Explaining Relationships Between Scientific Documents

Kelvin Luu[1,*] **Xinyi Wu**[1,*] **Rik Koncel-Kedziorski**[1]
Kyle Lo[2] **Isabel Cachola**[3] **Noah A. Smith**[1,2]
[1]University of Washington [2]Allen Institute for AI [3]Johns Hopkins University
{kellu,nasmith}@cs.washington.edu, {xywu,kedzior}@uw.edu,
kylel@allenai.org, icachola@cs.jhu.edu

Abstract

We address the task of explaining relationships between two scientific documents using natural language text. This task requires modeling the complex content of long technical documents, deducing a relationship between these documents, and expressing that relationship in text. Successful solutions can help improve researcher efficiency in search and review. In this paper, we operationalize this task by using citing sentences as a proxy. We establish a large dataset for our task. We pretrain a large language model to serve as the foundation for autoregressive approaches to the task. We explore the impact of taking different views on the two documents, including the use of dense representations extracted with scientific information extraction systems. We provide extensive automatic and human evaluations which show the promise of such models, and make clear the challenges for future work.

1 Introduction

The output of the world's scientists doubles roughly every nine years (Bornmann and Mutz, 2015). Consequently, researchers must devote significant energy to quickly understand how a new piece of research fits with a rapidly changing research landscape.

Several lines of research seek to reduce this burden on scientists. Citation recommendation systems suggest references to relevant published work (McNee et al., 2002; Bhagavatula et al., 2018). Intent classification systems help determine the type and importance of a citation in a work (Valenzuela et al., 2015; Cohan et al., 2019). Summarization systems aim to help researchers more quickly understand the basic ideas in a piece of research (Cohan and Goharian, 2015; Yasunaga et al., 2019). We draw inspiration from these works as well as broader challenges like explaining the connection between concurrent works or relating a new paper to those a reader is already familiar with.

Automatically describing inter-document relationships could decrease the time researchers devote to literature review. For instance, explanations for a new paper can be personalized to a particular reader by relating the new work to ones they have read before. Further, such technology could be incorporated into writing assistance systems to help less experienced or non-native writers better articulate the connection between their work and prior art. Additionally, users of citation recommendation systems can benefit from natural language explanations of recommendation system choices.

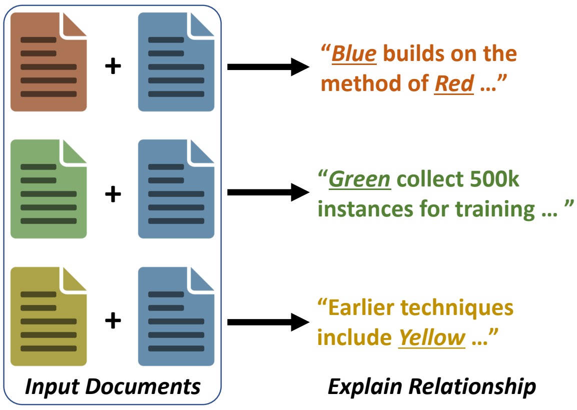
In addition to the utility of this task to scientists, it presents several interesting technical challenges. These include effectively representing the important information in a document, generating from a long-tailed technical vocabulary, and expressing the variety of connections between related scientific papers. Figure 1 illustrates how the same document is described differently in relation to different documents.

In this paper we use citing sentences to operationalize the problem of generating natural language explanations of the relationships between two scientific papers. Authors, when citing other work, oftentimes describe how their work relates to the cited work. To this end, we use in-text citation sentences as a naturally occurring proxy explanations for how two documents relate to each other. However, we generate such sentences from general representations of document content rather than the specific in-text locations where these sentences occur, as this task formulation can better facilitate the applications described above.

We approximate the explanation objective by having a GPT2 language model generate sentences containing citations given a pair of documents.

[*] Equal contribution.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics
and the 11th International Joint Conference on Natural Language Processing, pages 2130–2144
August 1–6, 2021. ©2021 Association for Computational Linguistics



+
"*Blue* builds on the method of *Red* ..."
"*Green* collect 500k instances for training ... "
"Earlier techniques include *Yellow* ..."
Input Documents
Explain Relationship

Figure 1: Given two scientific documents, the goal is to write the sentence describing the specific relationship between them. For a given document (in blue above), the output will vary depending the content of the other. (This image is best viewed in color.)

This approach relies on providing dense but informative representations of documents to use as conditioning context for the generation model. We explore the use of sentence-based contexts as input including document abstracts, introductions, and sampled sentences from the full document; we find that using introductions and abstracts works well. Finally, we improve our model's performance on automated metrics by using informative entities and terms to both construct dense input and rank the output relationship explanations.

In addition to standard automatic metrics, we perform human evaluations of technical outputs with a pool of annotators. In this work, we describe a series of stages of model development, each with its own experiments that, together, informed the task and our series of solutions.

Our contributions include: a novel dataset for the relationship explanation task; a domain-adapted GPT2 we release for left-to-right language modeling of scientific text; the SciGen model for describing document relationships; and an extensive expert evaluation and analysis of machine generated technical text.[1]

2 Related Work

The current work builds on recent research in scientific document understanding, including citation recommendation, intent categorization, and scientific document summarization. Citation recommendation systems suggest related works given a document or a span of text (McNee et al., 2002; Nallapati et al., 2008; Bhagavatula et al., 2018). Recently, researchers have sought to categorize citations using various ontologies of citation intents. Teufel et al. (2006) develop an annotation scheme and corresponding classification model for citation functions. Valenzuela et al. (2015) seek to discern "highly influential" citations from others. Jurgens et al. (2018) use six categories including "motivation," "uses," and "future work" among others. Cohan et al. (2019) condense this ontology to just three: "background," "method," and "result comparison." Intent classification can identify relationships between documents; our relationship explanation task extends this in two ways. First, data-driven freeform generation can express a wider array of relationships compared to a manually-defined label set. Further, our task framework could be used to describe relationships between works which do not actually cite each other, such as contemporaneous works. Unlike categorization techniques, we require no task-specific annotated data as we supervise with citing sentences that are readily available in scientific documents. In practice, citation classification is used to assist in suggesting relevant works to researchers; our work complements this goal by providing rationales for the recommendation and furthering progress toward explainable AI.

