Article

PAI-NET: Retrieval Augmented Generation Patent Network using Prior Art Information

Juho Bai * and Kyung Yul Lee

Posted Date: 20 February 2025

doi: 10.20944/preprints202502.1615.v1

Keywords: generative artificial intelligence; knowledge database retrieval system; document similarity; semantic search

 doi:10.20944/preprints202502.1615.v1

Article

PAI-NET: Retrieval Augmented Generation Patent Network using Prior Art Information

Kyung Yul Lee and Juho Bai *

College of Economics and Business, Hankuk University of Foreign Studies, 81, Oedae-ro, Mohyeon-eup, Cheoin-gu, Yongin-si, Gyeonggi-do, Republic of Korea
* Correspondence: **juho@hufs.ac.kr**

Abstract: Similar patent document retrieval is an essential task that reduces the scope of patent claimants' searches, and numerous studies have attempted to provide automated patent search services. Recently, Retrieval-Augmented Generation (RAG) based on generative language models has emerged as an excellent method for accessing and utilizing patent knowledge environments. RAG-based patent search services offer high utility as AI search services by providing document knowledge similar to queries. However, achieving satisfactory quality in similarity search services remains a challenging task, as search methods based on document similarity do not carefully consider the characteristics of patent documents. Unlike general document retrieval, the similarity of patent documents must take into account prior art relationships. To address this issue, we propose PAI-NET, a deep neural network for computing patent document similarities by incorporating expert knowledge of prior art relationships. We demonstrate that our proposed method outperforms current state-of-the-art models in patent document classification tasks through semantic distance evaluation on USPD and KPRIS datasets. PAI-NET presents similar document candidates demonstrating a superior patent search performance improvement of 15% over state-of-the-art methods.

Keywords: generative artificial intelligence; knowledge database retrieval system; document similarity; semantic search

1. Introduction

Knowledge retrieval systems in database-driven environments prioritize the delivery of query-relevant information. Within the patent domain, the integration of deep learning methods has accelerated the evolution of similar patent search capabilities. Although GPT-style decoder-only transformers demonstrate robust query responsiveness, they remain susceptible to hallucination. Retrieval Augmented Generation (RAG) systems have emerged as a preferred solution in knowledge-based expert systems, effectively mitigating hallucination while maintaining high-quality knowledge services. These systems continue to advance, leveraging patent databases to identify query-relevant documents and generate appropriate responses based on validated source material.

For patent document retrieval systems, selecting patent document candidates relevant to queries is crucial for service quality. Traditional patent search systems focus on applying deep learning models to patent classification systems based on registration criteria. Primarily, encoding model-type deep learning methods are being applied to classification tasks for automating similar patent search processes. Patent document classification methods based on Convolution Neural Networks (CNN), Transformer-Encoder based patent document classification methods, and patent classification methods utilizing hierarchical structures for large-scale patent documents have been proposed. Recent deep learning methods demonstrate superior performance in patent classification due to patent documents' typical textual document information characteristics. Despite the performance achieved through deep learning in patent classification tasks, similar patent searches still require expert knowledge for similarity evaluation tasks. RAG systems select similar document candidates through similarity evaluation. While cosine similarity is typically used for general similarity assessment, showing

 doi:10.20944/preprints202502.1615.v1

high similarity for content with similar structure and content, patent document similarity evaluation must consider technical relationships with prior art, unlike general document similarity evaluation. Although image classification methods have been proposed to reduce similarity distances between similar information for robust query handling, similarity distance learning methods are not effectively utilized in document classification due to rare significant differences in document representation similarity. Meanwhile, sematic search tasks are characterized by the difficulty of determining semantic similarity between documents based solely on document class information. Experts traditionally use ontological models like synonym dictionaries for similarity evaluation. However, ontological models present challenges in continuously expanding environments like patent information due to the ongoing expert resource update costs.

This paper proposes a novel model for selecting appropriate candidates for RAG-based patent document knowledge retrieval services. We utilize prior art relationship information instead of ontological information to reflect expert knowledge in similarity evaluations. Similar Patent Search Network using Prior Art Information (PAI-NET) is a contrastive learning-based multi-tasking model that applies similarity to prior arts in classification task models. Figure 1 illustrates the architecture of our Semantic Patent Knowledge RAG (Retrieval-Augmented Generation) system, which emphasizes semantic similarity analysis between priority art documents. The system comprises a retrieval agent, PAI-NET, prompt engineering module, and Large Language Model (LLM), with a particular focus on analyzing deep relationships between prior art documents.

