PATENT PRIOR ART SEARCHING AS CONTRASTIVE CLASSIFIER LEARNING

Julius Heitkoetter, Nicholas Abate, Sohini Baidya
Thinkstruct
Cambridge, MA 02139, USA

ABSTRACT

Recent work in natural language processing has significantly broadened the scope of information retrieval beyond keyword-based search. Patent law offers a compelling domain for these methods due to its large scale publicly available corpus of highly structured data. Yet, the current leading approaches to patent searching rely on legacy keyword-based systems, only partially augmented by large language model wrappers or generic semantic similarity comparisons. We cast patent search as a contrastive classification problem, learning a scoring function that directly captures the legal concepts of novelty and non-obviousness in the relationship between a patent and the prior art. Using a dataset of over 140,000 expert-labeled patent pairs, we train a neural network that induces a representation space optimized for finding anticipatory prior art in a large corpus of noisy data. Our highly interpretable model achieves near-perfect separation between anticipatory and non-anticipatory patent pairs and significantly outperforms the industry standard in both direct model scores and a simulated patent search example. Finally, we demonstrate how these models can get deployed at scale and are used to power a suite of patent intelligence tools.

1 INTRODUCTION

The field of patent law is largely characterized by the assessment of novelty and non-obviousness between a single focal invention and a *corpus* of prior art. If a reference, or a combination of references, in the prior art anticipates an invention or renders it obvious, we define a legal *overlap* between the invention's patent and the reference(s)[1]. In patent prosecution, this manifests as drafting a patent that claims the broadest possible scope of patentable technology without overlapping with any of the prior art. Meanwhile, in patent litigation, a common goal is to demonstrate that such an overlap does (or does not) exist between the contested patent and the prior art.

Two sources of complexity dominate this problem: how to define the relevant corpus and how to formalize overlap. In this study, we avoid much of the complexity of the former by picking the corpus to be all US patent grants and pre-grant publications published post-2000. We argue that this choice is not only well-motivated by existing patent-law applications, but also reflective of any commonly used corpus in the space. The more fundamental challenge lies in defining what constitutes as overlap between a patent's claim set and another document. Understanding this relationship is the core of how novelty, infringement, and innovation are assessed, and it is the primary focus of this work.

Current tools, such as the TotalPatentOne software from LexisNexis, the Patent Public Search by the US Patent and Trademark office (USPTO), or Patsnap's search tool rely on humans to define this overlap through hours of keyword queries and reading through hundreds of patents. Despite decades of research progress in the natural language processing space, the AI search tools currently in use in the industry can be broken into just two categories: (1) opaque deep research engines that frequently hallucinate and cannot be effectively trained to understand strict legal overlap or (2) models forced to rely on legacy keyword based search methods. As a result, there is space for significant contributions by taking disciplined data-driven NLP techniques and deploying them at scale.

[1] For a more formal definition, see Section 3.1

In this paper, we present a contrastive data-driven approach to learning a representation for patents that captures the structure required to understand the notion of novelty, infringement, and innovation. We demonstrate that not only does this representation space have a sharp, well-defined cutoff for classifying overlap, but we're also able to use this representation to dramatically outperform leading semantic search models. Additionally, we provide an interpretability analysis demonstrating alignment between the model's representations and expert-driven patent analysis. Lastly, we show that this representation scales to real-world deployments, enabling a search engine that vastly outperforms the current industry standard.

2 Relevant Work

Early approaches to patent searching, many of which are still widespread today, focus on manual keyword queries and boolean logic hand-crafted by patent examiners and attorneys. These approaches rely on human experts to carefully craft queries with a high recall rate, leading to the manual parsing of hundreds of patents. There has been much work put into enhancing this initial keyword query (Magdy & Jones, 2011), such as adding synonyms using thesauruses (Lupu et al., 2017), mining keywords from initial search results to boost the original query (Tannebaum & Rauber, 2015), and using citations to augment and enhance the original query (Fujii, 2007). Even with these enhancements, early approaches suffer from requiring dozens of keywords cast as a large net to ensure a high recall rate, resulting in many hours of manual work reading through hundreds of patents.

By the mid 2010s, researchers began applying natural language processing techniques to the space of patents, focusing on semantic text similarity. Taking inspiration from deep learning methods used in text-based patent classification (Grawe et al., 2017; Tran & Kavuluru, 2017; Yun & Geum, 2020), common approaches included creating patent-specific word-embeddings (Risch & Krestel, 2019) and developing models tuned to measure the similarity score of two patents (Helmers et al., 2019).

Today, the focus has shifted toward transformer-based large language models which are either fine-tuned on patent text (Lee & Hsiang, 2020; Bekamiri et al., 2024) or built as deep research engines (Perplexity Team, 2025; OpenAI, 2025). However, these approaches each have practical limitations that prevent them from being deployed at scale. Wrapping fine-tuned, patent-specific LLMs in large prompting frameworks increases susceptibility to hallucinations (Kalai et al., 2025) and provides limited interpretability into the reliability of the resulting outputs. Furthermore, using third-party deep research engines makes it impossible to train models to recognize strict legal overlap as opposed to mere semantic similarity, leading to noisy and unreliable results.

