PatentEdits: A Patent Dataset Built for Predicting Revisions

Anonymous ACL submission

Abstract

Patents are critical protections of a company's intellectual property and competitive advantage, as they grant inventors the exclusive rights to make, use and sell the disclosed invention for 20 years. In order to be granted, a patent must be deemed novel and non-obvious by the US Patent Office (USPTO). To meet this criteria, most patent agents and inventors will revise the language and scope of the claimed invention after official feedback is received.

To better understand what revisions lead to successful patents, we present the PatentEdits dataset, which contains 483,706 granted patent examples and is the first to align them before and after revision. We define and extract the following sentence or claim level edit actions in our dataset: a given draft claim is either *kept*, *merged*, *edited*, or *deleted*. For each patent we also include the USPTO examiner cited references, which can be used in edit action prediction.

We also demonstrate the promise of the following model pipeline for predicting the entire granted patent: 1) the prediction of edit actions on the draft claims followed by 2) the prediction of the revised claims with the edit actions.

1 Introduction

A patent application will likely be revised if it overlaps with a pre-existing patent or publication. When a USPTO examiner finds a draft claim that overlaps with a pre-existing patent, they cite the related reference and notify the inventor that the application will be rejected unless it is revised to be novel and non-obvious.

Most patent applications are revised. In a 2015 Yale Law study (Carley et al., 2015) it was found that 86% of patent applications receive a non-final rejection from the US Patent Office after first review. To overcome this rejection, patent writers will often add more detail and specificity where there is overlap with a prior invention. Adding too much detail, however, is a trade-off: the most valuable patent is one that is the most general and least descriptive, as it grants the inventor rights to any future invention that can be described in those terms. This leads to an incentive to change only what is necessary to establish semantic difference with related work.

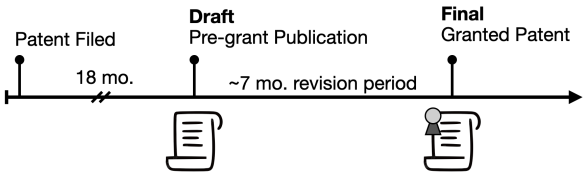

Figure 1: Simplified patent application timeline.

Existing work such as the Harvard USPTO Patent Dataset (Suzgun et al., 2023) focuses on the draft claims and predicting the patentability based on the first submission. However, given that most patent applicants receive a non-final rejection, we pose a different research question: how do patent writers, when faced with overlapping inventive concepts, successfully overcome the objections from the US patent office? More specifically, what edits to a set of claims are needed to overcome the prior work, and can these be learned by language models?

Our contributions are the following:

1. We introduce PatentEdits, the first bulk dataset which aligns the draft to final claims and examiner cited references.

2. We develop edit prediction models that incorporate the cited references for the novel task of claim edit prediction.

3. As a proof of concept, we show that knowing the edit labels can significantly improve the ability to predict the revised text from the starting draft claims.

2 The PatentEdits Dataset

PatentEdits consists of 483,076 patents and 1.3 million cited references from 2001 to 2014. Specifically, the dataset contains the patent claims text before and after revision as well as the claims text of cited references. This critical section of the patent describes the legal coverage of the invention claimed and is the primary focus during official review.

2.1 Data Collection

As there exists no single bulk source containing both the draft and final claims as well as the cited references, they were extracted and aligned from 4 separate USPTO datasets. In detail, utility patent claims text was extracted from two USPTO's Patent Claims Research Datasets (Marco et al., 2016), after which a third USPTO dataset, the Patent Examination Research Dataset (Graham et al., 2015) was used to align the initial claims text, called the pre-grant publication, to the final granted claims text. To obtain the list of examiner cited reference texts for each patent, the USPTO Office Action Citations Bulk Dataset was used.