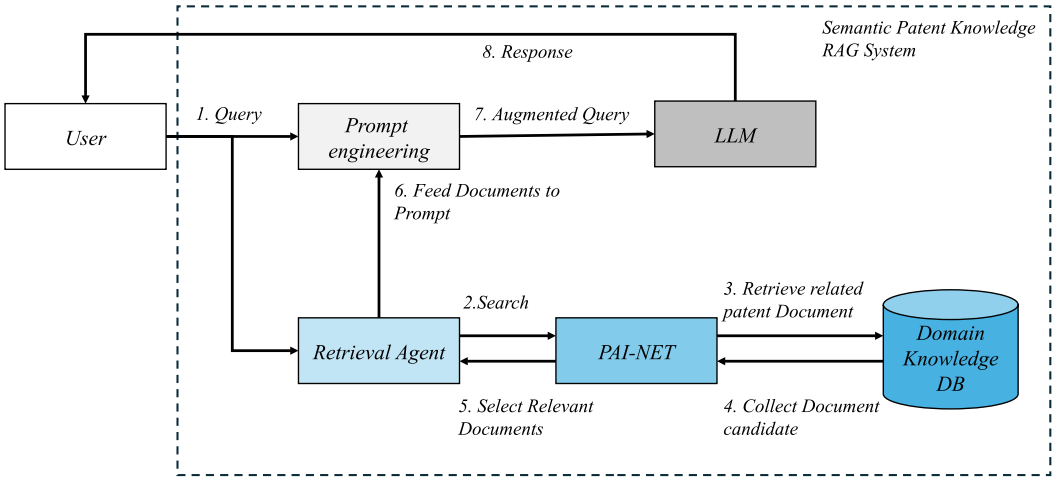

Figure 1. A concept of RAG(Retrieval Augmented Generative) System for Patent knowledge Querying

The core innovation lies in the interactions between steps 2-5, where the retrieval agent and PAI-NET collaborate to establish semantic relationships between prior art documents. When PAI-NET searches for prior art documents in the Domain Knowledge DB, it goes beyond simple keyword matching to understand the semantic connections between documents. Specifically, it calculates similarities by considering relationships between patent claims and specifications, citation networks, and technical field hierarchies. This approach ensures that retrieved prior art documents form an interconnected knowledge network rather than a collection of isolated documents. The Domain Knowledge DB functions as a knowledge graph that encodes complex relationships between prior art documents, rather than serving as a simple document repository. PAI-NET leverages this structured information to identify the most relevant set of prior art documents for a given query. This process considers not only technical features of patents but also filing dates, citation relationships, and technological evolution patterns within the field. Before being passed to the LLM(large language model) through the prompt engineering module (steps 6-7), the retrieved documents are structured according to their semantic relevance. This enables the LLM to comprehend the complex relationships between prior art documents and generate more accurate and contextually appropriate responses (step 8). This architecture overcomes the limitations of traditional RAG systems in patent search applications. By

 doi:10.20944/preprints202502.1615.v1

performing sophisticated analysis of semantic similarities between prior art documents, it provides more comprehensive and accurate patent search results. Furthermore, this deep document analysis significantly contributes to the thorough identification of relevant prior art documents during patent examination processes. In this process, our model applies a method of generating similar document groups based on prior art information between patents to extract query-relevant candidates. When these similar document candidates are presented, the service is provided by inputting prompts combining queries and candidate documents into generative language models to compose appropriate responses from candidate documents. Our main contributions are summarized as follows:

- PAI-NET is a novel model for similar patent search that improves document similarity evaluation performance by incorporating expert knowledge into similarity metrics.
- We demonstrate that prior art information can enhance similarity search performance without utilizing expert ontological information.
- PAI-NET performs both classification and similarity learning tasks while maintaining computational costs comparable to traditional classification-only models.
- We analyze and evaluate PAI-NET through extensive experiments on real patent datasets, demonstrating significant performance improvements in similar patent search tasks.

