PatentEdits: Framing Patent Novelty as Textual Entailment

Anonymous ACL submission

Abstract

A patent must be deemed novel and non-obvious in order to be granted by the US Patent Office (USPTO). If it is not, a US patent examiner will cite the prior work, or prior art, that invalidates the novelty and issue a non-final rejection. Predicting what claims of the invention should change given the prior art is an essential and crucial step in securing invention rights, yet has not been studied before as a learnable task. In this work we introduce the PatentEdits dataset, which contains 105K examples of successful revisions that overcome objections to novelty. We design algorithms to label edits sentence by sentence, then establish how well these edits can be predicted with large language models (LLMs). We demonstrate that evaluating textual entailment between cited references and draft sentences is especially effective in predicting which inventive claims remained unchanged or are novel in relation to prior art.

1 Introduction

Prior work in document edit prediction, such as in news domains (Spangher et al., 2022), do not consider how edits of a given document are influenced by the surrounding body of work. For inventors and examiners, considering the relation of a draft patent to prior work is a critical part of the patent application process, as shown in Fig. 1.

Predicting patent edits is also an important task: patents are critical protections of intellectual property that grant inventors the exclusive rights to make, use, and sell a disclosed invention for 20 years (USPTO, 2024). Unlike prior patent datasets such as the Harvard USPTO Dataset (Suzgun et al., 2023) that focus on the patentability of the first application, we focus anticipating what claims need to be edited in order to be original with respect to the cited references, or the *prior art*. Understanding how prior art influences patent edits would be of use to a majority of those applying for patents

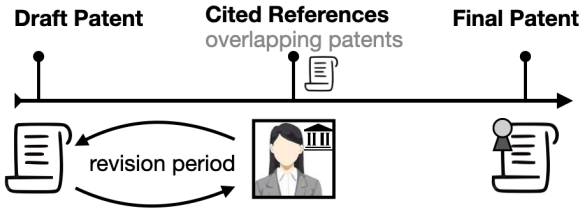

Figure 1: Simplified patent application timeline. PatentEdits aligns text data from the draft, cited references and final patents to understand patent revision.

and the US patent examiners themselves: in a 2015 Yale Law study, Carley et al. 2015 found that 86% of all patent applications are initially rejected by the Patent Office and then revised.

To address the need to study how patents are successfully rewritten after the first application, we introduce PatentEdits, a dataset built to characterize patent revisions and the impact of cited inventive overlap. Our contributions are the following:

1. We provide a corpus of 105k patents where text data from the draft, cited references, and final patent are aligned. We further algorithmically determine which patent sentences are *Kept*, *Edited*, or *Deleted*, and ground these labels with human evaluation.

2. We provide a procedure for adapting semantic retrievers for the patent domain with the PatentEdits dataset, by using an LLM aligned cited sentence as the positive example, the draft sentence as the anchor, the final sentence as the negative example, then fine-tuning retrievers with triplet loss.

3. With our edit prediction experiments, we demonstrate that entailment-based approaches are an effective means of evaluating patent claim novelty. We report that classification improves by including the cited references and focusing on the entailment between cited and draft patent claims.

2 The PatentEdits Dataset

PatentEdits consists of 105,000 utility patents from 2007 to 2014. Unlike prior patent datasets, each example in PatentEdits aligns the draft to the final granted patent text as well as the complete text of the patents or publications cited by the USPTO examiner. These cited references contain descriptions and ideas which invalidate the novelty of the draft patent, and by including them, we are able to use this context for edit prediction. An instance of the PatentEdits dataset is provided in Table 1.

In this work, we focus specifically on the *claim sentences* of the draft, cited, and final patents. This critical section of a patent describes the legal coverage of the invention claimed and is a primary focus during official review by the USPTO (Duening et al., 2020).

2.1 Dataset Sources

There exists no single bulk source that aligns both the draft and final claims as well as the cited references, so we extracted and aligned from 4 separate, publicly available USPTO datasets. Aligning the draft text to the final text is challenging as the draft patents are published separately from the final patents with different unique identifiers: the draft patent text only has a unique publication number, whereas the final granted patent only has a unique patent number.

