, K., Schofield, X., Ranathunga, S., Lepp, H., and Ghosal, T., editors (2021). *Proceedings of the Fifth Workshop on Widening Natural Language Processing*, Punta Cana, Dominican Republic. Association for Computational Linguistics.
178. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2018a). Glue: A multi-task

benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

179. Wang, C. and Sennrich, R. (2020). On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3544–3552, Online. Association for Computational Linguistics.
180. Wang, J., Zhu, L., Dai, T., and Wang, Y. (2020a). Deep memory network with bi-lstm for personalized context-aware citation recommendation. *Neurocomputing*, 410:103–113.
181. Wang, Q., Huang, L., Jiang, Z., Knight, K., Ji, H., Bansal, M., and Luan, Y. (2019). PaperRobot: Incremental draft generation of scientific ideas. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1980–1991, Florence, Italy. Association for Computational Linguistics.
182. Wang, Q., Zeng, Q., Huang, L., Knight, K., Ji, H., and Rajani, N. F. (2020b). ReviewRobot: Explainable paper review generation based on knowledge synthesis. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 384–397, Dublin, Ireland. Association for Computational Linguistics.
183. Wang, X., Zhang, Y., Ren, X., Zhang, Y., Zitnik, M., Shang, J., Langlotz, C., and Han, J. (2018b). Cross-type biomedical named entity recognition with deep multi-task learning. *Bioinformatics*, 35(10):1745–1752.
184. Wang, Z. and Liu, J.-C. (2020). Pdf2latex: A deep learning system to convert mathematical documents from pdf to latex. In *Proceedings of the ACM Symposium on Document Engineering 2020*, pages 1–10.
185. Wei, M., He, Y., and Zhang, Q. (2020). Robust layout-aware ie for visually rich documents with pre-trained language models. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2367–2376, New York, NY, USA. Association for Computing Machinery.
186. Wei, Q., Chen, Y., Salimi, M., Denny, J. C., Mei, Q., Lasko, T. A., Chen, Q., Wu, S., Franklin, A., Cohen, T., and Xu, H. (2019). Cost-aware active learning for named entity recognition in clinical text. *Journal of the American Medical Informatics Association*, 26(11):1314–1322.
187. Wu, T.-L., Li, C., Zhang, M., Chen, T., Hombaiah, S. A., and Bendersky, M. (2021). Lampret: Layout-aware multimodal pretraining for document understanding. *arXiv preprint arXiv:2104.08405*.
188. Xiao, W. and Carenini, G. (2019a). Extractive summarization of long documents by combining global and local context. 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)*, pages 3011–3021, Hong Kong, China. Association for Computational Linguistics. Code available at `https://github.com/Wendy-Xiao/Extsumm_local_global_context`.
189. Xiao, W. and Carenini, G. (2019b). Extractive summarization of long documents by combining global and local context. *arXiv preprint arXiv:1909.08089*.
190. Xu, J. and Durrett, G. (2019). Neural extractive text summarization with syntactic compression. 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)*, pages 3292–3303, Hong Kong, China. Association for Computational Linguistics.
191. Xu, J. H., Shinden, K., and Kato, M. P. (2021). Table caption generation in scholarly documents leveraging pre-trained language models. In *2021 IEEE 10th Global Conference on Consumer Electronics (GCCE)*, pages 963–966. IEEE.
192. Xu, S. and Wan, X. (2021). Neural content extraction for poster generation of scientific papers. *arXiv preprint arXiv:2112.08550*.
193. Xu, S. and Wan, X. (2022). Posterbot: A system for generating posters of scientific papers with neural models.
194. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., and 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*, pages 1192–1200.
195. Yamamoto, S., Fukuhara, Y., Suzuki, R., Morishima, S., and Kataoka, H. (2019). Automatic paper summary generation from visual and textual information. In *Eleventh International Conference on Machine Vision (ICMV 2018)*, volume 11041, pages 214–221. SPIE.
196. Yang, J., Zhang, D., Dong, G., and Peng, J. (2020). Figure captioning in scholarly literatures to augment search results. In *32nd International Conference on Scientific and Statistical Database Management*, pages 1–4.
197. Yang, L., Zhang, Z., Cai, X., and Dai, T. (2019). Attention-based personalized encoder-decoder model for local citation recommendation. *Computational intelligence and neuroscience*, 2019.
198. Yang, L., Zheng, Y., Cai, X., Dai, H., Mu, D., Guo, L., and Dai, T. (2018). A lstm based model for personalized context-aware citation recommendation. *IEEE access*, 6:59618–59627.