2. Related Work

2.1. Applying Retrieval Augmented Generation to Expert Domains

Domain adaptation of Retrieval Augmented Generation (RAG) for expert knowledge areas has emerged as a critical research direction, as conventional LLMs often struggle with specialized knowledge. Early approaches focused on joint training of retriever and generator components to adapt to domain-specific knowledge bases, showing notable improvements in specialized areas such as healthcare and news domains [1]. Building on this foundation, researchers have developed more sophisticated frameworks like PEER, which systematizes domain-specific tasks through precise question decomposition and advanced information retrieval while maintaining data privacy and cost efficiency [2]. The challenge of professional knowledge services has led to innovative approaches such as Knowledge Augmented Generation (KAG), which addresses the limitations of traditional RAG systems by incorporating knowledge graphs and enhancing logical reasoning capabilities. This approach has demonstrated substantial improvements in professional domains across multiple benchmark datasets [3]. For specialized domains with unique characteristics, such as Korean Medicine, researchers have developed Prompt-RAG, a vector embedding-free approach that has shown superior performance in terms of relevance and informativeness compared to conventional RAG models [4]. Recent research has also focused on specific challenges in handling tabular data within expert domains. The Tabular Embedding Model (TEM) has been developed to address the limitations of standard embedding models when dealing with complex numerical and tabular data, providing a more efficient solution for domain-specific applications [5]. Additionally, studies have shown that combining domain-specific fine-tuning with iterative reasoning mechanisms can significantly enhance question-answering accuracy in professional contexts [6]. While there have been studies [7,8] showing that RAG systems outperform general-purpose AI systems like GPT-4 in patent search applications, there remains a persistent demand for systems that better reflect the unique similarity characteristics of patent knowledge information.

2.2. Automated Patent Classification Methods

Among the various deep learning studies, models that show superior performance in text classification tasks are based on the Transformer method [9–14]. Deep Learning Approach is also making valuable achievements in classifying patent documents. DeepPatent [15] uses word2vector embedding methodes [16] and CNN methods at sentence level [17]. This method has the advantage of applying visual representation methods to the feature of words. The RNN-based method [18] utilizes a bidirectional recurrent model [19], which performs document classification tasks considering the flow of

 doi:10.20944/preprints202502.1615.v1

words. Meanwhile, the PatentBERT [20] is designed to take into account essential parts of the input document using the self-attention technique [9,10], which can be weighted to important keywords to reflect the interests of the document. Hierarchical Feature Extraction Model (HFEM) [21] and a Multi-Stage Extraction Network for Patent Document Classification(MEXN) [22] adopts a method of dealing with the entire document as a way of extracting and summarizing features of the document portion hierarchically. Using hierarchical models is a helpful method for considering whole data. However, it is unlikely that the evaluation will be reversed in the next step for the undervalued part in the previous step. The value of the input part may not be adequately evaluated depending on the size of the input range considered in the first step. Meanwhile, there are studies that classify various forms of drawing images [23,24] and studies that find drawing images in patent documents [25] at the individual classification level. So far, deep learning studies for patent documents have focused on classification tasks, which still pose challenges in improving the quality of services for use in similar search services.

2.3. Document Similarity

Latent semantic indexing (LSI) is one of the classic methods for document retrieval [26]. Search using LSI is a method of mapping the contents of a document to a latent space and searching for similar documents using euclidean operations between mapped features. Unlike methods with keywords as queries, these methods return results with close euclidean distances even if there are no accurately matched values. In contrast, features with close distances in potential space have the advantage of returning documents with similar meanings even if there are no identical keywords in real documents. Using these points, we can also perform nearest neighbor queries on similar documents using similarity metrics, such as cosine similarity. To provide results for nearby candidates, even in Latent spaces, where we project the features of documents using deep learning models. In this case, similar distance measurements between similar documents can be used as a measure of service quality [27], as high similarity documents must be kept at a close distance. This approach can be particularly effective in measuring patent document similarity, where technical concepts and innovations may be described using different terminology while maintaining similar underlying meanings. The ability to capture semantic relationships in the latent space makes it especially valuable for patent analysis and retrieval tasks.

2.4. Contrastive Learning Approaches

Studies of Contrastive Learning methods on image information [28] have been very meaningful in creating robust predictive models in the field until recently. Contrastive learning methods are metric learning methods that learn by comparing similar data, which can be mainly utilized in self-supervised learning. They have been applied in a variety of approaches.

In this regard, studies applying these methods in NLP have also been tried [29–32], which is difficult to distinguish compared to image information because of its different meaning. Relevant studies use fine-tuning methods using pre-trained models to compensate for these problems, which also seek to improve performance by following this trend. In particular, contrast learning methods utilizing similar pairs of documents are an important topic in this work. They can reflect users' perspectives on similarity in screening similar documents in more detail within the prediction category of classification tasks. In this regard, the most recent excellent work [31] proposes a method to reduce similarity distances based on classification predictions. However, in our work, we propose a model more suitable for real-world similar search environments by leveraging information from the prior art that reflects direct patent expert knowledge.