2.2 Extracting Draft and Final Patent Text

First, to align the final patent text to the correct corresponding draft we download the Patent Examination Research Dataset (Graham et al., 2015), which we found provides a mapping between the publication number and the patent number. We use SQL queries (Google BigQuery) based on publication number to extract the draft patent text, then queries based on patent number to extract the final patent text. We specifically choose to query from two USPTO's Patent Claims Research Datasets available on Google Cloud (Marco et al., 2016), which includes pre-processed, cleaned patent sentences. We conduct a merge of the text data found from draft and from final patents, keeping only the examples with both. Because these cleaned and pre-processed text sources span from 2007-2014, so too does PatentEdits.

ID	US Patent 7490780
D	["1. A radiation image projection apparatus comprising: an image acquisition unit. . . configured to project the radiation image as visible light onto the object. . . ", "2. The radiation image projection apparatus according to claim 1, further comprising: a radiation generating unit. . . "]
C	["1. An X-ray device . . . wherein the indicator means includes an acoustic alarm device . . . ' 2. An X-ray device . . . wherein the indicator means is arranged on one of: the X-ray detector and the X-ray source.. . . "]
F	["1. A radiation image projection apparatus comprising: an image acquisition unit. . . and a mirror configured to transmit radiation and reflect visible light;. . . ", "2. The radiation image projection apparatus according to claim 1, further comprising: a radiation generating unit. . . "]
ε	['edit', 'edit', 'del', 'keep', 'keep', 'keep', 'keep', 'keep', 'keep', 'edit', 'edit']
E	[(0, 0), (1, 1), (3, 2), (4, 3), (5, 4), (6, 5), (7, 6), (8, 7), (9, 8), (10, 9)]

Table 1: An instance in the PatentEdits dataset, which consists of the patent number **ID**, list of draft sentences **D**, a list of final patent sentences **F**, a list of cited reference sentences **C**, a list of edit labels for each draft sentence ε, and **E**, a list of found sentence edges.

2.3 Extracting Examiner Cited References

To obtain the text data from examiner cited references, we download the USPTO Office Action Citations Dataset and filter for the 35 U.S.C. 102 and 35 U.S.C. 103 objections, which correspond to office rejections for novelty and non-obviousness (Hellmann, 2024). This dataset only contains a list of unique identifiers, so as we did in Section 2.2. we query for the text data using those identifiers.

To make attribution of inventive overlap easier, we filtered the dataset for patents that have 1 or 2 cited references (48% of the original examples) and only include examples where the full citation context is found. We plan to release the full PatentEdits dataset and scripts in our project repository.

2.4 Edit Label Extraction

In order to study trends in patent revisions, we apply algorithms to synthetically label which sentences of the patent are *Kept*, *Edited*, or *Deleted*. We construct a bipartite graph and find edges between sentences of the draft and final and determine the *edit actions* possible on draft sentences. As shown in Fig. 2, after the sentence matches are found, edit actions are determined by the following set of rules:

- *Kept*: a draft sentence is labeled as *Kept* if it is matched to a final patent sentence and its similarity score is above a *Kept* threshold. We validate this threshold further with an additional human study.

- *Edited*: a draft sentence is labeled as *Edited* if matched to a final patent sentence and its similarity score is below the *Kept* threshold.

- *Deleted*: a draft sentence is labeled as *Deleted* if there is no final sentence with sufficient similarity above a *Deleted* threshold.

To find the best edges, we explore greedy algorithms to find these edges based on similarity scores between draft and final sentences. In this work, we do not consider sentence splits: we limit the number of outgoing edges to only one, from each draft sentence. In all approaches we have a *Deleted* threshold on similarity score, where if no edge is found with a sufficient similarity, that draft sentence is considered *Deleted*.

We first evaluate using the same sentence matching following (Spangher et al., 2022) using BLEU-4 (Papineni et al., 2002) as the scoring method. We consider this approach to be "draft-side greedy" as it matches every draft sentence to the grant sentence it has the highest BLEU-4 score with. In addition to BLEU-4 as the scoring method, we explore using BLEU-1, METEOR (Banerjee and Lavie, 2005), CHRF (Popovic, 2015), and BERTscore (Zhang et al., 2020).