Our work is also connected to a long history of research on summarizing scientific documents (Luhn, 1958; Paice, 1980). Work in this area has mostly used used abstracts or peer reviews as targets (Cachola et al., 2020; Cohan et al., 2018; Jaidka et al., 2017). In particular, Pilault et al. (2020) show that using a simple extractive summary as input for abstractive summarization of scholarly texts work well. Researchers have also used citing sentences as part of the input for summarization, recognizing the explanatory power of these texts (Nakov et al., 2004; Cohan and Goharian, 2017; Yasunaga et al., 2019). Ours is the first work to focus on learning to express the specific relationship between two documents from such sentences.

The closest work to our own is Xing et al. (2020), who pilot a task of in-line citation generation. Their goal is a model which can insert a citing sentence into a particular context within a document. Our work, on the other hand, aims to learn from citing sentences how to describe general relationships

[1] https://github.com/Kel-Lu/SciGen

between documents independent of particular in-document contexts. While the Xing et al. (2020) method may facilitate writing assistance, our task has applications in search and summarization. Because our task does not rely on a specific location in a document where the citation will go, solutions can be used at scale to provide users with general explanations of document relationships.

Our models rely heavily on recent advances in transfer learning in NLP. Large pretrained models such as BERT (Devlin et al., 2018) and GPT2 (Radford et al., 2019) have made strong advances on a number of tasks (Wang et al., 2019). It has also been shown that pretraining these models on domain-specific data further improves results on domain-specific tasks (Beltagy et al., 2019; Lee et al., 2019). In this work, we apply that methodology by adding a pretraining phase on in-domain data before finetuning a GPT2 model toward the explanation generation task. A key challenge when using pretrained language models for document-level tasks is how to select document content to fit within the limited context window of the model, which is a major focus of our work.

3 Task Overview

We aim to generate an explanation: a natural language sentence which expresses how one document relates to another. Explicit examples of such sentences are nontrivial to find in corpora, especially when annotation for a highly technical task is expensive. To this end, we use in-text citations in a scientific document to prior work as proxies for relationship explanations. We use these citing sentences as partial supervision for our task, and refer to them as "explanations."[2]

We distinguish one document as the *principal* document, from which we will draw explanations that reference the *cited* document. Let t denote an explanation drawn from principal document S, and S' denote S without t. Then let

$$P(t \mid S', C) \tag{1}$$

be the probability of t given S' and the cited document C. A good generation technique should maximize this probability across a large number of $\langle t, S, C\rangle$ triples, so that at inference time the model is able to generate a sentence t^* which accurately describes the relationship between new documents $\hat{S}$ and $\hat{C}$.

	total	average/doc.
documents	154K	–
tokens	813M	5.3K
unique tokens	7.1M	1.3K
explanations	622K	4.0

Table 1: Dataset statistics, total and per document.

Optimizing Equation 1 is made easier by modern representation learning. Pretrained neural language models like GPT2 have shown strong performance when generating sentences conditioned on a context. However, existing implementations of GPT2 limit the context window to 512 or 1024 tokens, far smaller than scientific documents. In this work, we explore ways to represent the documents' content for use with language models.

Data We use English-language computer science articles and annotation from S2ORC dataset (Lo et al., 2020). S2ORC is a large citation graph which includes full texts of 8.1 million scientific documents. We use 154K connected computer science articles, from which we extract 622K explanations with a single reference that link back to other documents in our corpus. We omit any sentences that cite more than one reference. We hold 5000 sentences for each of the validation and test sets. Detailed statistics can be found in Table 1. Information on dataset construction can be found in Appendix B.

Evaluation The most appropriate evaluation metric for this and many text generation tasks is human judgment by potential users of the system. Evaluating explanations of the relationships between scientific documents requires human judges with scientific expertise whose time and effort can be costly. While collecting human judgments in technical domains is relatively rare, we believe it to be an important step in evaluating our systems for this task. Thus, we conduct thorough human evaluations and analyses with expert judges. We make use of both larger scale expert evaluations yielding hundreds of judgements as well as smaller scale, deeper evaluations where we can effect a higher degree of quality control over fewer datapoints. Further, we make use of intermediate human evaluations in the development of our models, and supplement these evaluations with automatic metrics — BLEU

[2] Future work might seek to filter or systematically alter in-text citations to be more explanation-like, without otherwise changing our approach.

(Papineni et al., 2002) and ROUGE (Lin, 2004) that are established in other generation tasks.

4 Models

We develop several models for explaining document relationships. Following current work in neural text generation, we finetune the predictions of a large pretrained language model to our task (Section 4.1). In order to bring the language model into the scientific text domain, we do additional language model pretraining over full scientific texts. We also investigate approximate nearest neighbor methods to retrieve plausible human-authored explanations from the training data as a baseline (Section 4.2).

4.1 Neural Text Generation

Recent work has shown that finetuning large pretrained language models to text generation tasks yields strong results (Zellers et al., 2019). To this end, we construct SciGen, a model based on GPT2 (Radford et al., 2019), a transformer model trained on 40GB of internet text with a left-to-right language modeling objective (Vaswani et al., 2017). We do so by finetuning the predictions of the language model to generate explanations using different expressions of the principal and cited document as context.

To finetune GPT2 architectures for text generation, it is typical to concatenate the conditioning context $X = x_1 \dots x_n$ and target sentence $Y = y_1 \dots y_m$ with a special separator token ξ^y. To adapt this technique to our task, we construct the conditioning context X from the principal and cited documents and use the explanation as Y. We take j tokens from principal document $s_1, \dots, s_j$ along with k tokens from the cited document $c_1, \dots, c_k$ (which tokens to draw from the two documents is an independent variable that we explore experimentally). We then condition the generation of explanation Y on $X = s_1, \dots, s_j, \xi^x, c_1, \dots, c_k$, where ξ^x is a token used to indicate the end of the principal document. SciGen is trained to predict the explanation one token at a time as described above. More details on training can be found in Appendix A.

At inference time, the model is provided with an unseen principal/cited document pair. An explanation of their relationship is generated one token at a time using nucleus sampling (Holtzman et al., 2020). At timestep t, output token $\hat{y}_t$ is sampled from the top 90% of the distribution $P(\hat{y}_t \mid X, \xi^y, \hat{y}_1, \dots, \hat{y}_{t-1})$ (renormalized). The selected $\hat{y}_t$ is used to condition the prediction of subsequent tokens.