As a result, the industry is still dominated by early approaches which are built on legacy databases and searched using manual keyword queries. In practice, the AI tools in industry simply use LLMs to output these queries instead of humans, which inherit many of the same limitations as keyword-based search. The few examples of semantic tools that exist, like Patsnap, Orbit Intelligence, or Perplexity patents struggle with unreliable results due to a lack of data-driven training and validation.

3 Methods

3.1 Patent search as classification

In our paper, we take a disciplined information retrieval approach to searching, outlined in Figure 1; given a query, q, a ranking is performed over every document in the corpus $d \in \mathcal{D}$ using a scoring function $f(q, d)$. The search results are created by ordering documents according to $f(q, d)$, with the scoring function constrained so that higher scores correspond to greater overlap with the query.

In patent law, overlap is defined under 35 U.S.C. §§ 102 and 103, where the subject matter disclosed in one document anticipates or renders obvious the subject matter of another. Therefore, we characterize the perfect score function $\tilde{f}$ such that $\tilde{f}(q, d) = 1$ if q and d violate 35 U.S.C. §§ 102/103 and $\tilde{f}(q, d) = 0$ otherwise. [2].

[2]The scoring function $\tilde{f}$ must characterize the full legal complexity of 35 U.S.C. §§ 102 and 103, including but not limited to (i) the statutory exceptions to prior art under 102(b)(1), 102(b)(2), and 102(c); (ii) differences between pre-AIA and post-AIA statutory regimes, depending on when the patent application was filed; and (iii) Judicial doctrines that affect claim scope and comparison, such as the Doctrine of Equivalents.

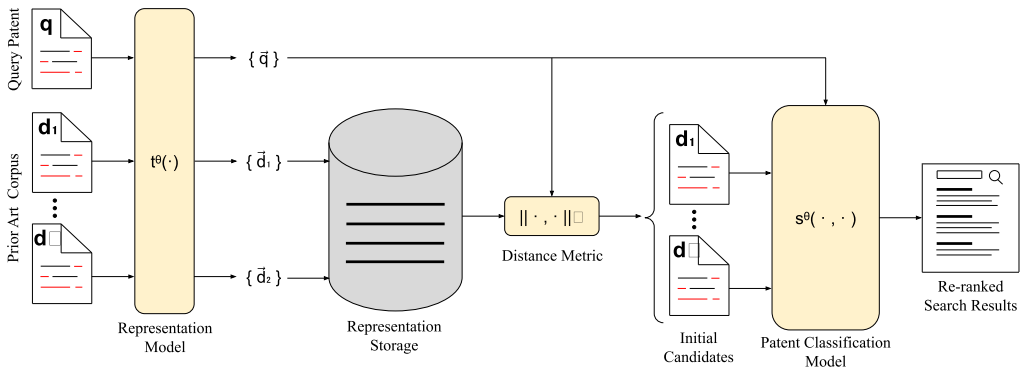

Figure 1: A sketch of the overall patent searching method, demonstrating representation generation from a patent corpus, initial candidate generation using a distance metric over the representations, and re-ranking based on a finetuned patent classification model.

Now, our goal is to learn the scoring function $\tilde{f}$ using a model

$$\tilde{f}(q, d) = s_\theta(q, d) \tag{1}$$

Training proceeds by minimizing a loss that penalizes deviations of $s_\theta(q, d)$ from the ideal score $\tilde{f}(q, d)$:

$$\mathcal{L}(q, d) = ||\tilde{f}(q, d), s_\theta(q, d)||_m \tag{2}$$

where $||\cdot, \cdot||_m$ denotes a distance metric. Importantly, the fact that $\tilde{f} \in \{0, 1\}$ means that we can characterize the patent search problem as a classification problem between positive, overlapping patents and background, non-overlapping patents. This allows us to cast patent search as a supervised classification problem whose outputs naturally induce a ranking over documents.

A direct consequence of this formulation is that the learned scoring function is prior art database agnostic, assuming the training corpus is representative of the prior art corpus in any specific deployment or use. Importantly, this entails that practical implementations of the learned scoring function require that the underlying document database reflects the full scope of prior art relevant under 35 U.S.C. §§ 102 and 103 to avoid systematic coverage bias.

3.2 Patent classification at scale

As of 2025, there are over 14 million patent grants and pre-grant publications in the US post 2000, meaning that for every search, we would have to perform a forward pass on our model s_θ 14 million times. Additionally, s_θ must not only capture the human language, but also the nuances of patent law in thousands of different domains, causing it to be quite large. The size of the model combined with the size of the corpus make any direct approach infeasible.

Instead of directly using s_θ over patent inputs, we generate representations of a specific patent p using a set of feature vectors $p_r = \{\vec{p}_r^{\,1}, \vec{p}_r^{\,2}, ..., \vec{p}_r^{\,n}\}$. The scoring function $s_\theta(\cdot, \cdot)$ is thus defined over patent representations rather than over raw patent documents. These representations are generated using a second model $t_\theta(p)$

$$p_r = t_{\theta'}(p) \tag{3}$$

Substituting this representation into the loss defined in Equation 2 yields

$$\mathcal{L}(q, d) = ||\tilde{f}(q, d), s_\theta(t_{\theta'}(q), t_{\theta'}(d))||_m \tag{4}$$

Training is then done by learning parameter sets θ and θ' which minimize the loss function $\mathcal{L}$.