2.2 Edit Label Extraction

Following Spangher et al. (2022), edit actions were determined by matching draft sentences to the granted sentences based on pair-wise sentence similarity (BLEU-4). A draft sentence is linked to the grant sentence it has the highest score with. As shown in Fig. 4, matched sentences are interpreted as edit actions by the following set of rules:

- a draft claim sentence is labeled as *kept* if a granted claim sentence exists that is identical.
- a draft claim sentence is labeled as *edited* if it is only draft claim linked to a given grant claim sentence. Often this includes adding inventive details.
- a draft claim sentence is labeled as *merged* if it is one of many draft claims linked to a given grant claim sentence and the inventive details combined.
- a draft claim sentence is labeled as *deleted* if its highest similarity score is below a threshold value.

We note that PatentEdits also tracks the added, or new granted claims, however in this work we focus on what happens to the initial draft claims. Fig. 4 gives an example of the sentence matching algorithm algorithm and the extracted edit labels.

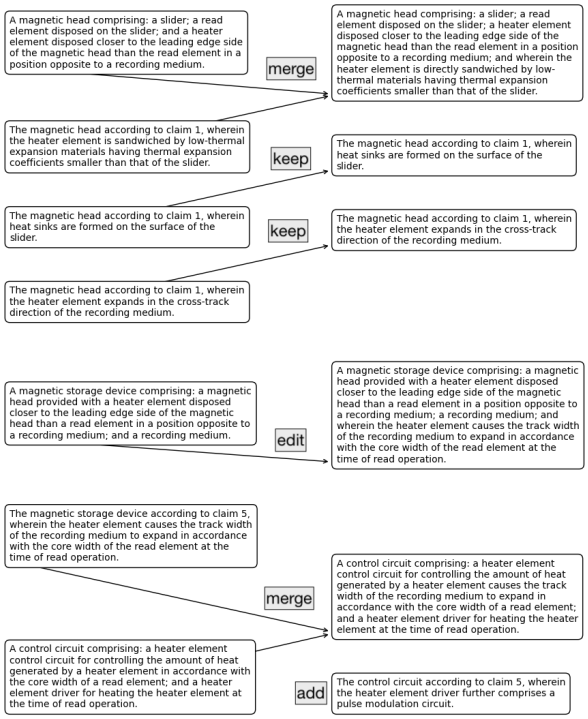

Figure 2: Shown are the extracted edit labels for US Patent 7561362. On the left are draft claims and on the right are granted claims, with edges denoting a sentence match. Additions are tracked but not yet considered.

2.3 Examiner Cited References

PatentEdits also includes a set of cited reference documents (usually prior patents) provided by the US patent examiner during the official review of the draft claims. Although there are cited references from the patent writers themselves, we extract the subset cited by the US patent examiner, as these are the specific prior patents that must be worked-around with claim changes.

During patent prosecution, the examiner and patent writer may directly discuss the specific claims which must be changed, as well as the specific overlap in the prior patent cited; however, these exchanges are not readily available. As we detail later in Section 3, we can model this conversation by retrieving the most semantically similar sentences from the cited documents.

2.4 Dataset Analysis

As shown in 3, a first key insight is that roughly half of the draft claims are kept as-is, resulting in a large degree of overlap between the draft and

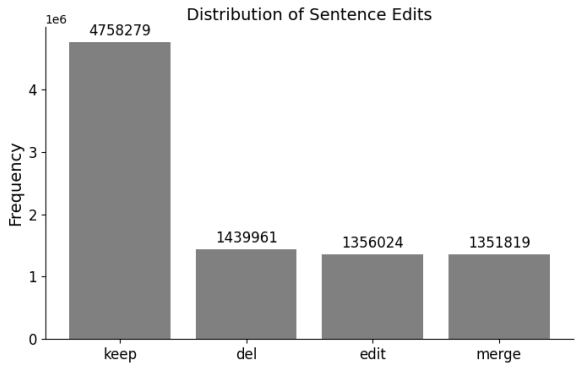

Figure 3: We observe that edit actions in PatentEdits are class-imbalanced. Most claim sentences are *kept*.