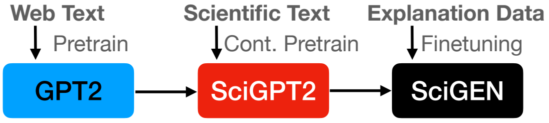


Figure 2: Overview of the construction of SciGen. We take the pretrained GPT2 and continue pretraining on scientific texts. We then finetune using data in Table 1.

Context The primary question we investigate with the SciGen model is what kind of input is best for describing the relationship between the principal and cited documents accurately and informatively. Since models based on GPT2 have a small context window relative to the length of scientific documents, we investigate the use of abstracts, introductions, or non-citing sentences sampled from throughout the document as conditioning context. The effectiveness and description of these approaches is described in Section 5. Based on our findings with sentence-based contexts and information retrieval systems, we then explore the possibility of representing the cited document text as a list of important concepts rather than fluent text, in Section 6.

Language Model Pretraining Prior work has shown that pretraining on in-domain data improves the performance of large language models on domain-specific tasks (Beltagy et al., 2019; Gururangan et al., 2020). Inspired by this, we continue pretraining the GPT2 model in the science domain to produce SciGPT2, which we use as the underlying language model for SciGen described above. SciGPT2 starts from the standard pretrained GPT2-base model and is trained for an additional 75k gradient updates at batch size of 64 (effectively a single epoch over 4.8 million abstracts and body paragraphs) with a language modeling objective. Figure 2 illustrates the process.

We observed significant improvements in the quality of SciGen outputs after replacing the underlying GPT2 language model with the domain-specific SciGPT2 model. We saw a perplexity improvement in a held-out set and, in informal inspections, qualitative improvements as well.

When using pretrained language models, text from task-specific test data cannot be guaranteed to be absent from the large task-independent corpora upon which these models are trained, which may improve model performance compared to models without this exposure. For the experiments described in this work, we train a version of SciGPT2 only on documents appearing in the training data, so that the principal documents and target sentences in the test data are unseen by the language model. We provide this and a full-corpus version of SciGPT2 as resources for future research.[3]

4.2 Retrieval with Approximate Nearest Neighbors

While neural text generation techniques have advanced significantly in recent years, their outputs are still inferior to human authored texts. For some tasks, it is better to retrieve a relevant human-authored text than to generate novel text automatically (Fan et al., 2018). Is this also the case when generating explanations?

To answer this question, we use an information retrieval (IR) baseline. We adapt an approximate nearest neighbor search algorithm to find similar pairs of documents. The basic search procedure is as follows: Given a test instance input (S, C) for principal S and cited document C, we find the set $\mathbf{N}_C$, the nearest neighbors to C in the training data. For each document N_C from $\mathbf{N}_C$, let $\mathbf{N}_S$ be the set of documents that cite N_C. This means that each $N_S \in \mathbf{N}_S$ contains at least one citing sentence t' which cites N_C. We use the t' associated with the (N_S, N_C) pair from the training which is closest to (S, C) as the explanation of their relationship.

We measure the closeness of two pairs of documents using the cosine distances between vector representations of their abstracts. The abstract of each document is encoded as a single dense vector by averaging the contextualized embeddings provided by the SciBERT model of Beltagy et al. (2019) and normalizing. The distance between (S, C) and neighbors (N_S, N_C) is computed as:

$$\alpha \cos(S, N_S) + \beta \cos(C, N_C) \qquad (2)$$

where α and β control the relative contribution of the two document similarities. We explore setting both α and β to 1, or tuning them to optimize BLEU on the validation data using MERT (Och, 2003).

5 Representing Documents with Sentence Selection

Methods for the related task of citation recommendation have made use of abstracts, which perhaps act as sufficient summaries of document content. Building on this, we represent the principal and cited documents with the first 450 tokens of either their abstracts, introductions, or sentences randomly sampled from throughout the full document.[4] In this section, we answer two questions: 1) do neural generation models with sentence-based context outperform the IR baseline and 2) does the type of sentence-based context (abstract, introduction, sampled) matter? We answer these questions by performing both automatic and human evaluations.

5.1 Automatic Evaluation

We compare the SciGen and IR systems using BLEU (Papineni et al., 2002) and ROUGE (specifically L; Lin, 2004). The "Sentence-based" rows of Table 3 show the test set performance of the IR system and the best SciGen models when provided with the different sentence-based input context combinations.[5] We assesss statistical significance as well by bootstrapping with 1000 samples in each of 100 iterations. We find that context *does* make a difference for SciGen, and that a slight but statistically significant performance improvement comes from using the introduction of the principal document rather than the abstract.[6] We do not, however, find enough evidence to reject the null hypothesis that any particular representation of the cited document's content (abstract, intro, or random sample) is sufficient.

We find that using the introduction of the principal document paired with the abstract of the cited document performs best, and so we select these for human evaluation. The IR systems perform well, obtaining slightly better scores in some settings. We choose the MERT-optimized version for human evaluation.

[3] https://github.com/Kel-Lu/SciGen

[4] We exclude any sentence with a citation from being sampled in all conditions. This context type is also only used for the cited document and not the principal document.

[5] The performance of our best SciGen models can be found in Table 3 and the automatic test set evaluations of all systems can be found in Appendix F.

[6] $p < 0.01$ after Bonferroni correction.

	Specific	Correct	S&C	*agr*
SciGen	72.3	64.0	55.0	*70.5*
IR	74.8	46.3	40.0	*77.5*
Gold	81.4	72.1	68.0	*83.8*
agreement	*69.8*	*71.4*	*63.1*	

Table 2: Human evaluation of SciGen (intro × abs) and IR (abs × abs) systems compared with gold explanations in percent. S&C represents those that were both specific and correct. All differences significant at $p < 0.01$ except SciGen vs. IR specific.

5.2 Human Evaluation

We conduct a human evaluation to determine, given a particular pair of principal and cited abstracts, how *correct* and *specific* the generated explanation of their relationship is. By "correct" we mean: does the explanation correctly express the factual relationship between the principal and cited documents? Because generic explanations such as "This work extends the ideas of Chomsky and Halle (1968)", while possibly factual, do not express a detailed understanding of the documents' relationship, we ask judges whether the explanation describes a specific relationship between the two works. An explanation can be specific even it is incorrect.