Under this approach, the representation space is learned to extract features that are informative for better classification of patents that overlap. This means that the representations encode much of the domain-specific notion of overlap between patent-document pairs. For this reason, we can find distance metric $||\cdot, \cdot||_{m'}$ such that

$$s_\theta(t_{\theta'}(q), t_{\theta'}(d)) \approx_a ||t_{\theta'}(q), t_{\theta'}(d)||_{m'} \tag{5}$$

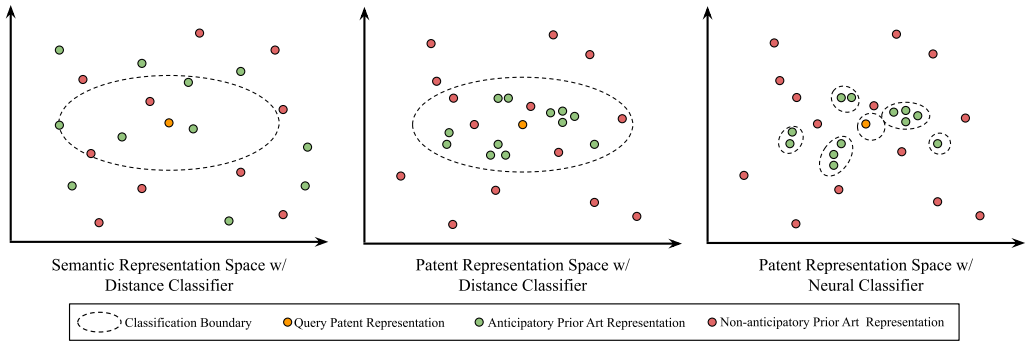

Figure 2: Comparison of different representation spaces used throughout the method, projected onto two dimensions and including samples of positive patents that anticipate the query patent and background patents that do not anticipate the query patent. The standard semantic representation space used by the industry standard (left), the trained patent representation space induced by $t_{\theta'}(\cdot)$ using the approximation metric $||\cdot, \cdot||_{m'}$ (middle), and the trained patent representation space induced by $t_{\theta'}$ using the patent classification model $s_\theta(t_{\theta'}(\cdot), t_{\theta'}(\cdot))$ (right).

where $||\cdot, \cdot||_{m'}$ is much faster to compute at scale than $s_\theta(\cdot, \cdot)$. Therefore, in deployment for every patent p in our corpus, we first pre-compute the patent representations $t_{\theta'}(p)$. Then for a specific query q, we use the distance metric $||q, \cdot||_{m'}$ to find the top n candidates which are then re-ranked by the true model, s_θ. An illustration of the induced representation spaces for these processes can be seen in Figure 2.

By tuning the parameter n in accordance to the strength of the approximation, $\approx_a$, this approach allows us to efficiently perform complex patent searches with accuracy virtually equal to s_θ.

3.3 DATASET

The training and evaluation corpus consists of approximately $140,000$ hand-labeled patent pairs including both pre-grant and granted US publications drawn from public and licensed sources, including filings from the US Patent and Trademark Office (USPTO). We restrict the corpus to post-2000 USPTO filings, which ensures consistent metadata formats and focuses the model on patents still within or near their enforceable lifetime. Each patent pair is annotated by subject-matter experts who classify both the novelty of a claim and citation strength of the corresponding supporting or conflicting document. To ensure data quality, we included inter-annotator review and periodic spot-checks from data engineers and legal professionals. We define patent pairs in the following three segments:

- **Positive patent pairs** — Pairs of patents where the cited document *fully anticipates* or teaches the claimed invention, consistent with the standard of a 35 U.S.C. § 102 anticipation rejection.
- **Negative patent pairs** — Pairs that share overlapping concepts or technical language but do not meet the threshold of anticipation. These examples challenge the model to distinguish between superficial similarity and true disclosure.
- **Background (noise) pairs** — Unrelated patent pair combinations sampled uniformly from the corpus, serving as control examples to calibrate the model's discrimination boundaries.

During training, we adopt a contrastive learning approach that leverages positive and negative patent pairs, while evaluation is performed using positive and background pairs to better reflect real-world patent search conditions. This composition enables the model to learn fine-grained distinctions between genuinely anticipatory prior art and superficially related references, mirroring the analytical rigor applied by examiners and attorneys.

To demonstrate how patent pairs were labeled, representative excerpts showing *positive*, *negative*, and *background* patent pairs are presented in Appendix A.

The corpus was randomly split into an 80% training set and a 20% evaluation set. The split was performed blindly, and the evaluation data was never exposed during training or tuning, ensuring unbiased performance estimates.

4 RESULTS

4.1 SCORING METHODOLOGY

To evaluate model performance, each predicted similarity score was compared against a binary ground truth label $y \in \{0, 1\}$, where $y = 1$ denotes a positive pair and $y = 0$ denotes a background pair. This score was determined by subject-matter experts prior to evaluation. We use two complementary scoring methods in our evaluation approaches.

First, we compare the distributions of similarity scores assigned to positive versus background pairs in the labeled dataset. Clear separation between these distributions indicates that a model can reliably distinguish anticipatory prior art from irrelevant documents. This distributional analysis provides a global view of each model's ability to assign higher scores to truly related patents while suppressing noise.

Second, we evaluate retrieval behavior using a controlled simulated search scenario. For a given query patent with a known set of k positive pairs, we rank all documents by model score and count how many results must be examined before recovering each of the k relevant patents. This simulates real-world prior art search; models that retrieve all k references early in the ranking demonstrate higher practical utility and substantially reduced review burden.

