**Building a More Effective Patent Prior Art AI Search Tool**

1. **Problem**

Google Patents’ current search functionality is limited in its ability to return relevant and accurate patent prior art search results, especially when users input long and detailed queries. This shortcoming stems from the search engine’s inability to effectively interpret inventors' variations in the use of complex keywords and phrases in their patents. These variations, often the result of draftsman's art, may express similar concepts but differ syntactically, leading to inconsistent or incomplete search results. This inefficiency often leads intellectual property lawyers who search prior art to significant delays and misunderstanding of patents.

1. **Objectives, Proposed Solution and Methods**

The objective of our project is to improve the accuracy and relevance of patent prior art search results by addressing the limitations of Google Patents. We aim to enable intellectual property lawyers to perform searches with both short and detailed queries while ensuring our patent prior art AI search tool tailors the results to their queries. We want to present this information through a webpage that will re-rank Google Patents searches based on revised relevancies.

Our proposed solution emphasizes a tailored ranking system based on semantic similarity. This will allow intellectual property lawyers to quickly identify prior art that aligns more closely to their specific queries than the output of Google Patents, greatly improving the speed and efficiency of prior art searches. We use Python, embeddings, and cosine similarity to re-rank the output of Google Patents to show the most pertinent prior art. This will bridge the gap between language variations due to draftsman’s art.

In our program, we ask the user to give two inputs: a short keyword search and a detailed holistic search. We use Python to control the back-end, where we use Selenium to scrape Google Patents for the initial search results and extract all critical information, except the text of the patent. To get the text, we scrape the HTML from the patents, such as within the abstract, summary of invention, detailed description of invention, claims, and specification.

Then, a custom function parses the HTML structure of the patents. We perform text preprocessing with spaCy to remove stopwords, punctuation, and irrelevant parts of a sentence. After preprocessing, we embed each sentence from the text using sciBERT, which is a model especially useful for technical and scientific language. Most importantly, we must evaluate the relevance of each patent. To do this, we calculate the cosine similarity between the holistic search embedding and sentence embeddings from the scraped patent text.

Our back-end places the sentence with the highest cosine similarity score for each patent and inserts it into the Pandas dataframe. The dataframe is then re-sorted based on the sentences with the highest cosine similarity. Finally, we display the re-ranked, most relevant patents at the top of the HTML search page based on the cosine similarity score. The closest scores are highlighted in green, followed by yellow and red, as the results populate the rest of the page. With each patent result, we also list the Google Patents link and provide the most relevant sentence snippets.

1. **Successes**

Our model significantly improves the Google Patents search process by addressing its primary limitations and saving patent lawyers valuable time in conducting prior art searches. Google Patents typically relies on keyword-based ranking, and any longer or more detailed queries rarely lead to useful results, because previous patents rarely use the exact same wording as the searcher. These generic keyword-based searches produce a massive amount of mostly irrelevant results (see Appendix A), forcing patent lawyers to spend hours manually reviewing patents to find relevant prior art.

Our model solves this problem by using a semantic ranking system that evaluates the intended meaning behind both the query and the content of patents (see Appendix B). Using SciBERT embeddings, the tool ranks results based on their true relevance to the query, not just keyword matches. This approach brings the most useful patents to the top of the list, ensuring that lawyers see results closely aligned with their search. Further, each result includes the most pertinent sentence from the patent, giving searchers a quick preview without reading the entire document to confirm that the search properly produces results as intended. The tool then color-codes the result’s cosine similarity to the nuanced query—green for highly relevant, yellow for moderately relevant, and red for less relevant—so users can receive a quick and simple signal about the applicability of a search result. This color-coding system also provides researchers input into the quality of their keyword searches. If the keyword search does not produce results with any sentences similar to the holistic query, then either there are no similar results, or the keyword search needs to be refined in order to produce better results.

These features allow patent lawyers to conduct faster, more accurate searches. By reducing the effort needed to sift through dozens of irrelevant results, our model not only increases efficiency in identifying the prior art that matters most, but also saves clients expenses that would otherwise be incurred for generic searches.

1. **Limitations**

While our model offers substantial improvements to the standard patent search functionality, it also has some limitations that may affect its operation and efficacy. One issue we encountered lies in the process of HTML scraping. To gather text data from the prior art produced in Google Patents searches, our model relies on Selenium, which, while effective for our purposes, is slow and resource-intensive because it automates an entire web browser to extract the content it scrapes. This inefficiency becomes particularly apparent when handling large-scale search results, as Selenium processes each page individually and requires substantial system resources. Alternative solutions, such as SerpApi’s Google Patents API or the PatentsView API, are available and could streamline the process by providing structured data directly without the need for web browser automation, offering faster and more reliable results. However, these options are cost-prohibitive, making them unreasonable under our current budgetary constraints.

Additionally, the model faces some challenges with processing time and computational demand. Embedding lengthy patent texts and calculating cosine similarity for each snippet requires considerable processing power, which in turn, results in slower overall performance. Depending on the computer used to run the model, each search can take between ten to twenty minutes to produce the intended results. This limitation affects the scalability of our model and the user experience, especially for searches involving numerous patents, but it can easily be mitigated by using computers with more optimal processing capabilities. One possible solution is to limit the model to scraping a limited number of sentences and then analyze those, rather than scraping the entirety of a patent, which can be upwards of eighty pages long.