While there are patent document classification studies using various deep learning methods, methods to yield candidates for similar documents in patent domains are still challenging. The problem is that patent documents have an imbalanced data distribution between labels in a multi-label environment. In addition, the meaning of patent similarity can be applied differently from the similarity of general-purpose documents. We consider the characteristics for similarity evaluation. Our

 doi:10.20944/preprints202502.1615.v1

work aims to design a robust deep learning model that can improve classification task performance and recommend highly similar document candidates even in the imbalanced dataset domain [33].

3. PAI-NET: RAG Patent Network Using Prior Art Information

RAG Patent network using Prior Art Information(PAI-NET) is designed for similar patent search services. The PAI-NET framework has a multi-tasking structure that learns the similarity distance between documents and classification tasks together. To this end, PAI-NET has a conjured triple encoder structure for multi-tasking tasks and utilizes objective functions for similarity distance learning tasks and classification tasks.

Figure 2 shows how proposed method utilizes prior art information. Figure 2a shows the relationship between the prior art document and the claimed patent(target document) as domain knowledge, the most technically similar relationship with 1 to 1 matching. Meanwhile, Figure 2b shows how to generate traditional ontological structures. In Figure 2b, in addition to prior art relationships, patent experts leverage domain knowledge to further apply the linkages between each document to generate a searchable graph structure. On the other hand, in Figure 2c, the proposed method combines non-similar patent document information with the prior art pair. It uses it to consider the similarity distance between the prior art and target patent document relatively in the learning task.

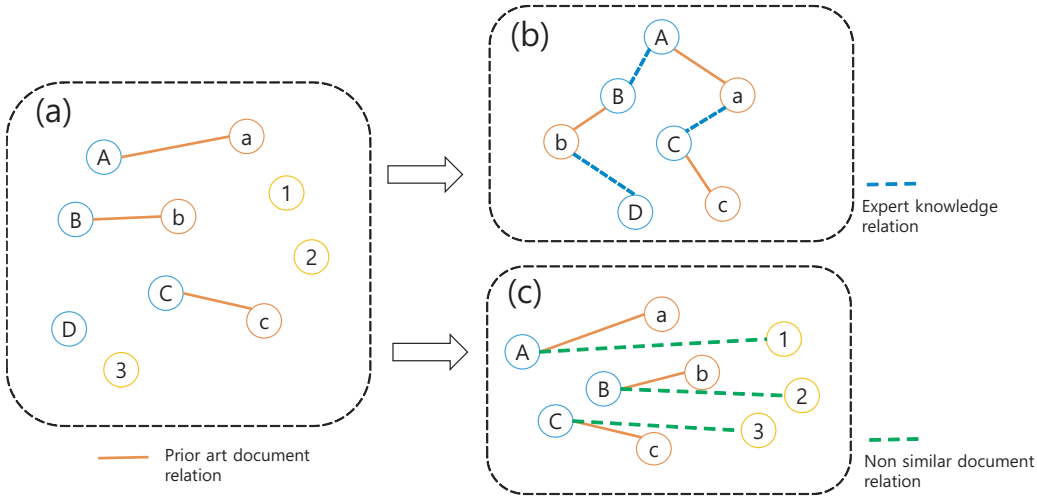

Figure 2. A concept of patent document connections using domain knowledge information relationships : (a) is prior art and target document relationships. (b) is traditional ontology structure with expert knowledge. (c) is contrastive triple pair with prior art and non similar document.

Our framework uses the set S of three different documents in parallel at one input step. The three document inputs are used as anchors S_a, positives S_p, and negatives S_n, respectively. And each document is transformed into a set of embedding tokens through an embedding process [10]. The encoder $\mathcal{F}$ uses three conjured Transformer encoders that share weights, and the embedded document set S_i entered are converted into document feature set h_i of h_a, h_p, h_n through the encoder corresponding to each location. Each transformed document feature h_a, h_p, h_n is used for classification tasks and similarity distance learning tasks. Document features that pass through the anchor encoder are used as inputs to the classifier $\mathcal{G}$ for classification objective function computing. Anchor document feature h_a is also used for similarity distance learning, where they are used for objective function computes and the positive document feature h_p and the negative document feature h_n. We design our framework without increasing computational time cost compared to existing single classification tasks using a parallel encoder batch process and share the weights of each encoder so that encoders can focus on achieving the goal of the objective function.

Figure 3 and the following equations represent the overall process:

 doi:10.20944/preprints202502.1615.v1

$$h_i = \mathcal{F}(S_i), \quad S_i \in \{S_{ai}, S_{pi}, S_{ni}\}, \tag{1}$$

$$v = \mathcal{G}(h_{ai}), \quad h_i \in \{h_{ai}, h_{pi}, h_{ni}\}, \tag{2}$$

$$p = Softmax(W_v v + b_v). \tag{3}$$

We describe how to combine pairs of documents in the preprocessing step and the embedding process at Section 3.1.