We compare the *principal intro* × *cited abs* SciGen setting against the tuned IR system. For calibration, we also elicit judgments for the gold explanations extracted from principal documents along with the correct principal and cited abstracts. In all three cases, we ensure that the principal document appeared in the ACL anthology to ensure annotator expertise. In total we solicit 37 NLP researchers and collect over 800 judgments, with over 100 for each system/quality dimension combination.

Further details of our evaluation can be found in Appendix D. We perform error analysis on these judgments as well as an additional study to validate human judgments; these are detailed in Appendix E and Appendix G. Table 2 shows the percentage of "yes" judgments versus the total of "yes" and "no" judgements for each system/quality combination, along with pairwise agreement rates.[7] Gold texts received the highest scores for all dimensions of text quality from the evaluators as well as the highest agreement rate. We can also see that IR systems tend to produce incorrect explanations more often than not.

The SciGen system performs quite well in this analysis, with a majority of outputs deemed correct. We observe a larger difference in specificity between SciGen and gold texts, indicating that SciGen, like many neural text generation systems, often generates vague and generic sentences. These generations tended to be vacuous such as "(Cited) This work is an extension of the paper." Specificity is key for future downstream applications such as automated literature review and will need to be improved for those tasks.

6 Using IE-Extracted Term Lists

Compared to the gold explanations, we found that our generated explanations miss important phrases such as unique model or dataset names and other lower-frequency terms; generally, they lacked specificity. The missing phrases typically appear in the cited document after the abstract and introduction.[8] Naïvely sampling from the full text does not capture them due to sparsity.

To address this issue, we explore more sophisticated information extraction (IE) techniques for constructing the conditioning context for SciGen. Recent work has shown that pretrained language models can adapt to disfluent inputs such as linearized trees and graphs (Ribeiro et al., 2020). Inspired by this, we investigate whether we can use lists of salient words and phrases to effect a dense representation of the cited document in the conditioning context. Specifically, we construct a list of document-specific terms using tf-idf to score unigrams and entities extracted with a state-of-the-art scientific NER system. The paradigm is illustrated in Figure 3.

Tf-idf Tf-idf is a measure of the frequency of a term in a document, normalized by the document frequency of that term. In our use, we calculate the tf-idf score for each unigram in the cited document. We keep the 100 highest scoring terms w_i sorted in descending order of scores. The terms of this list are concatenated with a special token ξ^{tf} to signal that this part of the input is structured as a list rather than conventional text. The resulting context $X^{tf} = w_1, \xi^{tf}, w_2, \xi^{tf}, ..., \xi^{tf}, w_{100}$ is used to represent the cited document to the SciGen model.

[7] That gold texts do not achieve perfect scores demonstrates a limitation of our evaluation setup, due in part to the fact that judgments are based on document abstracts rather than their full texts. We take steps to resolve this limitation in our subsequent analysis in Section 6.2.

[8] A quantitative analysis of this phenomenon is available in Appendix H.

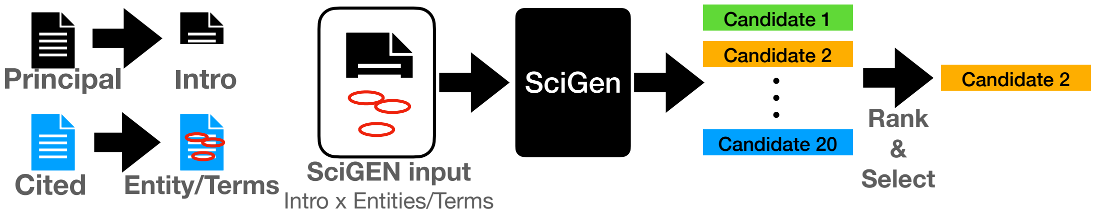


Figure 3: Overview of SciGEN using terms/entities. We generate a list of candidates and rank them according to mean reciprocal rank to the input entities.

Entities We extract entities from abstracts with the DyGIE++ information extraction framework (Wadden et al., 2019) using the model trained on SciERC (Luan et al., 2018), a dataset of scientific document abstracts with entity and relation annotations.[9] The extracted entities e_i from the cited document are sorted by their tf-idf scores compared to all entities in the corpus. As above, a special token ξ^e is used to concatenate entities and help the language model distinguish this list from conventional text. If there is additional room in the context window we append the unigrams with the highest tf-idf to the end of the listed entities until the window is full. In that case, the cited document context X^e is $e_1, \xi^e, e_2, ..., \xi^e, e_n, \xi^{tf}, w_1\xi^{tf}, ..., w_m$, where n is the number of entities and m is $100 - n$.

6.1 Entity-Based Ranking

Maynez et al. (2020) point out that summarization systems frequently struggle with factuality and generate hallucinations unfaithful to input documents. We observe this problem with some generated explanations as well: popular, topical terms like 'CNN' would appear in explanations of papers using LSTM models, for example. To combat hallucinations and promote factual accuracy we include a ranking mechanism that rewards generated explanations with higher coverage of important entities from the conditioning context.[10]

The process we use is as follows: first, we generate a large space of candidate explanations for a given input document pair from SciGEN via nucleus sampling. We then extract the entities from each candidate using the DyGIE++ IE system. Where possible, we match entities from the candidates with the entities extracted from the cited document. To account for textual variation between the explanations and the input documents, we use a similarity threshold to make soft alignments.[11] We then select the candidate that has the highest mean reciprocal rank of matched entities against the input as the explanation for this document pair.

6.2 Manual Analysis

We conducted a manual correctness analysis of the generated explanations from a sentence-based (intro × abs) and IE-based (intro × tfidf generate and rank) model. Two of the authors judged 50 datapoints from each system using a similar setup to that described in Section 5.2, but with the single objective of judging correctness on a 3-way scale: Correct; Too Vague (but not incorrect); and Incorrect. Additionally, the authors made use of the full text of the input documents to make decisions for cases where not enough information is available in the abstract. This resulted in a more accurate though much more time-consuming evaluation process compared to the previous evaluation. After judging all datapoints independently, the two authors discussed disagreements until a consensus was reached.

The results of this analysis are shown in Table 4. We see a slight increase in correctness with the IE-based model compared to the sentence-based model, though the difference is small.