In our next set of experiments, we explore a "final-side greedy" approach, assigning a final patent sentence to the draft sentence it has the highest similarity score with. Similarly to the draft-side greedy approach, we explore using Rouge-L, Rouge-1, METEOR, CHRF, and BERTscores as the similarity score, and select the best score based on alignment with human annotations.

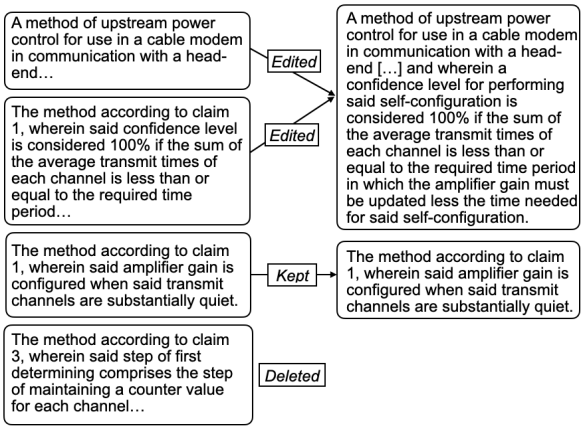

Figure 2: Shown are the extracted edit labels for US Patent 8677435. On the left are draft claims and on the right are final patent claims, with edges denoting a sentence match.

Motivated by the observation that many draft sentences can be *Edited* into a single final sentence, we implement a match-and-cover algorithm for final-side greedy approaches to ensure that all relevant draft sentences are properly attributed to a given final patent sentence. More specifically, we exhaustively search for all draft sentences that could match with the remaining grant sentence, after a longest common subsequence (LCS) is removed following a match. We detail the exact logic in Alg. 1.

Algorithm 1 Match-and-Cover Procedure for Final-Side Greedy Matching

Input: Set of draft sents D, set of final sents F, matching threshold τ, fraction limit ϵ
Output: Matches of draft to final sentences
for each final sentence $f \in F$ **do**
    while text in f left > than frac limit ϵ **do**
        Find best unmatched draft d^* for f
        Identify LCS between d^* and f
        if match score is >= to threshold τ **then**
            Remove LCS from f
            Attribute d^* to f
            Add d^* to "matched" set
        else
            break
        end if
    end while
end for

	Embedding		Lexical		Ngram	
Algorithm	Scoring Method	F1	Scoring Method	F1	Scoring Method	F1
Draft-side Greedy	Roberta-Large	83.0	ChrF	60.7	BLEU-1	80.0
	Deberta-large-mnli	82.7	METEOR	82.0	BLEU-4	82.9
Final-side Greedy	Roberta-large	87.8	ChrF	35.6	Rouge-1	87.1
	Deberta-large-mnli	88.6	METEOR	87.8	Rouge-L	**90.3**

Table 2: Sentence Matching F1 scores against human annotations using embedding, lexical, and ngram-based scoring methods and greedy matching.

2.5 Human Evaluation of Edit Labels

To evaluate the quality of the automatic matching between draft and final sentences of the document, we instruct human annotators to match sentences between draft and final patent documents for 90 patents in the PatentEdits dataset.

For all the sentences of the draft and final patent, annotators are instructed to match a given draft sentence to a granted sentence if they have substantial overlap in meaning, even in instances where the inventive detail has been paraphrased in the revised granted sentence. If there is no substantial conceptual overlap between the two sentences, or if it is unclear how the two sentences are semantically related, we instruct the annotator not to match the sentences. We use our sentence-matching algorithms on the same 90 human annotated patents, then report the F1 scores with the human annotated labels as the ground truth label.

An edge between a draft and final sentence can mean a sentence has been either *Edited* or *Kept*. To determine the difference with an automatic means (after sentence edges are found), we set up an additional human study: we select a random subset of 100 examples of matched sentence pairs and evaluate if they are *Kept* or *Edited*. We then asked annotators to mark the paired draft and final sentence as *Edited* if there was substantial conceptual difference between the two sentences, or *Kept* if the two sentences were exactly the same, with the exception of different claim numbering or typos. We then use this human evaluation study to determine the *Kept* threshold for all our different approaches.