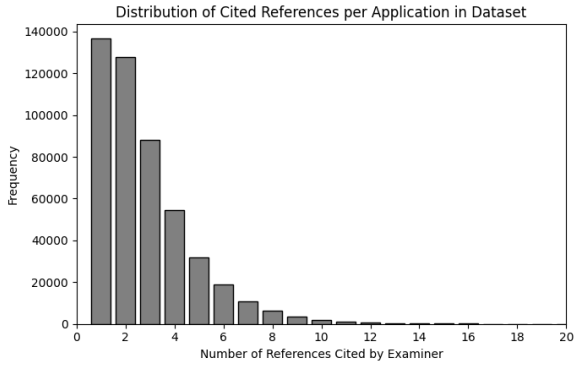

Figure 4: Most patent applications will have 1 or 2 cited references, but others have many. We leverage semantic search to find the most relevant reference on downstream tasks.

final claims. This has implications for both edit-prediction and revision prediction as further detailed in the next sections. When most sentences are kept the same, locating where the revision occurs becomes an crucial preliminary step when predicting the final granted claims.

3 Predicting Edits with References

In this section we set up preliminary studies to explore how edit labels can be predicted with text classifiers. For these experiments and for the revision prediction in the next section, we use a 10k random subset of the full dataset, with an 80-10-10 train-val-test split. We also filter out for patents that have completely been rewritten and patents that were not revised at all. The experiments in this section are intended to illustrate how the included cited references and edit labels can be used, and suggest that more complex strategies might be needed to effectively leverage the cited references.

3.1 Task 1: Sentence-only Edit Prediction

Given only the context of the draft sentence, we want to understand whether we can predict the edit action. This task investigates whether it is possible, given the claim text alone, to predict the edit action, such as by learning to identify vague language that could not possibly meet the novelty criteria.

3.2 Task 2: Sentence+Citation Edit Prediction

Given both the context of the draft sentence and the most semantically similar cited reference sentences, we try to predict the edit action. This task suggests how powerful the examiner cited references are in influencing the edit outcome. As a pre-processing step we align the top-k most relevant sentences in the cited documents to each draft sentence using semantic search with sentence embeddings. We then simply concatenate the top 2 reference sentences for each draft sentences to that sentence for input.

3.3 Reference Retrieval

For **Task 2** we leverage neural retrieval models, similar to those outlined in Sentence-BERT (Reimers and Gurevych, 2019) to retrieve the top-k most relevant sentences in the cited reference documents for each draft sentence in the dataset. Specifically we use *gte-large-en-v1.5* (Li et al., 2023) a BERT-like encoder pre-trained on QA tasks and semantic search. In general, we found that semantic similarity searches worked better than automatic metrics due to the presence of many paraphrases of the same inventive concepts.

3.4 Edit Prediction Experiments

For these sentence-level prediction tasks, we utilized the RoBERTa-base architecture (Liu et al., 2019), a pre-trained BERT-based language model. For both tasks, we utilize under-sampling of the majority class on the training dataset to ensure that predictions for all classes are learned.

We separately fine-tune two RoBERTa-base models for edit classification, one trained on draft sentences with 2 reference sentences and one without the references. For both models we use the same batch size of 32, 6 training epochs, and a learning rate of 2e-5, with 500 steps of warm-up. Note these models only have the context from a single draft sentence and sentences are shuffled across patents during training.

	AUC	Kept	Edit	Merge	Del
Sent	64.0	55.4	29.1	20.7	24.5
Sent+Cit	63.1	55.1	33.5	20.6	23.3

Table 1: Micro AUC and F1 (OvR) for each edit action are reported. We use RoBERTa-base for classification.

	BLEU-4	R-1	R-2	R-L
GPT4 (merge)	45.3	73.0	59.6	69.2
GPT4 (edit)	44.6	74.4	60.2	72.3
BART (all Δ)	**53.5**	**77.5**	**65.9**	**75.6**

Table 2: Sentence level results on changed sentences only. Fine-tuned BART outperforms GPT4 baselines.

	BLEU-4	Rouge-L	BERT
LongT5	55.4	81.7	72.0
GPT4 w. Edits	60.0	83.5	77.8
BART w. Edits	63.6	85.2	**79.0**
Draft Doc.	64.5	86.1	**79.0**

Table 3: Document level test results. BART and GPT4 edit 33% of the total sentences in test. Despite the editing, we see that fine-tuned BART matches the semantic similarity of the draft document to the final claims

3.5 Edit Label Prediction Results

As shown in Table 1, given a single draft sentence and two reference sentences, we do not observe significant difference in edit classification performance between the model trained with two reference sentences and the model trained without. We believe this could indicate either that 1) more draft sentence context is needed or 2) that more reference context is needed. We hypothesize incorporating more context of the surrounding sentences may also improve predictions for the minority classes i.e. edit vs. merge vs. delete.