Together, these two evaluations capture both the statistical and operational performance of the model.

4.2 SCORE DISTRIBUTIONS ACROSS METHODS

The three distributions in Figure 3 illustrate how each representation method separates positive and background patent pairs. These plots provide a quantitative view of how effectively each model assigns higher scores to anticipatory prior art and suppresses scores for unrelated documents. Greater separation between the two distributions corresponds to a more accurate scoring function and directly impacts downstream retrieval quality.

In the industry standard approach, we observe high scores for both positive and background pairs, as well as a long-tailed background distribution. This indicates limited discriminative power, as both positive and background pairs receive comparable scores and exhibit poor separation. In the context

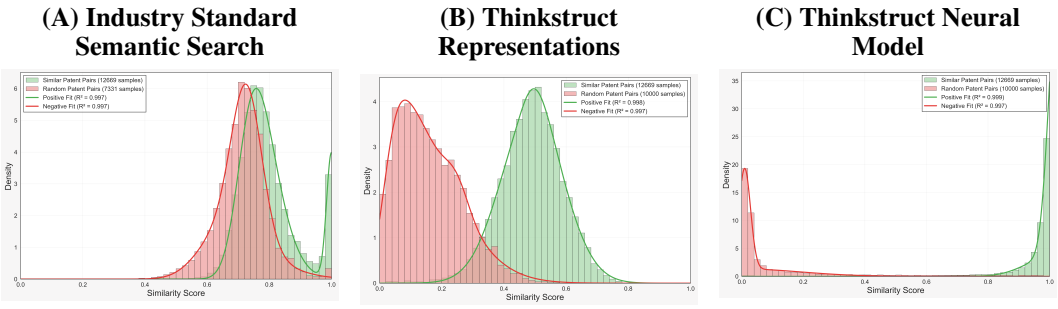

Figure 3: Comparison of similarity score distributions across three representation methods. Each subplot shows the distribution of similarity scores assigned to positive patent pairs (green) and background patent pairs (red). **(A)** The baseline embedding model exhibits heavy overlap between the two distributions, indicating poor separation between relevant and irrelevant prior art. **(B)** Thinkstruct's domain-specific representations reduce this overlap by lowering scores assigned to non-similar pairs while preserving high scores for similar pairs. **(C)** The full Thinkstruct neural model produces near-complete separation, demonstrating strong discriminative performance and minimal noise.

of a patent search where retaining high classification recall rates is important, the overlap between background and positive distributions manifests as elevated false positive rates, requiring extensive downstream filtering and manual review.

By utilizing domain specific representations, we can effectively suppress the background distribution, leading to increased separation between the two distributions. With the final addition of Thinkstruct's neural model, we observe a near-perfect separation between the positive and background distributions, indicating strong discriminative performance that dramatically improves false positive rates while keeping recall rates high.

4.3 Efficiency in Simulated Search

To understand how these distributional improvements affect practical retrieval performance, we evaluate how quickly each model surfaces known relevant documents in a ranked list. This simulated experiment mimics real prior art search, where practitioners care not only about similarity scores, but about how many documents must be reviewed before encountering legally meaningful prior art. As shown in Figure 4, the Thinkstruct model retrieves all 10 known relevant patents within the top 30 ranked results, whereas the industry standard approach requires examination of over 300 documents to achieve the same coverage. This corresponds to an improvement in retrieval efficiency by an order of 10 times over the industry standard approach, directly translating to reduced review burden and faster legal analysis.

4.4 Interpretability

Interpretability in the context of classification models refers to the extent to which humans can understand the factors that drive a model's predictions. In our setting, this corresponds to understanding how different features of the input patent pairs influence the model's similarity score. In addition to producing a scalar similarity score, Thinkstruct's neural network exposes intermediate *attention*