2.6 Automatic Edit Label Performance

We report the resulting performance of these algorithms and scoring methods in Table 2. We find our best sentence scoring algorithm is a final-side greedy approach which uses Rouge-L scoring, with a *Deleted* threshold of 0.45 and a remaining fraction threshold of 0.3. Our best sentence matching algorithm achieves an F1 score of 90.3 against human annotations. This F1 score is comparable to the best matching F1 score of 89.5 seen in NewsEdits (Spangher et al., 2022) with a BERT-based algorithm and between revisions of news articles.

We found that using BLEU-4 with a *Kept* threshold of 0.88 resulted in the best agreement with the human labeled *Edited* vs. *Kept* examples, with an F1 score of 0.91 (*Edited* F1) and 0.97 (*Kept* F1). To obtain the edit labels for the entire dataset, we apply the final-side greedy, Rouge-L matching algorithm and the *Kept* threshold (BLEU-4 of 0.88) to the entire PatentEdits dataset.

3 Training Semantic Retrievers

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; Thus these exchanges are not readily available to researchers, as these interviews can be conducted over the phone and are not fully on public record. However, in PatentEdits, because we have the text data of all the cited patents in the patent application process, it is possible to extract that specific context. To extract the specific context from the cited references, we rank the sentences from the cited documents based on their semantic similarity to a given draft claim and extract only the most relevant ones.

We are motivated to use retrievers to rank, of all cited sentences, which are most similar to a given draft sentence, then extract the top-k candidates. However, without specific fine-tuning retrievers may not find the correct passages or find irrelevant ones. In this work, we fine-tune sentence-BERT (Reimers and Gurevych, 2019) to semantically retrieve relevant inventive detail, by using labels extracted by GPT4o and from PatentEdits. We outline the following general procedure for adapting off-the-shelf retrievers to the patent domain below and in Fig 3:

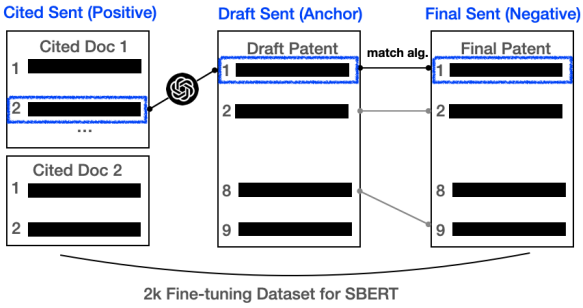

Figure 3: We obtain high quality retrievals with ChatGPT4o, then treat it as the positive, the draft sentence as the anchor, and the final sentence as the negative, for triplet loss. We treat the final patent sentence as a negative example as we assume it has been rewritten to be semantically different from the draft. One training example for the retriever is one triplet of sentences.

1. First, we extract from our training split (see Section 4) a subset of 2000 *Edited* sentences.

2. Next, for each draft sentence in the subset, we prompt an LLM (in our case GPT4o) to copy the sentence the cited claim is most likely to be infringing on.

3. With the 2K dataset, we use the extracted cited, draft, and final claim as the positive, anchor, and negative examples to train a semantic encoder with triplet loss (Schroff et al., 2015).

4. We then use the lightweight, fine-tuned sentenceBERT (SBERT) to retrieve the top cited sentence from the cited documents for the rest of the dataset.

We use triplet loss to specifically leverage an important aspect of the dataset: by sentence matching and edit labeling, we have found the final sentences have been edited to semantically distance themselves from their draft versions. The triplet loss is defined as:

$$\mathcal{L}(A, P, N) = \max(0, d(A, P) - d(A, N) + \alpha)$$

By using triplet loss, we train our encoder to map draft sentences (A) closer to the relevant cited references (P), while distancing from the semantically changed final patent sentences (N). We use this fine-tuned retriever to extract the most relevant cited reference sentence for every draft sentence in our dataset, and set up entailment experiments for edit prediction.

4 Patent Edit Classifiers

In this section we establish baselines for classifying the future edit of a given patent sentence given the cited references and the text of the draft sentence. For these experiments, we define 10k random subset of the full dataset, with an 80-10-10 train-validation-test split. We explore small and large models, vary the context size, as well as the model objective.