6.3 Automatic Evaluation

The "IE-based" rows of Table 3 show the results of automatic metrics for the systems described in this Section. We find that these metrics improve significantly in the settings where the principal document is represented by its introduction and the cited document is represented either as a list of terms or entities, with a slight advantage for entities. The models conditioned on intro × tfidf context outperform all other sentence-based, retrieval, and IE-based models.

7 Discussion

Example system outputs for selected test datapoints are shown in Table 5. The first example illustrates

[9] We found relation annotations to be noisy on inspection.
[10] An oracle ranking is shown in Appendix I.
[11] We use difflib and a 0.7 similarity threshold for matching.

	Method	Context	BLEU	ACL-BLEU	Rouge-L
Sentence-based	SciGen	principal abs × cited abs	9.82	10.40	8.4
		principal intro × cited abs	9.92	11.22	8.7
		principal intro × cited intro	9.80	10.54	8.8
		principal intro × cited sampled	9.81	10.31	8.7
	IR	source abs × cited abs	9.93	10.50	9.7
		+MERT	10.23	10.29	9.8
IE-based	SciGen	principal intro × cited tfidf	13.17	16.75	12.0
		principal intro × cited entities	13.41	13.42	11.8
	+Ranking	principal intro × cited tfidf	13.50	15.10	12.3
		principal intro × cited entities	13.16	14.47	11.8

Table 3: Automatic test set evaluation of generated texts for a subset of our systems. ACL-BLEU denotes the BLEU scores of the subset of examples we use for human evaluation (see Section 5.2). The full results can be found in Appendix F.

	Correct	Vague	Incorrect
Sentence-based	11	7	32
IE-based	13	6	31

Table 4: Results of Manual Analysis

1	Principal:	*A Syllable-based Name Transliteration System*
	Cited:	*A Joint Source-Channel Model for Machine Transliteration*
	SciGen:	Following *Cited* , Chinese characters are considered as Pinyin sequence.
2	Principal:	*Recovering discourse relations: Varying influence of discourse adverbials*
	Cited:	*The Benefits of a Model of Annotation*
	SciGen:	The two text collections provided by *Cited* were used for training, and the other two text collections were used for evaluation.
3	Principal:	*Coreference Resolution for Swedish and German using Distant Supervision*
	Cited:	*Collective Cross-Document Relation Extraction Without Labelled Data*
	SciGen:	It is one of the most widely used distant supervision techniques and is inspired by techniques proposed by *Cited.*
4	Principal:	*Neural Text Generation in Stories Using Entity Representations as Context*
	Cited:	*Delete, Retrieve, Generate: A Simple Approach to Sentiment and Style Transfer*
	SciGen:	The authors of *Cited* proposed a model that combines neural generation with user interaction to create an object-centric reading experience.

Table 5: Example explanations. The given texts are the document titles and the SciGen outputs. In the last example, the two documents **do not** cite each other.

a case where the model identifies a correct relationship between the two documents. In this instance, they both use the pinyin representation for Chinese characters in their transliteration models.

Output 2 demonstrates a failure of the explanation generation system. The principal document deals with the topic of discourse relations, the automatic identification of which is a long-standing machine learning task. However, this particular document is an analysis paper, and does not involve any training.

Output 3 is an example of a “Too Vague (but not incorrect)” case from the analysis in Section 6.2. Here again the explanation generated by SciGen is topical, dealing with the concept of “distant supervision” that is key to both input documents. However, this sentence fails to capture the specific use that the principal makes of the research described in cited document.

The final example, output 4, showcases potential for our system to explain concurrent work. The generated text summarizes the *cited* and implies that *principal* will build on that work. However, selected papers are both concurrent generation papers published in the same venue and do not cite each other. This appears to be a weakness in using citation sentences as proxies for relationship explanations. Citations of contemporaneous work occur less frequently, so these types of sentences appear less often in training. Similarly, relationship explanations between papers with more distant connections (e.g., “multi-hop” in the citation graph) are missing in our training data.

In addition to missing some relationships, not all citation sentences are useful as explanations. As pointed out by other work, citation sentences can often be simple summaries of the cited work (Qazvinian and Radev, 2008; Cohan and Goharian, 2017). Alternatively, they can be too specific to be useful, as seen in Output 1, where a higher-level summary might be more useful. Future work could focus on curating better training sets for our task.

It is notable that the SciGen model usually outputs syntactically correct and topical explanations, even given the difficulty of the vocabulary in this domain. This is consistent with many recent findings using domain-specific language models.

The fluency and appropriateness of SciGen's generations shows the promise of generating explanations which accurately capture the relationship between two documents. Based on the results obtained here, we expect pretrained scientific language models to persist as a foundation. Future work should focus on two complementary goals: ensuring the factual accuracy of the generated text and improved modeling of the cited document. Factual accuracy is difficult to enforce in language model-based text generation systems, especially where inference includes sampling procedures. The use of information extraction for contexts showed promise in Section 6; other methods of incorporating information like grounding to knowledge bases could help prune false or irrelevant statements.

Combining knowledge graphs with language models and generation is an active research area that has shown promise in other domains (Bosselut et al., 2019; Koncel-Kedziorski et al., 2019; Peters et al., 2019). Applying this line of work to scientific text by modeling input documents as knowledge graphs of their content may help algorithms better understand the cited document, provide distant supervision for concurrent work, and result in better outputs.

8 Conclusion

We have described a task of explaining the relationship between two scientific texts and its connections to facilitating researcher productivity. We employ a large, publicly available dataset of scientific documents to train a domain-adapted left-to-right language model for use in text generation applications and beyond. We explore a collection of techniques for representing document content including using abstracts, introductions, sampled sentences, and lists of informative terms and entities. We conduct thorough human and automatic evaluations to determine the relative strengths of each representation for expressing document relationships in natural language text.

Acknowledgements This research was supported in part by the Office of Naval Research under the MURI grant N00014-18-1-2670. We thank the members of Noah Smith's (ARK), Hanna Hajishirzi's (H2Lab), and Luke Zettlemoyer's research groups for their participation in our study. We thank members of Noah's ARK for their helpful comments and the anonymous reviewers for their feedback.

Ethical considerations The authors received an IRB approval for this human annotation project. The project was classified as "no-risk." All participants in the study were volunteers and gave explicit, informed consent for the study.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: Pretrained language model for scientific text. In *EMNLP*.