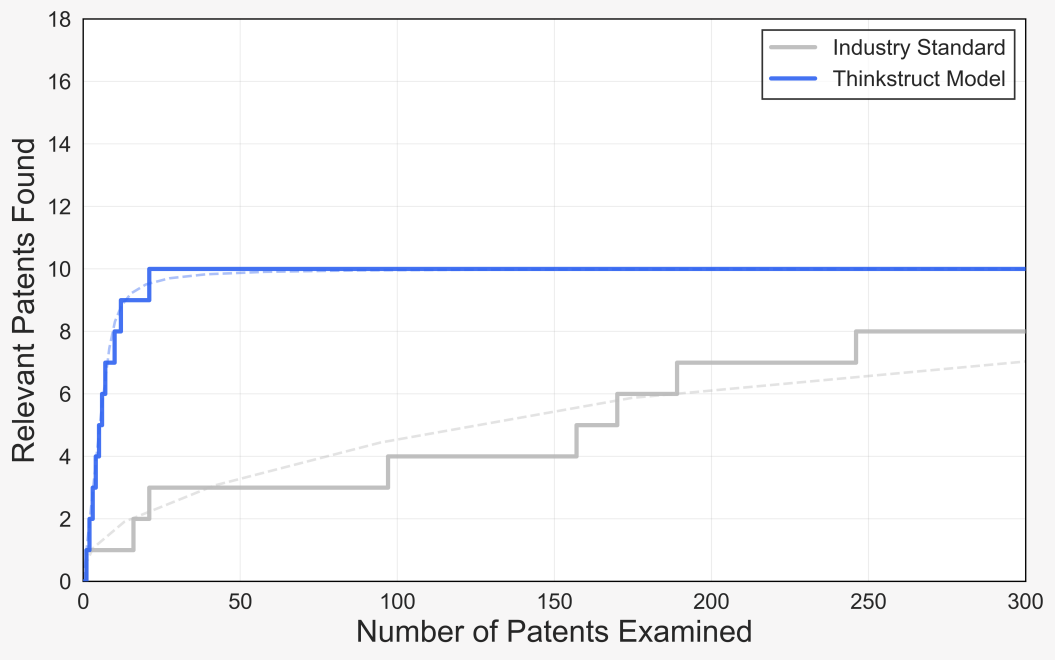

Figure 4: Cumulative retrieval performance for a query patent with 10 known relevant references. The x-axis shows the number of ranked results examined, and the y-axis shows how many of the 10 relevant patents have been retrieved at each cutoff. The industry standard model (gray) requires reviewing more than 300 documents to recover all relevant references, reflecting substantial noise in its ranking. In contrast, the Thinkstruct model (blue) retrieves all 10 relevant patents within the top 30 results.

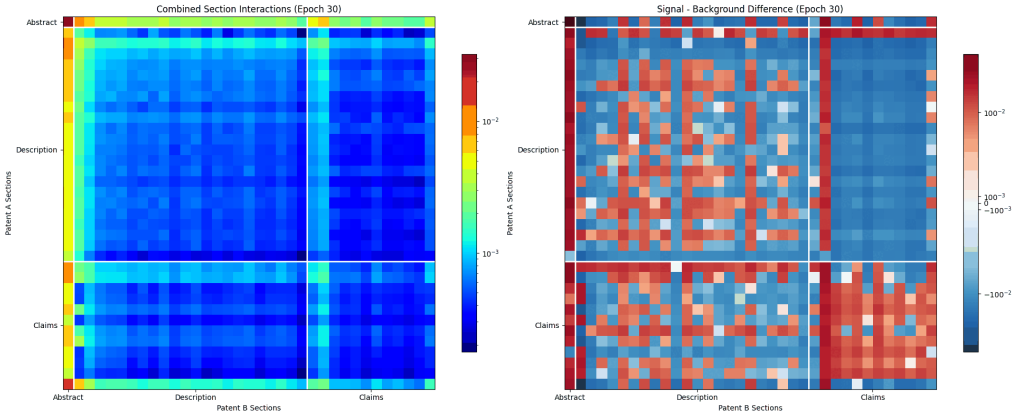

Figure 5: Attention-based interpretability heatmap for a representative patent pair at the 30th training epoch. The left panel shows the aggregate attention mass across section–section interactions between the two patents, while the right panel shows the difference in the aggregate attention mass between signal and background interactions. Warmer colors indicate interactions that contribute more strongly to the model's classification. The model assigns significant weight to interactions involving patent abstracts and claim sets, with comparatively weaker signal arising from description-level interactions.

weights over document segments. These weights indicate which portions of the input patents most strongly influenced the final similarity score, providing insight into how the model differentiates between relevant and background pairs and achieves strong score separation.

Figure 5 shows the distribution of attention weights averaged over the validation set at the 30th training epoch, illustrating the learned interaction structure between sections of paired patents. The left panel shows that the model assigns strong discriminative weight to interactions involving the abstract of one patent and the full structure of the other, indicating that abstracts function as a global reference point and likely are used to initially determine similarity. We also see a strong interaction between the claim sets, particularly among independent claims, reflecting their central role in defining patent scope. Description-level interactions exhibit comparatively weaker and less structured signal, suggesting that the model does not rely on broad topical similarity.

Together, these patterns demonstrate that the model's relevance scores are driven by localized, legally meaningful connections between the documents rather than diffuse textual overlap, much like a patent professional. The use of the abstract as a global reference point and the isolation of strong signal from the claim set closely align with how patent practitioners assess relatedness, providing an interpretable basis for the model's relevance judgments.

5 Implementation

The performance results described in Section 4 translate directly into the capabilities of the Thinkstruct platform. Underlying models like the model described in Section 3 power a suite of applications designed to streamline research and provide actionable patent intelligence. Each feature leverages verifiable and reproducible reasoning architectures, enabling accurate, legally meaningful results with minimal noise. The following subsections outline the primary modules and value propositions of the Thinkstruct system.

5.1 Invalidity Search

The invalidity search module represents the most direct application of the underlying retrieval engine. It identifies anticipatory prior art capable of invalidating or limiting a claim under 35 U.S.C. §102 or 103. By applying the patent-specific representations and neural ranking approach described in Section 4 along with deterministic filters to date prior art, the system retrieves highly relevant documents early in the ranked list, reducing the number of non-relevant results by an order

of magnitude. This saves hours of time and ensures that the results you receive provide an accurate and complete landscape of the patent space.

Reference Documents	Publication ID	Similarity Score	Invalidity
Systems and Methods for Autonomous Vehicle	US11812345B2	Very High	Invalidity
Machine Learning-Based Medical Image Analysis	US20230345678A1	Very High	Invalidity
Data Encryption Techniques Using Quantum Key	US11698754B1	Very High	Invalidity
AI-Assisted Legal Document Review	US20240218901A1	High	Invalidity
Adaptive Control Algorithm for Autonomous Drones	US20240123876A1	Low	Invalidity
Biodegradable Polymer for Medical Imposter	US11904567B2	Low	Invalidity
Energy-Efficient Solar Tracking Assembly	US11876543B2	Very High	Invalidity
Machine Learning-Based Fraud Detecti...	US20240100456A1	High	Invalidity

Figure 6: User interface view of our invalidity search results, which are all automatically mapped to produce claim charts.