4 Predicting Revised Text with Edits

In this section, we evaluate how useful the edit actions themselves are, by taking them as perfectly found, then using them to predict the text of the final granted patent. By using edit actions as a guide, a model can learn to selectively edit a few sentences while keeping the majority of them, similar to how human writers are minimally revising patents after examiner feedback.

4.1 Task 3: Revision without Edit Context

Given the entire set of draft claims, we predict or generate the entire set of revised claims without the use of the extracted edit labels. We define this task to better understand whether the logic for editing vs. keeping can be learned implicitly by an attention-based transformer model.

4.2 Task 4: Revision with Edit Context

Lastly, we define the next task as follows: given a single draft claim of the patent (multiple in the case of merges) and the edit label, we predict and generate the revised granted claim. By revising a sentence at a time and excluding the context of the surrounding draft sentences, we explore how predictive the edit label context is alone.

4.3 Revision Prediction Experiments

For **Task 3** we utilize long-context models such as LongT5 (Guo et al., 2022) with efficient attention mechanisms. These efficient transformers enable us to process the entirety of the 800-1200 word patent draft without truncation and also generate longer outputs. We fine-tune LongT5 for 3 epochs, with Top-p sampling of 0.9 and temperature of 1.1 on the training dataset, and report the results in Table 3.

For **Task 4** we consider an approach without extensive fine-tuning and utilize in-context learning with the prompts and examples defined by the edit actions. To choose representative examples for in context learning, we leverage sentence embeddings as outlined in 3.3 to retrieve the top-k (k=2) most similar draft claims in the training set that have the same edit action. We then construct a prompt that includes the most relevant examples with the same edit action, as well as the given draft claim and edit action.

For **Task 4** we also fine-tuned BART (Lewis et al., 2020) to specifically rewrite the edit and merge labelled sentences while keeping or deleting the others according to the edit labels. BART was fine-tuned on the train dataset for 5 epochs, with a fixed learning rate of 5e-5 with an Adam optimizer and decoded with Top-p of 0.9 and a temperature of 1.1.

We score this approach on a sentence and document level: we first compare predicted edit sentences to the actual edit sentences as shown in Table 2 then compare at a document level by aggregating all the machine edited and unchanged sentences back into the full patent. At a document level, we also include the similarity scores of the initial draft patent claims (Draft Doc. in 3)