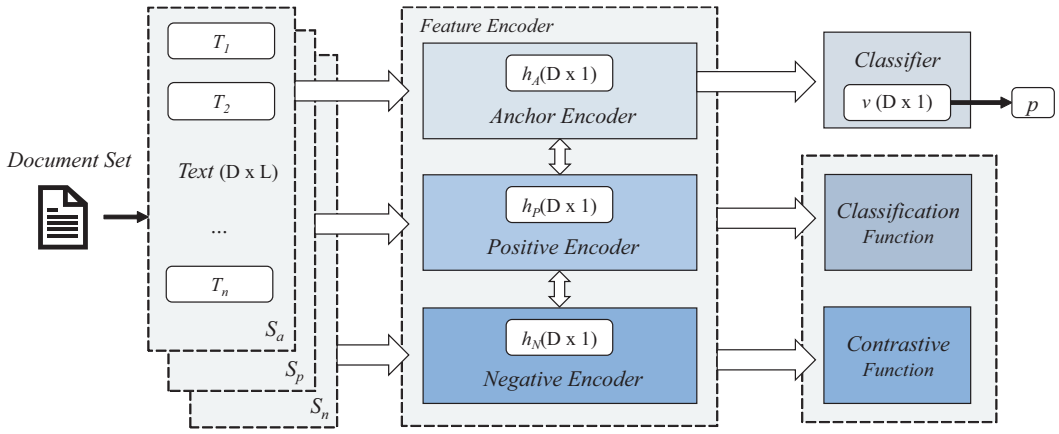

Figure 3. A overall concept of PAI-NET

The document embeddings after the preprocessing process are used as input to the encoder, (1) indicates feature encoding process for multi-tasking process which described in Section 3.2. (2) present classification task in Section 3.3, and the description of the objective function for multi-tasking is covered in Section 3.2. Lastly, the aggregated features pass through a final fully connected layer and softmax for label prediction learning in (3), while being utilized in parallel as features for document similarity evaluation.

 doi:10.20944/preprints202502.1615.v1

Algorithm 1 A pseudo code of PAI-NET

Input: Document Set $S = [S_a, S_p, S_n]$
Output: Document feature v_a
Initialization : initialize shared weight w of document encoders $\mathcal{F}_a$,$\mathcal{F}_p$,$\mathcal{F}_n$ and set α value
○ ***Training phase***
S_a = target Document set (**a**nchor)
S_p = the prior art document pair set of S_a (**p**ositive)
S_n = non relevant document set of S_a (**n**egative)
{▷process mini-batch size learning}
for each batch size s of S **do**
  $v_a = \mathcal{F}_a(S_a)$
  $v_p = \mathcal{F}_p(S_p)$
  $v_n = \mathcal{F}_n(S_n)$ {▷each v encoded by parallel process of $\mathcal{F}$}
  $\nabla w_{Classification} = L_{Classification}\ (v_a)$ by 8
  $\nabla w_{Margin} = L_{Margin}\ (v_a, v_p, v_n)$ by 9
  $\nabla w_{Total} = \nabla w_{Classification} + \alpha \cdot \nabla w_{Margin}$
  $w = w + \nabla\ w_{Total}$
  {▷update shared w of document encoders}
end for
○ ***Similar search phase***
{ ▷find similar candidate documents S_c with target S_a}
$v_a = \mathcal{F}_a(S_a)$
for all Candidate Set S_c **do**
  $v_c = \mathcal{F}_a(S_c)$
  get Top-K v_k of MAX($v_a \cdot v_c$)
end for
return Candidate S_c by using index of v_k

3.1. Pre-Processing

Patent documents have hierarchical classification labels. In general, patent documents with the same classification label have higher similarity as the classification level gets deeper. However, having the same label does not mean they similar because each patent document has different technical claims, even if it has the same classification category. We take this into account and utilize the records of prior art information that experts consider most similar to the target patent document in the patent registration or prior art investigation process. We design a method to compute the relative similarity distance of document features closely by using document information and classification information from each patent document to learn the model and use leading literature similar to the target document as inputs.Also, while patent document collections can be classified into multiple overlapping classes in a multi-class format, as shown in Figure 4, they have a highly skewed distribution, making it difficult to select similar documents based solely on similar classification attributes.