Chandra Bhagavatula, Sergey Feldman, Russell Power, and Waleed Ammar. 2018. Content-based citation recommendation. In *NAACL-HLT*.

Lutz Bornmann and Rüdiger Mutz. 2015. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *JASIST*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Çelikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *ACL*.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S. Weld. 2020. Tldr: Extreme summarization of scientific documents. In *EMNLP*.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. In *NAACL-HLT*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *NAACL-HLT*.

Arman Cohan and Nazli Goharian. 2015. Scientific article summarization using citation-context and article's discourse structure. In *EMNLP*.

Arman Cohan and Nazli Goharian. 2017. Contextualizing citations for scientific summarization using word embeddings and domain knowledge. In *SIGIR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *ACL*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *ACL*.

Ari Holtzman, Jan Buys, M. Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *ICLR*.

Kokil Jaidka, Muthu Kumar Chandrasekaran, Devanshu Jain, and Min-Yen Kan. 2017. The cl-scisumm shared task 2017: Results and key insights. In *BIRNDL@SIGIR*.

David Jurgens, Srijan Kumar, Raine Hoover, Dan McFarland, and Dan Jurafsky. 2018. Measuring the evolution of a scientific field through citation frames. *TACL*.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text generation from knowledge graphs with graph transformers. In *NAACL*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. 2020. S2ORC: The Semantic Scholar Open Research Corpus. In *Proceedings of ACL*.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *ACL*.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *ACL*.

Sean M. McNee, Istvan Albert, Dan Cosley, Prateep Gopalkrishnan, Shyong K. Lam, Al Mamunur Rashid, Joseph A. Konstan, and John Riedl. 2002. On the recommending of citations for research papers. In *CSCW*.

Preslav I Nakov, Ariel S Schwartz, and Marti Hearst. 2004. Citances: Citation sentences for semantic analysis of bioscience text. In *SIGIR*.

Ramesh Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. 2008. Joint latent topic models for text and citations. In *KDD*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL*.

Chris D. Paice. 1980. The automatic generation of literature abstracts: An approach based on the identification of self-indicating phrases. In *SIGIR*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *EMNLP*.

Jonathan Pilault, Raymond Li, Sandeep Subramanian, and Christopher Pal. 2020. On extractive and abstractive neural document summarization with transformer language models. In *EMNLP*.

Vahed Qazvinian and Dragomir R. Radev. 2008. Scientific paper summarization using citation summary networks. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 689–696, Manchester, UK. Coling 2008 Organizing Committee.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Leonardo FR Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2020. Investigating pretrained language models for graph-to-text generation. *arXiv preprint arXiv:2007.08426*.

Simone Teufel, Advaith Siddharthan, and Dan Tidhar. 2006. Automatic classification of citation function. In *EMNLP*.

Marco Valenzuela, Vu A. Ha, and Oren Etzioni. 2015. Identifying meaningful citations. In *AAAI Workshop: Scholarly Big Data*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *EMNLP*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. SuperGLUE: A multi-task benchmark and analysis platform for natural language understanding. In *NeurIPS*.

Xinyu Xing, Xiaosheng Fan, and Xiaojun Wan. 2020. Automatic generation of citation texts in scholarly papers: A pilot study. In *ACL*.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander Richard Fabbri, Irene Li, Dan Friedman, and Dragomir R. Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *AAAI*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *NeurIPS*.

A Training Details

We perform task-adaptive (continued) pretraining and then finetuning on the task to construct SciGPT2 and SciGen respectively. SciGPT2 starts from the standard pretrained GPT2-base model and is trained for an additional 75k steps at batch size of 64 (effectively a single epoch over 4.8 million abstracts and body paragraphs) with a language modeling objective. We then finetune SciGPT2 to build SciGen for various contexts. For all variants, we finetune the underlying language model for an additional 10 epochs, or approximately 100k steps with batch size of 64.[12]

The hyper-parameters are in Table 6. We provide code for training and evaluating our model as well.[13] Our code is based on HuggingFace's implementation of GPT2-small (117M parameters). We trained on EC2 P3.8x machines which had 4 NVidia Tesla v100 GPUs each. Both models took 24 hours to finish training.

The only hyperparameter we tune is the learning rate. We compared 1e-4 and 6e-5 for our learning rates and used validation perplexity for model selection.

Hyperparameter	Pretrain	Finetune
Epochs	1	10
Effective batch size	64	64
Learning rate	1e-4	1e-4
Weight decay	0.00	0.05
Warmup proportion	0.05	0.10

Table 6: Hyperparameters for the further pretraining and finetuning.

B Dataset Construction

We use data from S2ORC[14] in both the additional pretraining and finetuning. In the former case, we use S2ORC's text with a mask over all citation references. For finetuning, we specifically train on processed data.

We process our data by extracting principal context, cited context, and target sentence triplets. For each principal document, we extract (1) the citation sentences, (2) the principal document context, and (3) the citation's cited document context. We truncate the citation sentence to 100 tokens and the contexts to 450 tokens. Any remaining space is padded with a special token. The two contexts and the target citation sentence are then concatenated together with special separator tokens.

Figure 4 depicts our dataset construction. To construct the data splits, we randomly select 500 principal documents for both test and validation sets. The citation sentences that occur in these principal documents are used as examples in the test (5310 examples) and validation (5164 examples) sets. Of the remaining examples where the principal documents were not in evaluation sets, we throw out any citation sentences that use an evaluation document as the cited document. The resultant examples are used for training (609509 examples). This construction allows us to ensure that the cited document is unseen at test time.[15]

C Examples

See Table 7.

D Human Evaluation Details

To ensure no annotator sees the output of more than one system on each datapoint, we randomly select 50 datapoints for each system (*principal intro* × *cited abs*, IR, and Gold explanations) from the subset of our test data whose principal documents appear in the ACL anthology. We collect judgments from 37 NLP researchers with varying levels of expertise, the majority of whom are graduate students. Each judge is given 15 datapoints for each of the specificity and correctness qualities. Judges are shown a table of datapoints asked to mark whether each meets ("Yes") or fails to meet ("No") the condition. Judges are permitted to label "?" or skip examples they feel uncertain about or unqualified to judge, which we ignore.