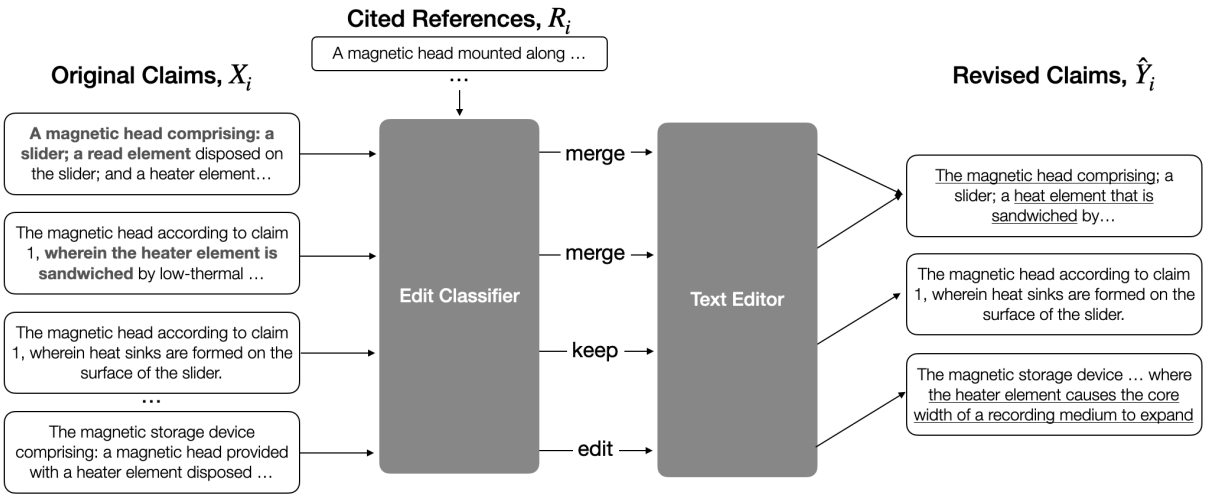

Figure 5: Our proposed pipeline with PatentEdits: we develop edit classifiers with Tasks 1 and 2 and explore text editing approaches in Tasks 3 and 4.

4.4 Revision Prediction Results

On a sentence revision level, fine-tuned BART outperforms our baseline model GPT4 with in-context learning in regards to metrics of n-gram similarity. Our results demonstrate that even given only the context of a single sentence and the edit action, we can reasonably predict the revised and granted claims. However, we do note the need for human evaluation to comprehensively assess revision quality between approaches. In Fig. 6, we show an example of a generated revision vs. an actual one.

On a document revision level, we show that the models that utilize the edit labels outperform the models without, confirming that learning the edit labels prior to revision prediction can significantly improve generation quality. As most sentences are kept, it is likely that models without sentence-edit predictions (labels in our POC experiment) will rewrite sentences that should be kept.

33% of the total sentences in the test set were labelled as merged or edited and 13% are deleted. Thus, for the BART and GPT4 document level predictions, at least a third of the sentences are changed. Despite the significant changes, we see that BART with the edit labels approaches the semantic similarity of the draft claims with the final claims, as measured with BERTScore (Zhang et al., 2020). We consider this a promising result for this edit based revision approach: we see a high semantic similarity with the final claims on par with the initial claims, all while enforcing changes to a third of the draft sentences.

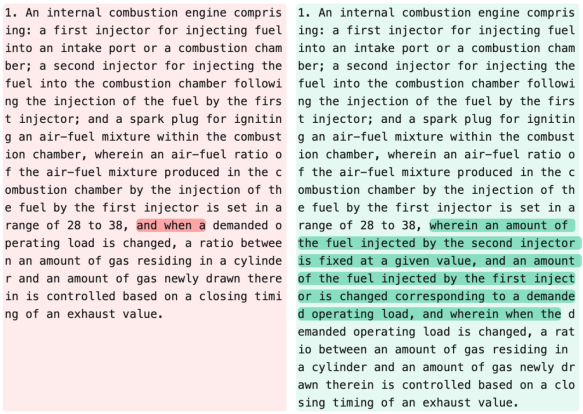

Figure 6: A merged claim generated by BART on the left vs. the actual merged claim on the right. We note that more detail is added in the real merged claim, however there is high n-gram overlap between predicted and actual. Visual with DiffChecker (Diffchecker, 2023)

5 Related Work

Pre-existing patent datasets for machine learning such as the Harvard USPTO Patent Dataset (Suzgun et al., 2023) focus on classifying initial patentability, or predicting patent field class. In contrast, we build PatentEdits to understand how the patent writer overcomes the prior cited references by revising their patent claims.

Lee and Hsiang (2020) described fine-tuning GPT-2 for claim generation. In our approach, we bring new focus towards using the sentence-level edit as prompt context and retrieve relevant edit examples using PatentEdits as a database.

The definition and extraction of sentence-level edit labels extends upon the work of Spangher et al.

(2022) in the News domain: we adapt these methodologies for the patent domain by focusing on using examiner cited references to predict edits and focusing on predicting the revised patent claims text. The concept of using edit tags to guide generation is similar to the approach outlined by Malmi et al. (2019) with their LASERTAGGER model, however we define the use of sentence-level edit labels to guide generation at the sentence level, rather than word level.