In our setting, we construct a triple-input document dataset by adding non-similar documents to consider the similarity between the prior art document and the target document in classification task learning. Given the challenges of quantitatively evaluating the technical similarity between patent documents, a method that considers relative similarity distances in learning is a detour yet reasonable method for searching similar documents.

Let $\mathcal{T} = [T_1, T_2, ..., T_N]$ and $\mathcal{L} = [L_1, L_2, ..., L_N]$ are the set of documents S and their labels. We use a triple pair dataset of inputs:

$$S = [S_a, S_p, S_n], \tag{4}$$

In Equation (4), we group the target document as S_a(anchor), the prior art document as S_p(positive), and documents that are not similar to the target document as S_n(negative) into one input pair. Before feeding the data, we split each document into word tokenizing with a fixed-size word length l, and all words are embedded to D dimensional features as $S_{a,p,n} \in \mathbb{R}^{D \times l}$. We add a classification token $[CLS]$ to the front of the token array of each embedded document. Using a classifi-

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 February 2025 doi:10.20944/preprints202502.1615.v1

cation token $[CLS]$ for classification tasks is an easy way to summarize the features of a document as a single length dimension.

We design a triplet conjured encoder as an encoding task method that extracts document features for document classification and similarity learning tasks. Each encoder uses a stacked encoder such as a Transformer mechanism [9], which extracts the document feature as:

$$\mathcal{F}(S_i) = \sum_{l=L-s}^{L} \mathcal{A}^l{}_S, \tag{5}$$

$$[CLS]_i^l = \mathcal{A}_S^l(Q_i^{l-1}, K_i^{l-1}, V_i^{l-1}), \tag{6}$$

where Q, K, and V indicate query, key, and value for the Transformer encoder, respectively. These feature vectors are also embedded by concatenating $[CLS]$ token and S_i as $Q_i = K_i = V_i = \{[CLS], S_i\}$. Here, $\{\cdot, \cdot\}$ denotes concatenation. L is the total number of stacked encoders, and l is the index of the stacked encoder. L and t determines how many encoder layers are used to summarizing paragraph S_i. Considering the efficiency of learning, we use contextualized word embedding method [34] and pre-trained language model as used by Encoding Transformer [10].

Each encoder receives input from the anchor S_a, positive S_p, and negative S_n embeded document and outputs summarized document features as h_a,h_p and h_n. The summarized document features are computed from the objective function and used for learning similarity distancing. Meanwhile, the anchor document feature h_a is used for classification tasks. Each encoder performs operations independently and in parallel. However, the three encoders share the same weight, which is inspired by relevant study [35]. In the encoding process of PAI-NET, the anchor encoder can consider the learning experience of each encoder in regulating the similarity distance between documents.