5.2 Claim Mapping

The claim mapping module extends the retrieval framework from document-level similarity to limitation-level reasoning. It decomposes each claim into its constituent limitations and aligns these with corresponding evidence in reference documents. This alignment is guided by semantic representations of both legal and technical language, enabling interpretable mappings that show exactly where and how a reference anticipates or overlaps with a given claim. By transforming complex similarity scores into explicit textual mappings, Thinkstruct bridges the gap between algorithmic retrieval and the legal reasoning required for claim analysis.

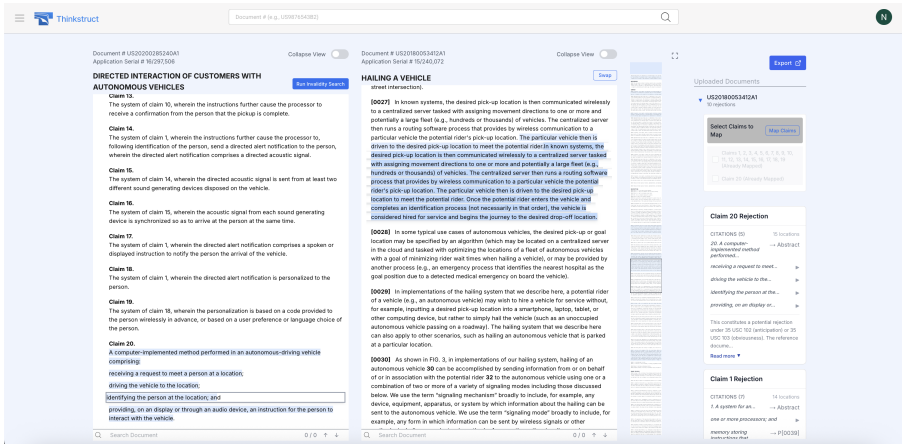

Figure 7: User interface view of our comparison page, allowing for automatic claim mappings with specific highlights and an exportable chart.

5.3 Infringement Monitoring

Thinkstruct's infringement monitoring module applies similar retrieval and reasoning architecture along with alternative models trained over datasets of products to produce results in a continuous, real-time setting. The system automatically scans the internet for newly published specifications, manuals, and other product-specific documents and compares them to a portfolio of protected claims to detect potential overlaps or encroachments. Using our own models, we filter out superficial linguistic similarities and focus on substantive claim overlap, reducing false positives in automated watch lists. This enables firms and in-house counsels to maintain continuous awareness of emerging infringement without the burden of manual review.

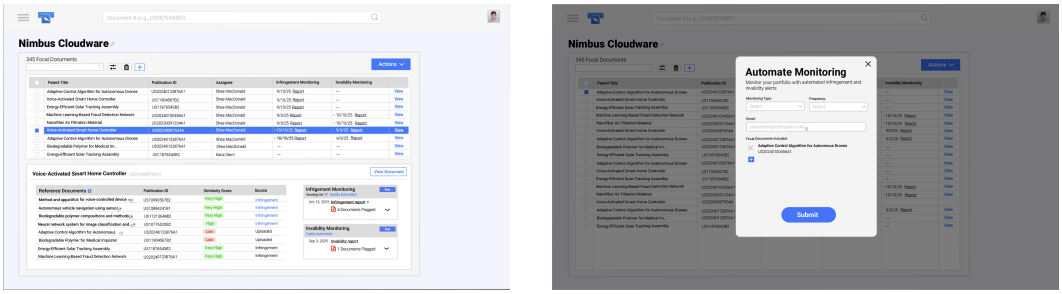

Figure 8: Screenshots of the Thinkstruct platform illustrating portfolio management and infringement monitoring.

5.4 Patentability Analysis

The patentability module generalizes the invalidity search beyond the patent corpus, incorporating non-patent literature such as academic papers, technical disclosures, and public databases. By ingesting heterogeneous data sources into a unified representation space, the model can evaluate novelty and prior art coverage across domains traditionally siloed from patent search tools. This broader evidence base supports stronger, defensible filings by identifying disclosures that would otherwise go undetected by patent-only search methods.