We ensure that the 10k subset covers the same time span as the full 105k dataset, and that the experimental splits have the same relative ratio of edit classes. We 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 leveraged, and suggest the importance of the cited reference to the edit prediction task.

4.1 Edit Classification Experiments

Sentence-only Edit Prediction Given only the context of the draft sentence, we seek to predict the most likely the edit for a given draft sentence. Here, we investigate whether it is possible to predict the edit action given only the claim sentence, such as by identifying vague conceptual language that could not possibly be novel.

Sentence+Citation Edit Prediction Given both the context of the draft sentence and the most semantically similar cited reference sentences, we predict the edit action on the draft sentence. This experiment explores the signal that examiner-cited references have in influencing the edit outcome. To incorporate the most relevant cited sentence, we naively concatenate it to the end of the draft sentence, and include token delimiters. Specifically we add "[claim]" before the draft sentence and "[cited]" before the cited reference sentence.

Window Edit Prediction So far, we have only considered the context from a single sentence and/or a single cited reference sentence. In this experiment, we explore how increasing the context window and allowing models to process more context can improve edit label predictions. To prepare this context for an LLM, we simply concatenate the previous, current and following draft sentences into a text chunk and use as label the edit of the center draft sentence. As a starting point, we use a window of three draft sentences to predict one edit.

Context	Model	Weighted	Micro	Macro	Kept	Edited	Deleted
Sent	Roberta-Base	50.3	49.6	46.1	60.0	41.3	37.0
Sent+Cit	Roberta-Base	50.4	49.7	46.4	59.7	41.7	37.4
Window	Roberta-Base	52.3	51.9	47.7	62.8	42.7	37.8
Sent	Roberta-Large	54.4	54.4	49.8	65.4	45.4	38.7
Sent+Cit	Roberta-Large	54.0	53.3	49.7	64.0	45.4	37.6
(Cit,Sent)	Roberta-Large-MNLI	55.5	57.3	49.1	**70.1**	42.7	34.5
(Cit,Sent)	BART-Large-MNLI	**55.6**	**57.4**	49.5	69.8	42.7	35.9

Table 3: Test F1 scores for edit prediction on the experimental 10K dataset. We see that simply concatenating the closest extracted citation to the draft text does not improve results, but treating the citation and draft sentence as an entailment pair (*premise*, *hypothesis*), using models pre-trained on entailment, and mapping predicted entailment classes to edit classes greatly improves overall F1 score, especially for the *Kept* category.

Cited to Draft Entailment In this experimental set-up, we re-frame the edit prediction task as classifying textual entailment. If the draft claim *Entails* from the most relevant cited claim then it infringes and should be *Edited* or *Deleted*. If the draft claim *Contradicts* the cited claim premise, than the draft claim can be *Kept* as it is original or novel.

To validate the mapping between entailment labels and edit labels, we first conducted a Chi-squared test of independence between the predicted entailment labels and the edit labels using an entailment model without fine-tuning (microsoft/deberta-large-mnli) (He et al., 2021). The test yielded a p-value 2.6e-22, which is significantly lower than our significance level of $\alpha = 0.05$. As a result, we reject the null hypothesis that the predicted entailment labels and the edit labels are independent and conclude that there is a statistically significant association between the two sets of labels.

For the entailment experiments we fine-tune Hugging Face (Wolf et al., 2020) transformer models Roberta-Large-MNLI and BART-Large-MNLI, which are pre-trained on the Multi-Genre Natural Language Inference (MultiNLI) dataset (Williams et al., 2018). During training, we consider the most relevant cited sentence as the premise and the draft sentence as the hypothesis. To adapt the labels to edit prediction, we simply map the *Contradiction* label to *Kept*, *Neutral* to *Edited*, and *Entailment* to *Deleted*.

4.2 Additional Experimental Set-up

For these sentence-level prediction tasks, we primarily used the RoBERTa architecture (Liu et al., 2019), an optimized BERT-based transformer. For all of the three context scenarios, we use under-sampling of the majority class on the training dataset to mitigate class imbalance in the dataset. For experiments with Roberta-base models we use a same batch size of 32, 6 training epochs, and a learning rate of 2e-5. For the Roberta-large models, we reduce the batch size to 16 to fit the training on a 40GB GPU, reduce the number of training epochs to 3, but keep the same learning rate of 2e-5. We used a learning rate warm-up of 500 steps.