Another limitation comes from the use of cosine similarity to rank search results. Cosine similarity scores can be skewed by differences in sentence length, as longer sentences tend to contain more keywords and context that can artificially inflate their similarity score, even if the content is only partially relevant. Conversely, shorter sentences may lack sufficient overlap with the detailed query, causing their relevance to be underestimated. This disparity can result in overly long or short sentences inaccurately appearing to be more relevant. Moreover, relying on a single sentence with the highest similarity score may not always represent the patent’s overall relevance to the query. Although this limitation could create potential concerns with the result, we aimed to address the issue by including a snippet of the most relevant sentence to the query so that the searcher could recognize erroneous results and disregard them if it is clear that the result is not relevant to the purpose of their search.

Finally, the difficulty in measuring error in our tool’s results poses a challenge for the goals of this project. Unlike various machine learning models that we experimented with throughout this course, which have used metrics like accuracy and loss to provide clear feedback on model performance, the subjective nature of determining “what is relevant to this patent search” makes it difficult to quantitatively evaluate the model’s effectiveness. This is another reason why we felt that including a snippet of the most relevant sentence within each patent would be helpful in allowing searchers to gauge the accuracy of the result without reading the entire document. Even without the re-ranking or color-coding capabilities, this snippet display is a substantial step up from the normal Google Patents search result. Although it still would not create a numerical score for the model’s performance, a potential improvement to our methodology could be calculating the average cosine similarity of the top 5% of sentences within a patent rather than relying solely on the highest score. This approach could offer a more balanced assessment of relevance while reducing the risk of outliers influencing rankings. On the other hand, such calculations may require greater processing power, potentially leading to even longer search times.

Despite these challenges, the current model consistently delivers search results that are significantly more tailored and relevant to a searcher’s goals than those provided by standard Google Patents searches. While some limitations, such as inefficient scraping methods and slow search times, have clear solutions that could be implemented with additional resources, other improvements to the model will require further research and experimentation. Nonetheless, the model establishes a starting point for improving the efficiency and accuracy of the patent search process while recognizing various avenues for future development.

1. **Future Development Possibilities**

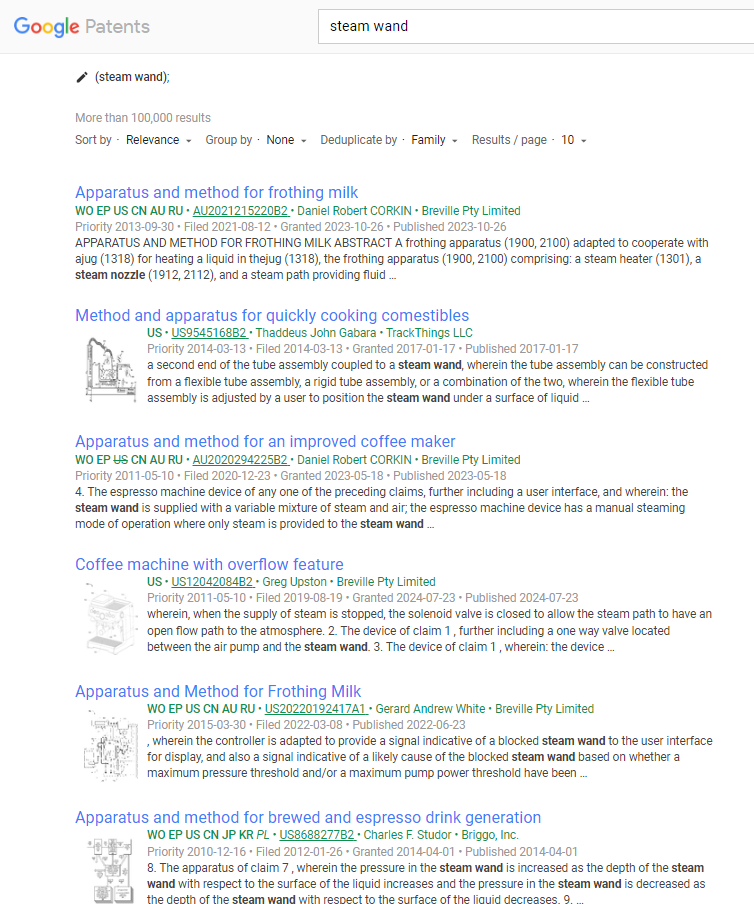
Despite the great success of the project, there is room for further development. Possible improvements fall into two main categories: refinement of the current process and implementation of new technology.

To further refine the current processes, patent filtering is the obvious adjustment. Relevant refining information is already gathered in the data. For better use as a tool for researching patentability, implementation of priority date filters would be paramount. Limiting research to patents with priority dates before or between certain dates would allow practitioners to return only patents with potentially determinative relevance. For alternative uses, such as freedom to operate analyses, refining the patent search results by country of origin could be highly relevant. The HTML data parsed under the tag ‘metadata’ indicates which countries may have patent applications or patents granted under the World Intellectual Property Organization (WIPO) Patent Cooperation Treaty. By filtering by country of origin, the application could be more optimized for researchers interested in evaluating the risk of patent infringement litigation.

Further development would include the implementation of more algorithms. We built a version of the application that includes K-means clustering as part of the results. But with the limited nature of the application, it was not a helpful addition to the algorithm. However, if the program was built out to include more search results, then K-means clustering would be a useful addition when joined with other technologies. A cluster is not particularly helpful if it is not labeled effectively. Thus, one potential way to effectively implement a clustering algorithm would be to have the abstract of each patent in a cluster extracted and fed into an LLM together, perhaps via Grok, and let a traditional LLM create a label for each cluster.

1. **Conclusion**

Google Patents remains the dominant program in the patent search field, but there is room for improvement. Through the use of traditional text pre-processing techniques and specialized sentence-level neural network embeddings, simple parsing algorithms and cosine similarity calculations can meaningfully improve Google Patents search results to better serve the needs of the everyday patent researcher.

**Appendix A**

**Appendix B**