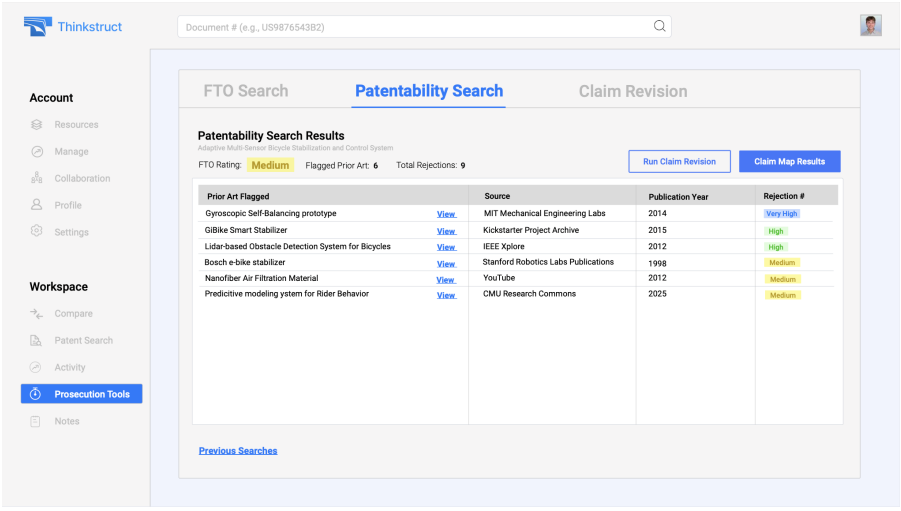

Figure 9: User interface view of the patentability feature, allowing users to check draft claims against prior art.

5.5 Freedom to Operate (FTO)

The Freedom to Operate module combines the patentability, invalidity, and infringement workflows into an integrated risk assessment tool. It evaluates whether a proposed product or technology is likely to infringe existing patents while also determining whether those patents themselves are valid in light of prior art. By unifying retrieval, mapping, and monitoring within a single analytical framework, Thinkstruct provides a holistic assessment of legal exposure, reducing uncertainty before commercialization and guiding strategic R&D decisions.

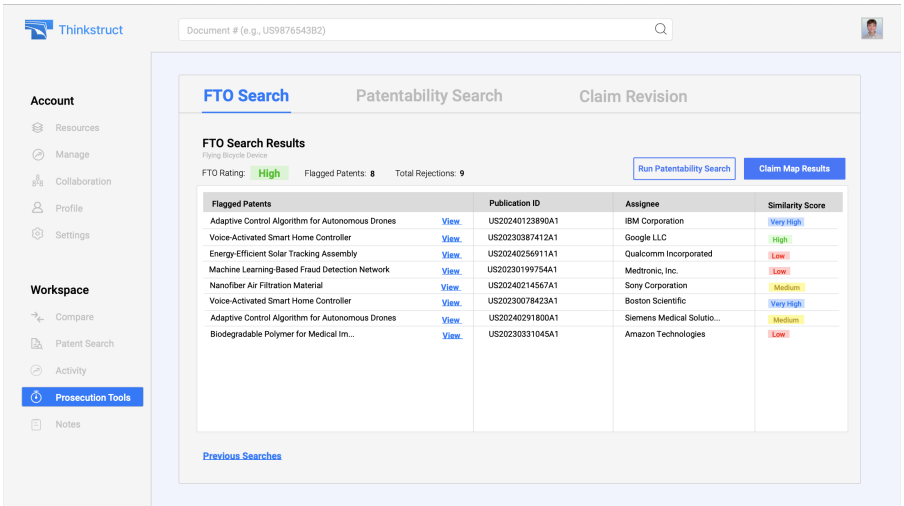

Figure 10: User interface view of the FTO feature, allowing users to check new inventions or potential ideas against patents, papers, and products to get a full risk evaluation on development of new technology.

Together, these modules demonstrate how Thinkstruct model architecture generalizes across distinct workflows. By adapting the retrieval and reasoning layers for each task, the Thinkstruct platform achieves both domain precision and operational scalability across the patent lifecycle, transforming what were once manual, fragmented analyses into an integrated, data-driven process.

6 Conclusion

We presented a disciplined data-driven information retrieval approach to patent searching, characterized by a classification model s_θ trained on 140,000 labeled data points. We demonstrated the ability of this model to vastly outperform the current industry standard, both in direct model scores and in a simulated patent search example. Furthermore, we interpreted the mechanisms of the model to show similar patterns to what is practiced by legal professionals. Lastly, we showed that our search can be implemented efficiently at scale, enabling the development of production-grade patent intelligence tools that materially improve how prior art is identified, evaluated, and monitored at scale.

Acknowledgments

This work was funded by Thinkstruct, including access to A100 and T4 GPU resources.

References

Hamid Bekamiri, Daniel S. Hain, and Roman Jurowetzki. Patentsberta: A deep nlp based hybrid model for patent distance and classification using augmented sbert. *Technological Forecasting and Social Change*, 206:123536, 2024. ISSN 0040-1625. doi: https://doi.org/10.1016/j.techfore.2024.123536. URL https://www.sciencedirect.com/science/article/pii/S0040162524003329.

Atsushi Fujii. Enhancing patent retrieval by citation analysis. In Wessel Kraaij, Arjen P. de Vries, Charles L. A. Clarke, Norbert Fuhr, and Noriko Kando (eds.), *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007*, pp. 793–794. ACM, 2007. doi: 10.1145/1277741.1277912. URL https://doi.org/10.1145/1277741.1277912.

Mattyws Grawe, Claudia Martins, and Andreia Bonfante. Automated patent classification using word embedding. pp. 408–411, 12 2017. doi: 10.1109/ICMLA.2017.0-127.

Lea Helmers, Franziska Horn, Franziska Biegler, Tim Oppermann, and Klaus-Robert Müller. Automating the search for a patent's prior art with a full text similarity search. *PLOS ONE*, 14(3): 1–17, 03 2019. doi: 10.1371/journal.pone.0212103. URL https://doi.org/10.1371/journal.pone.0212103.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language models hallucinate, 2025. URL https://arxiv.org/abs/2509.04664.