5 Edit Prediction Results

As shown in Table 3, simply concatenating citations is not enough to improve draft edit prediction. Across the small and large models, we don't observe a significant change in performance by simply including the cited reference text as additional input.

Our most notable result is that entailment between the citation and draft sentence significantly improves overall edit prediction specifically for the *Kept* class, which corresponds to the *Contradicts* entailment label. Using the entailment between the cited and draft sentence gives us the strongest weighted F1 score of all the models, however we do note that for the *Deleted* class, the entailment models perform worse than the others.

We also see the expected trends: larger models outperform smaller ones, and larger context windows improve performance. More specifically, we see that Roberta-Large performs better than Roberta-Base for both with and without citation context, and we also see that the windowed context performs better than any other of the Roberta-Base model contexts.

The F1 scores we see are comparable to what is reported in other research works on document edit prediction, such as NewsEdits (Spangher et al., 2022) and LASERTAGGER (Malmi et al., 2019).

6 Discussion

In the classification experiments, including the cited references improved predictions on which sentences would be edited when we framed edit prediction as an entailment evaluation: we interpret this as confirming our assumption that the cited references provide critical information on how the patent should be rewritten. In these experiments, we use lightweight models and relatively small contexts, only the context of a single draft sentence and at most 2 additional sentences. As our windowed experiment suggests, including more of the draft document and cited references could further improve the classification performance.

Our results suggest that in general, the *Edited* and *Deleted* classes are harder to predict, as they have lower classification F1 scores for all experiments than *Kept*. We do not believe this is simply due to *Kept* being the majority class, as we account for class imbalance by under-sampling the majority class and equalizing the number of each edit class examples in training. Instead, we hypothesize that once the set of offending claim sentences are identified (the ones not *Kept*), the decision about which of the remaining to edit or delete may either be arbitrary or dependent on other factors beyond inventive overlap with the cited references.

7 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 construct PatentEdits to understand how the prior cited references influence the revision of patent claims.

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.

8 Conclusion

In this work we curate a dataset of 105k patents to characterize document level change as a function of prior work, aligning text data from the draft, cited references, and final versions. We enable edit prediction by algorithmically labeling patent sentences (or claims) into edit categories such as *Kept*, *Edited*, or *Deleted* and ground our labels with extensive human evaluation. We also provide a procedure for using PatentEdits to adapt semantic retrievers to the patent domain. By leveraging an LLM-aligned cited sentence as the positive example, the draft sentence as the anchor, and the final sentence as the negative example, we fine-tune the retrievers through triplet loss. Our edit prediction experiments validate the effectiveness of entailment-based approaches in assessing patent claim novelty. We demonstrate that incorporating cited references and focusing on entailment between cited and draft claims significantly improves classification performance.

Ethical Considerations

Limitations

The edit actions in PatentEdits are determined based on rules and automatic metrics and verified with human evaluation. While the annotators were able to manually verify truthfulness for a subset of examples, the quality and correctness of the automatic means may further improve with expert evaluation, i.e. by patent agents or USPTO examiners.

In this work we do not consider predicting the “added" claims, but they do appear and can be tracked with PatentEdits. We consider *Added* claims as substantively differen from classifying draft claim outcome, however we note that other works such as NewsEdits or LASERTAGGER do define tasks related to anticipating whether a sentence will be added before or after a given sentence.

With our preliminary studies on sentence-only context, we do not utilize the unique independent-dependent claim relationships: claim independence refers to the aspect that some 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."

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.

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 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

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

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

Thomas N. Duening, Robert D. Hisrich, and Michael A. Lechter. 2020. Protecting your intellectual property. In *Technology Entrepreneurship: Taking Innovation to the Marketplace*, chapter 6, pages 151–181. Academic Press.

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.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Tim Hellmann. 2024. The most common rejections: 102, 103, and 112(b). Accessed: 2024-10-14.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, PA, USA. Association for Computational Linguistics.

Maja Popovic. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

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

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 815–823. IEEE.

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.

USPTO. 2024. Utility patent application process. Accessed: October 13, 2024.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, volume 1, pages 1112–1122.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*.