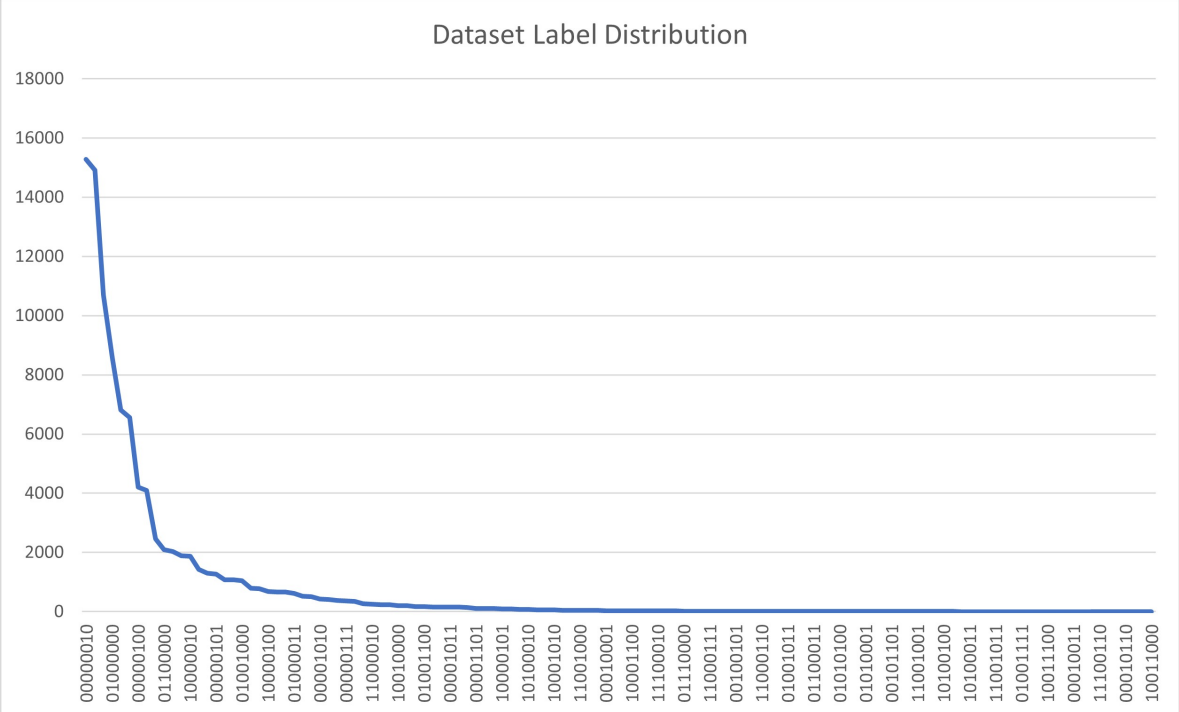

Figure 4. Dataset Label Distribution: x-axis is the label name and the y-axis is the number of labeled document

3.2. Objective Function

3.2.1. Total Loss

We use a sum of functions for classification tasks and functions for similarity distance computing to perform multi-tasking tasks. Recent works [36,37] have used the sum of classification and margin losses for the robustness of classification queries. Inspired by these parts, we use them for the

 doi:10.20944/preprints202502.1615.v1

robustness of similar document search queries. Total loss is expressed in a way that adds a percentage of margin losses to classification losses as:

$$L_{Total} = L_{Classification} + \alpha \cdot L_{Margin}, 0 \leq \alpha \leq 1, \tag{7}$$

where α refers to the coefficient of the ratio of the margin loss considered in the total loss. Margin loss values are adjusted below a certain percentage compared to classification loss values because margin loss information is used as supportive information for classification tasks. If the classification results are not the same label and yet in a similar relationship, the margin loss value can act as counter noise to the classification task results, which requires loss balancing.

3.2.2. Classification Loss

The classification task of patent documents is a multi-class multi-label problem, so we use the BCEWithLogitsLoss function for this as follows:

$$\begin{aligned} l(x, y) = L_{Classification} &= \{l_1, ..., l_N\}, \\ l_n &= -[y_n \cdot \log \sigma(x_n) + (1 - y_n) \cdot \log(1 - \sigma(x_n))], \end{aligned} \tag{8}$$

where target x and predicted label y for coordinate training loss.

3.2.3. Margin Loss

We use triplet loss to bring the similarity distance between the target document and the prior art document close. Triple loss controls the distance between the prior art and non-similar documents as:

$$L_{Margin}(a, p, n) = max\{d(a_i, p_i) - d(a_i, n_i) + margin, 0\}, \tag{9}$$

$$d(x_i, y_i) = \|x_i - y_i\|_p \tag{10}$$

where a_i, p_i and n_i indicate target document(anchor), prior art document(positive) and non-similar document(negative), respectively. The *margin* is a constant factor to control the distance between positive and negative features. If the margin value gets an increase, robust of the similar search query is enhanced. However, the size of latent feature space is limited, and we need to determine the range of a margin factor to get a query performance. In (10), the distance function uses Euclidean distance.

3.3. Document Classification

The end of the network is the single fully connected (FC) network for document classification. The output of the anchor encoder is fed into a 8-way $softmaxs$, which produces distribution over the size of section category(CPC).

3.4. Evaluation Metrics

To quantitatively assess the model's ability to rank similar documents highly, we employ the **Mean Reciprocal Rank** (MRR) metric. MRR effectively measures how well the model positions pre-defined relevant documents at higher ranks in a retrieval task. For a set of queries Q, MRR is formally defined as:

$$\mathbf{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{11}$$

where $|Q|$ is the number of queries, and $rank_i$ is the position of the first relevant document for the i-th query. The reciprocal rank $\frac{1}{rank_i}$ assigns higher scores when relevant documents appear at higher positions (e.g., 1.0 for first position, 0.5 for second position, etc.).

 doi:10.20944/preprints202502.1615.v1

In our evaluation framework, each query document is paired with one known similar document (positive) and multiple dissimilar documents (negatives). The MRR score reflects the model's ability to consistently rank the positive document above the negative documents. A higher MRR value indicates better performance, with a perfect score of 1.0 indicating that the model always ranks the similar document at the first position. This metric is particularly suitable for our task as it directly quantifies the model's effectiveness in identifying and prioritizing semantically similar patent documents in a retrieval context.