E Validity of Human Judgments

To test the validity of the human judgments in Section 5.2, we conduct an additional human evaluation of gold explanations paired with different kinds of mismatched inputs: (1) the correct principal document and a random cited document, (2) the correct cited document but a random principal document (3) random principal and cited documents selected from ACL anthology. Conditions 1 and 2 allow us to see whether human judges accept sentences which align with only one or the other of the input documents; condition 3 provides a lower

[12] We use a triangular learning rate schedule with 10% warmup and a maximum learning rate of 0.0001.

[13] https://github.com/Kel-Lu/SciGen

[14] https://github.com/allenai/s2orc

[15] We also include code for data processing and splits in our repository; see footnote 13.

Principal
Machine translation is important for eliminating language barriers in everyday life. To train systems which can produce good quality translations large parallel corpora are needed. Mining parallel sentences from various sources in order to train better performing MT systems is essential, especially for low resource languages. . . .

Cited
Similarity search finds application in specialized database systems handling complex data such as images or videos, which are typically represented by high-dimensional features and require specific indexing structures. This paper tackles the problem of better utilizing GPUs for this task. . . .

IE-based representation of cited
it ⟨|ENT|⟩ lopq ⟨|ENT|⟩ opq ⟨|ENT|⟩ bucket selection ⟨|ENT|⟩ gpu heap implementation . . . ⟨|TFIDF|⟩ quantizer ⟨|TFIDF|⟩ memory ⟨|TFIDF|⟩ gemm ⟨|TFIDF|⟩ lane ⟨|TFIDF|⟩ warp ⟨|TFIDF|⟩ k-nn . . .

Sentence-based SciGen
For this purpose, we followed the formulation of the greedy algorithm from (*Cited*) for comparing similarity lists obtained from n-best lists

IE-based SciGen
In line with previous work (*Cited*), we use a hash-based distance measure to calculate the similarity.

Citing sentence
We calculate sentence similarities of each possible pairs which can be done efficiently even for large inputs (*Cited*).

Principal
With the development of wireless technologies and popularization of smart phones, mobile traffic grown 4000-fold over the past 10 years and is expected to continue its growth at a compound annual growth rate of 53 percent from 2015 to 2020 (*Cited*). The resulting problem of energy consumption on the Information and Communications Technology (ICT) has become a serious issue. . . .

Cited
We consider the problem of minimization of sum transmission energy in cellular networks where coupling occurs between cells due to mutual interference. The coupling relation is characterized by the signal-to-interference-and-noise-ratio (SINR) coupling model. Both cell load and transmission power, where cell load measures the average level of resource usage in the cell, interact via the coupling model. . . .

IE-based representation of cited
non-linear power coupling equation -lrb- npce -rrb- ⟨|ENT|⟩ non-linear load coupling equation -lrb- nlce -rrb- nlce ⟨|ENT|⟩ average level of usage . . . ⟨|TFIDF|⟩ base ⟨|TFIDF|⟩ r ⟨|TFIDF|⟩ iap ⟨|TFIDF|⟩ load ⟨|TFIDF|⟩ cellular . . .

Sentence-based SciGen
Based on the hybrid beamforming design, the authors of (*Cited*) studied the joint beamforming and power control of massive MIMO cellular networks, and proposed a multi-stage energy-efficient and fair power allocation algorithm in an RAN architecture.

IE-based SciGen
In (*Cited*), the load-coupled problem was addressed and the authors derived the optimal power allocation policy for the worst-case load constrained system considering the two forms of load arrival and power consumption.

Citing sentence
Proof: The proof is similar to that of Theorem 1 in (*Cited*).

Principal
Our lives are increasingly reliant on multimodal conversations with others. We email for business and personal purposes, attend meetings in person, chat online, and participate in blog or forum discussions. While this growing amount of personal and public conversations represent a valuable source of information, going through such overwhelming amount of data, to satisfy a particular information need, often leads to an information overload problem(*Cited*). . . .

Cited
The increasing complexity of summarization systems makes it difficult to analyze exactly which modules make a difference in performance. We carried out a principled comparison between the two most commonly used schemes for assigning importance to words in the context of query focused multi-document summarization: raw frequency (word probability) and log-likelihood ratio. . . .

IE-based representation of cited
raw frequency ⟨|ENT|⟩ log-likelihood weighting scheme ⟨|ENT|⟩ log-likelihood ratio weighting ⟨|ENT|⟩ focused summarizer . . . ⟨|TFIDF|⟩ 0.12717.these ⟨|TFIDF|⟩ topicfocused ⟨|TFIDF|⟩ summaries ⟨|TFIDF|⟩ generic . . .

Sentence-based SciGen
For generic single-query summaries, these measures often give better performance than the traditional measures (*Cited*). . . .

IE-based SciGen
This score is the same as that used by (*Cited*) to generate query-focused summaries from document classification.

Citing sentence
In this work, we use log-likelihood ratio to extract the signature terms from chat logs, since log-likelihood ratio leads to better results(*Cited*).

Table 7: Examples of system inputs and outputs from test set.

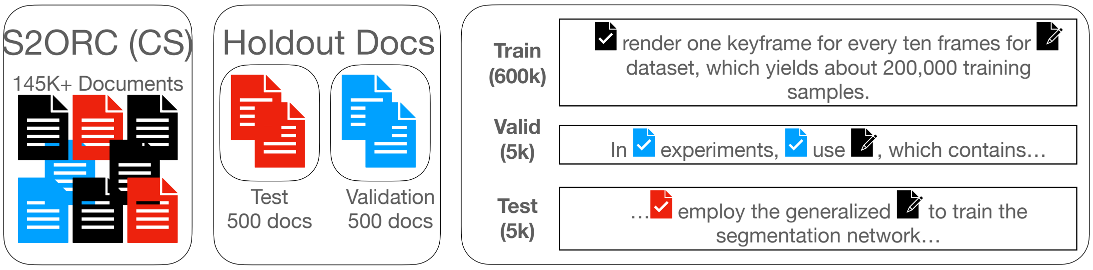

S2ORC (CS)
145K+ Documents
Holdout Docs
Test
500 docs
Validation
500 docs
Train
(600k)
render one keyframe for every ten frames for dataset, which yields about 200,000 training samples.
Valid
(5k)
In experiments, use , which contains…
Test
(5k)
… employ the generalized to train the segmentation network…

Figure 4: Dataset construction from the CS subset of S2ORC. For the far right image, we the documents with checkmarks represent the principal and those with a pencil represent the cited.