199. Yang, X., Yümer, M. E., Asente, P., Kraley, M., Kifer, D., and Giles, C. L. (2017). Learning to extract semantic structure from documents using multimodal fully convolutional neural network. *CoRR*, abs/1706.02337.
200. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.
201. Yasunaga, M., Kasai, J., Zhang, R., Fabbri, A. R., Li, I., Friedman, D., and Radev, D. R. (2019a). Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7386–7393.
202. Yasunaga, M., Kasai, J., Zhang, R., Fabbri, A. R., Li, I., Friedman, D., and Radev, D. R. (2019b). Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. *CoRR*, abs/1909.01716. Dataset available at `https://michiyasunaga.github.io/projects/scisumm_net/`.
203. Ye, H. and Wang, L. (2018). Semi-supervised learning for neural keyphrase generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4142–4153, Brussels, Belgium. Association for Computational Linguistics.
204. Ye, X., Yue, Z., and Liu, R. (2021). Mba: A multimodal bilinear attention model with residual connection for abstractive multimodal summarization. In *Journal of Physics: Conference Series*, volume 1856, page 012070. IOP Publishing.
205. Yin, J. and Li, X. (2017). Personalized citation recommendation via convolutional neural networks. In *Asia-Pacific web (APWeb) and web-age information management (WAIM) joint conference on web and big data*, pages 285–293. Springer.
206. Yu, F., Tang, J., Yin, W., Sun, Y., Tian, H., Wu, H., and Wang, H. (2021). Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3208–3216.
207. Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. (2020). Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*.
208. Zerva, C., Nghiem, M.-Q., Nguyen, N. T., and Ananiadou, S. (2019). Nactem-uom@ cl-scisumm 2019. In *BIRNDL@ SIGIR*, pages 167–180.
209. Zhang, C., Zhang, Z., Li, J., Liu, Q., and Zhu, H. (2021). Ctnr: Compress-then-reconstruct approach for multimodal abstractive summarization. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.
210. Zhang, J., Zhao, Y., Saleh, M., and Liu, P. (2020). PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.
211. Zhang, Z., Meng, X., Wang, Y., Jiang, X., Liu, Q., and Yang, Z. (2022). Unims: A unified framework for multimodal summarization with knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11757–11764.
212. Zhu, C., Hinthorn, W., Xu, R., kai Zeng, Q., Zeng, M., Huang, X., and Jiang, M. (2020a). Boosting factual correctness of abstractive summarization with knowledge graph. *ArXiv*, abs/2003.08612.
213. Zhu, J., Li, H., Liu, T., Zhou, Y., Zhang, J., and Zong, C. (2018). Msmo: Multimodal summarization with multimodal output. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 4154–4164.
214. Zhu, J., Xiang, L., Zhou, Y., Zhang, J., and Zong, C. (2021). Graph-based multimodal ranking models for multimodal summarization. *Transactions on Asian and Low-Resource Language Information Processing*, 20(4):1–21.
215. Zhu, J., Zhou, Y., Zhang, J., Li, H., Zong, C., and Li, C. (2020b). Multimodal summarization with guidance of multimodal reference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9749–9756.
216. Zhu, X.-D., Turney, P. D., Lemire, D., and Vellino, A. (2015). Measuring academic influence: Not all citations are equal. *JASIST*, 66:408–427.
217. Zuo, M. and Zhang, Y. (2020). Dataset-aware multi-task learning approaches for biomedical named entity recognition. *Bioinformatics*, 36(15):4331–4338.