Jieh-Sheng Lee and Jieh Hsiang. Patent classification by fine-tuning bert language model. *World Patent Information*, 61:101965, 2020. ISSN 0172-2190. doi: https://doi.org/10.1016/j.wpi.2020.101965. URL https://www.sciencedirect.com/science/article/pii/S0172219019300742.

Mihai Lupu, Florina Piroi, and Veronika Stefanov. *An Introduction to Contemporary Search Technology*, pp. 47–73. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017. ISBN 978-3-662-53817-3. doi: 10.1007/978-3-662-53817-3_2. URL https://doi.org/10.1007/978-3-662-53817-3_2.

Walid Magdy and Gareth Jones. A study on query expansion methods for patent retrieval. *International Conference on Information and Knowledge Management, Proceedings*, 10 2011. doi: 10.1145/2064975.2064982.

OpenAI. Deep research system card. System card, OpenAI, February 2025. URL https://cdn.openai.com/deep-research-system-card.pdf. Accessed: 2026-01-13.

Perplexity Team. Introducing perplexity patents: AI-powered patent search for everyone, October 2025. URL https://www.perplexity.ai/hub/blog/introducing-perplexity-patents. Perplexity Blog.

Julian Risch and Ralf Krestel. Domain-specific word embeddings for patent classification. *Data Technologies and Applications*, 53, 03 2019. doi: 10.1108/DTA-01-2019-0002.

Wolfgang Tannebaum and Andreas Rauber. Patnet: A lexical database for the patent domain. In Allan Hanbury, Gabriella Kazai, Andreas Rauber, and Norbert Fuhr (eds.), *Advances in Information Retrieval*, pp. 550–555, Cham, 2015. Springer International Publishing. ISBN 978-3-319-16354-3.

Tung Tran and Ramakanth Kavuluru. Supervised approaches to assign cooperative patent classification (cpc) codes to patents. In *International Conference on Mining Intelligence and Knowledge Exploration*, 2017. URL https://api.semanticscholar.org/CorpusID:522219.

Junghwan Yun and Youngjung Geum. Automated classification of patents: A topic modeling approach. *Computers Industrial Engineering*, 147:106636, 2020. ISSN 0360-8352. doi: https://doi.org/10.1016/j.cie.2020.106636. URL https://www.sciencedirect.com/science/article/pii/S0360835220303703.

A PATENT PAIR EXAMPLES

This section provides representative excerpts from the annotated patent pair dataset used throughout the study. Each example consists of an anchor claim excerpt paired with a reference document excerpt, and is labeled according to the classification scheme described in Section 3.3. The examples are intended to illustrate the qualitative distinctions between positive, negative, and background (noise) patent pairs, rather than to exhaustively characterize the dataset.

Table 1: Representative examples from the annotated dataset

Pair Type	Anchor Claim (excerpt)	Reference (excerpt)
Background (Noise)	Fine particles comprising a polymer of a compound expressed by the following formula (1), wherein in the formula (1), R represents a hydrogen atom, a halogen atom, an alkyl group or an alkenyl group, "m" represents an integer of 1 to 10, and at least one of R represents the alkenyl group, wherein the fine particles further comprise a macromer.	The method of claim 1, further comprising: providing a user interface to a client device, the user interface comprising a display of one or more devices of the building automation system; and receiving a user selection of the archetypal device via the user interface.
Positive	The ultra-high heat flux chemical reactor of claim 1, wherein the steam reacts with both the biomass and the methane, but biomass and methane does not react with each other, and wherein a steam (H_2O) to carbon molar ratio is in the range of 1:1 to 1:4, and the temperature is high enough that the chemical reaction occurs without the presence of a catalyst.	In an embodiment, the biomass feed is in the form of solid particles. The biomass feed particles or droplets are entrained in a gas as they move through the reactor. In an embodiment, metal oxide particles are fed into the reactor concurrently with biomass particles. In another embodiment particles of a fossil fuel such as coal are fed into the reactor concurrently with biomass particles. In an embodiment, no additional catalyst is added to the biomass feed.

Pair Type	Anchor Claim (excerpt)	Reference (excerpt)
Negative	A moving assist method for assisting a vehicle which includes an internal combustion engine and an electric motor as a driving source when it moves from a current position to a destination, the moving assist method including: for each section obtained by dividing a traveling route from the current position to the destination, planning regularly or irregularly, by a mode planning unit, one traveling mode from a first mode of not maintaining a charge storage amount of a battery and a second mode of maintaining the charge storage amount of the battery, based on a traveling load associated with the section; and generating periodically, by an information generation unit, traveling load information that is referenced by the mode planning unit, the moving assist method comprising: when replanning the traveling mode based on the traveling load information generated by the information generation unit, deferring, by the mode planning unit, an execution of the replanning of the traveling mode a period to which from an execution of a previous planning does not reach a predetermined period, in addition to planning regularly.	Herein, the route information acquisition unit 3 can use, for example, the in-vehicle car navigation system to acquire the present location through a receiver (GPS sensor) for receiving position information from a satellite positioning system such as a GPS (Global Positioning System) or the like in the car navigation system and search for the route information from incorporated map data. Further, by connecting a portable terminal, a PDA (Personal Digital Assistant), or a smartphone carried by the driver or a fellow passenger to the vehicle energy management device 100, a navigation function (application) incorporated therein may be used as the route information acquisition unit 3.