	Correct
random cited	45.8
random principal	46.9
both random	17.6

Table 8: Correctness judgements of incorrect citing sentences (percentages).

bound. We collect 107 human evaluations of correctness across these conditions, again allowing annotators to skip datapoints they are unsure of. The results, shown in Table 8, indicate that human judges will sometimes accept a explanationas long as one of the principal or cited documents is correct, but at a lower rate than seen in Table 2 when both documents are correct. We note that both papers in the mismatched cases are drawn from the ACL anthology, meaning there is some amount of topical coherence in their pairing. There is no indication from this experiment that either the principal or cited document is a stronger influence on a judge's correctness decision, although a larger sample size might make a clear determination.

F Further Detail on Automated Metrics Results

We provide a more detailed report of performance on the automated metrics in Table 9 which includes all of our models. The no principal × cited abs model uses no information from the principal to make its retrieval decision, demonstrating the importance of relational information.

G Error Analysis

We investigate the reasons why gold explanations are marked incorrect in our first human evaluation in Section 5. One hypothesis for this gap could be that the grammar of the sentences influenced human judgment. To test this claim, one author annotated each gold example for grammatical correctness and verified these annotations with a commercial writing assistant system.[16] We find that gold explanations with errors are more likely to be classified as incorrect (41.1%) than those without errors (25.4%). These results may partially explain why evaluators rated a portion of the gold sentences as incorrect.

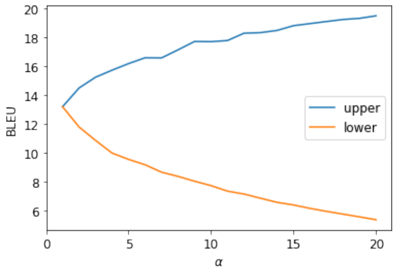


Figure 5: Upper and lower bounds of BLEU for different choices of α.

H Analysis of Token-Wise Overlap

To get a straightforward idea of how much information useful for the relationship sentence is provided by each type of context representation, we calculate the averaged percentage of token-wise overlap of the input and the gold relationship sentence, as shown in Table 10. While a larger overlap does not guarantee a better performance, we found the best performing SciGen systems, with or without ranking, among those using context showing largest overlaps with gold sentences.

I Oracle Study

We conduct an oracle study to see the potential of ranking. Figure 5 shows the upper bound and lower bound of the BLEU score if we independently generate α samples for each pair of (S, C) using SciGen and optimally choose the one with the highest BLEU and the one with the lowest. With $\alpha = 20$, an ideal ranking system could result in a BLEU score as high as 19.50. That provides evidence that

[16] https://www.grammarly.com

	Method	Context	BLEU	Rouge-1	Rouge-2	Rouge-L
Sentence-Based	SciGen	principal abs × cited abs	9.82	10.7	0.6	8.4
		principal abs × cited intro	9.39	10.7	0.6	8.4
		principal abs × cited sample	9.60	10.7	0.7	8.5
		principal intro × cited abs	9.92	11.1	1.0	8.7
		principal intro × cited intro	9.80	11.1	1.1	8.8
		principal intro × cited sampled	9.81	10.9	0.9	8.7
	retrieval	principal abs × cited abs	9.93	14.2	0.7	9.7
		+ MERT (BLEU)	10.23	14.3	0.7	9.8
		no principal × cited abs	9.79	14.1	0.6	9.6
IE-based	SciGen	principal intro × cited tfidf	13.17	15.0	1.3	12.0
		principal abs × cited entities	13.10	14.3	0.8	11.4
		principal intro × cited entities	13.41	14.7	1.4	11.8
	+Ranking	principal intro × cited tfidf	13.50	15.5	1.6	12.3
		principal abs × cited entities	13.28	14.7	1.0	11.6
		principal intro × cited entities	13.16	15.0	1.3	11.8

Table 9: Automatic evaluation of generated texts for all of our systems.

	None	Cited abs	Cited intro	Cited tfidf	Cited entities
None	N/A	18.74	22.95	22.03	22.16
Principal abs x	22.28	32.27	35.69	35.35	35.43
Principal intro x	32.61	41.15	42.81	43.24	43.32

Table 10: Token-wise overlap with gold relationship sentence.

generate-and-rank systems have the potential to surpass generate-only systems regarding BLEU. Our proposed ranking mechanism manages to achieve higher BLEU in some cases, though it performs far below an ideal ranker. That suggests potential future work on further improvements of ranking.

J Auto-Completion

We notice the diversity of expressing the relationship even between the same pair of principal and cited documents. We test whether our SciGen could capture such diversity if provided with different triggers. Our experiment shows that, if we provide the first three words of the relationship sentence to SciGen and ask it to generate the rest of the sentence, the BLEU score of the generated part could be boosted to 21.38. That suggests a use case of SciGen, where a more personalized relationship sentence could be generated given the beginning of the sentence.

K Explaining without Citations

One direction of future work is the ability to provide natural language relationship explanations to pairs of papers without a direct citation link. Table 11 gives examples of SciGEN output for concurrent papers at NAACL 2018. None of these papers cited each other, and thus there was no supervision in generating the explanations.

Principal:	*Learning Joint Semantic Parsers from Disjoint Data*
Cited:	*A Transition-based Algorithm for Unrestricted AMR Parsing*
SciGen:	For *Principal's* task, *Principal* will annotate each graph G with semantic roles from a set of annotations M, that are ”required” in accordance with the AMR graph grammar *Cited*.
Principal:	*Looking Beyond the Surface: A Challenge Set for Reading Comprehension over Multiple Sentences*
Cited:	*CliCR: A Dataset of Clinical Case Reports for Machine Reading Comprehension*
SciGen:	For all of these datasets, *Principal* focuses on performance using a common reading comprehension metric, like F1-score *Cited*.
Principal:	*Attentive Interaction Model: Modeling Changes in View in Argumentation*
Cited:	*Exploring the Role of Prior Beliefs for Argument Persuasion*
SciGen:	*Cited* The primary strategy of this dataset is to focus on the important messages, which are important to people with different viewpoints.

Table 11: Example relationship explanations for pairs of papers that appeared in the same track at NAACL 2018. These papers did not cite each other. These examples have some post-processing done that replaces first person pronouns with "*principal*".