6 Conclusions

We introduce the first bulk dataset that aligns the claims text data of patents before and after revision. Given the data insight that most draft sentences are kept, we demonstrate that PatentEdits can be leveraged to build a model pipeline that first predicts edit actions and selectively revises sentences. In this work we also provide experiments which explore the most effective approaches for predicting edits with the cited references and draft sentences. Finally, we demonstrate the importance of edit labels by showing how using the labels to selectively revise can significantly improve the prediction of revisions.

7 Ethical Considerations

7.1 Limitations and Risks

The edit actions in PatentEdits are determined based on rules and automatic metrics, in other words, heuristics rather than human evaluation. While the authors were able to manually verify truthfulness for a subset of examples, it may require further expert evaluation to establish the ground truth of the edit actions for the entire dataset.

Another limitation of this work is that we do not consider predicting the “added" claims. Although the PatentEdits dataset identifies these added grant claims, we do not define any edit prediction for added claims, as other works such as NewsEdits or LASERTAGGER do, i.e. whether a claim will be added before or after a given draft claim. We note that predicting the text of added grant claims (which do not have details in common with the draft claims) may require context from beyond the claims text section of the patent.

Another key limitation of the sentence-level approaches chosen for revision prediction is the ability to replicate the unique format and structure of the patent itself: specifically the aspect that sentences in a patent will refer and extend off of other sentences, i.e. “wherein the golf glove of claim 1 further comprises of a velcro fastener." Specifically, for **Task 4**, sentence level edits from GPT4 are simply concatenated together for the document level comparisons. However, a true patent would ensure the correct dependencies between sentences: although we did not take this step, this re-formatting may be achievable with a post-processing model or algorithm.

7.2 Privacy and Risks

We do not believe there to be any significant privacy risks associated with this dataset as patents are a matter of public record, and PatentEdits is aggregated from bulk datasets shared by the USPTO for the express purpose of research into the patent prosecution process. Although the USPTO Office Action dataset does contain personal identifiers for patent agents and examiners, only the examiner cited references were collected from that data source.

7.3 Computational Resources and Libraries

The PatentEdits dataset was processed with a TPU from Google Colab with 334GB of memory as well as with Google BigQuery. We share the processing code to obtain the PatentEdits dataset from the original sources, however extracting from scratch will require those resources. The fine-tuning experiments in this work are conducted using a NVIDIA V100 GPU with 40GB of GPU memory. The use of GPT4 in **Task 4** requires OpenAI credits, and a total of $25 was expended for experiments and predictions with prompting.

We use HuggingFace libraries and models in this work, such as RoBERTa for edit prediction and encoders sentence-BERT from the Transformers library for extracting most similar edit examples as well as cited references. For evaluation, we utilize publicly available NLP libraries such as NLTK, scikit-learn, bert-score and rouge.

References

Michael Carley, Deepak Hegde, and Alan Marco. 2015. What is the probability of receiving a u.s. patent? Yale Journal of Law and Technology.

Diffchecker. 2023. Text compare. Accessed: June 15, 2024.

Stuart J.H. Graham, Alan C. Marco, and Richard Miller. 2015. The uspto patent examination research dataset:

A window on the process of patent examination. Technical report.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Jieh-Sheng Lee and Jieh Hsiang. 2020. Patent claim generation by fine-tuning openai gpt-2. *World Patent Information*, 62:101983.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 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.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. 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 5054–5065, Hong Kong, China. Association for Computational Linguistics.

Alan C. Marco, Joshua D. Sarnoff, and Charles deGrazia. 2016. Patent claims and patent scope. USPTO Economic Working Paper 2016-04, USPTO. Available at SSRN: https://ssrn.com/abstract=2844964.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. Accessed: 2024-06-12.

Alexander Spangher, Xiang Ren, Jonathan May, and Nanyun Peng. 2022. NewsEdits: A news article revision dataset and a novel document-level reasoning challenge. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 127–157, Seattle, United States. Association for Computational Linguistics.

Mirac Suzgun, Luke Melas-Kyriazi, Suproteem Sarkar, Scott D Kominers, and Stuart Shieber. 2023. The harvard uspto patent dataset: A large-scale, well-structured, and multi-purpose corpus of patent applications. In *Advances in Neural Information Processing Systems*, volume 36, pages 57908–57946. Curran Associates, Inc.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.