3.5. Implementation Details

We train networks for 10 epochs with an initial learning rate is set to 0.001 and divided by 0.1 every 10 epoch. Datatset is consist of 100,000 training and 20,000 validation documents. In our experiments, we use 768 dimensions of embedding vector size D for each word, and set the input size l to 100 words. We use a fine-tuning method with pre-trained model of Bert [10] for efficient learning.

We evaluate PAI-NET for 10 times with difference random seeds, then report the average performances. We use Adam optimizer [38] and apply early stopping method by EM score. We used a $7800X3D$ AMD CPU, 192 GB RAM, and RTX-4090 24GB GPU RAM and A100 cloud system for the implementation.

3.6. Ablation Studies

In this work, we have taken steps to determine the degree of margin distance and the ratio of margin loss values to combine classification and margin functions. To this end, we first use a method to determine the margin distance with low losses for classification tasks and then determine the ratio sequentially.

3.6.1. Margin Distance

When a document feature is projected into a latent region, the larger the distance between the classification sets of each document feature, the more robust the model can be seen. Thus, theoretically, the greater the margin distance, the greater the degree of robustness. However, since the margin distance cannot be increased indefinitely within a limited area, the experimental margin distance between the classification sets must be maintained appropriately, and the closer the margin distance between similar documents. In this regard, Table 1 demonstrates the rationale for selecting appropriate coefficients for the margin distance based on the classification performance of the document. Evaluation of the classification performance of documents is not used when training margin distances. However, it is reasonable to select margin distances that have the most negligible impact on classification results performance to combine them with classification tasks in a multi-task environment. Therefore, in this work, we compute the margin distance based on the label classification performance and choose three as the coefficient.

Table 1. Margin constant evaluation

Margin distance	MRR	D_p	D_n	$\lvert D_p - D_n \rvert$
2	0.8914	**0.881**	0.629	0.252
3	0.8913	0.713	0.022	0.691
3.5	**0.8939**	0.725	0.019	0.706
4	0.8889	0.756	0.031	0.725
5	0.8847	0.775	**-0.009**	**0.784**

Meanwhile, visualizing the dispersion of document features over margin distances is Figure 5, as previously described, the degree of overlap of the features is high, making it challenging to evaluate similarity or classification criteria between documents. On the other hand, if the margin distance is 10, it can be seen that document features are spread evenly throughout the area. However, there is a problem of poor cohesion according to the classification boundary.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 February 2025 doi:10.20944/preprints202502.1615.v1

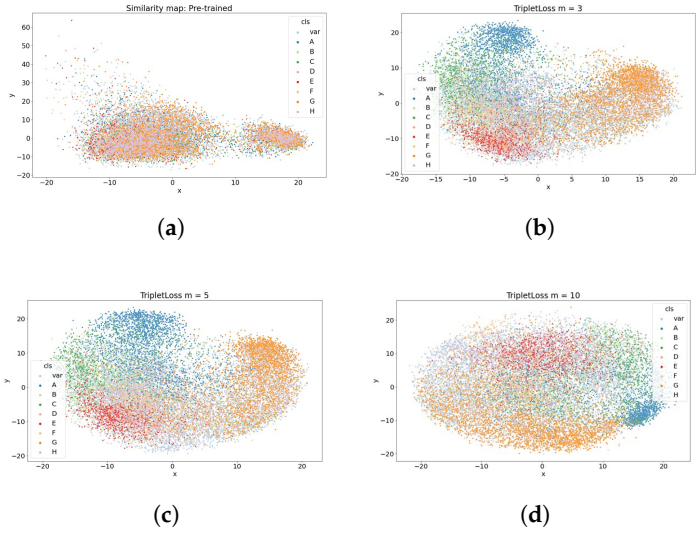

Figure 5. Visualization of margin (a) m = 0 , (b) m = 3 , (c) m = 5 , (d) m = 10

3.6.2. Margin Loss Ratio

Margin loss functions utilize datasets requiring expert similarity assessments, thus sufficient training data for each classification label must be secured to achieve classification performance through similarity learning alone. However, as patent datasets contain imbalanced label sets, margin loss and classification functions are combined for learning efficiency. Hence, experiments were conducted by varying the margin loss ratio from 0 to 0.5. Table 2 demonstrates classification performance (EM score, exact matching % score) when margin loss is added to classification loss at these ratios. As α increases, classification performance deteriorates since similarity data pairs are constructed independently of classification criteria. Higher values introduce noise into classification loss. When margin loss function ratios exceed certain thresholds, document classification performance can degrade. Therefore, ratios must be determined at levels appropriately reflecting each metric. Through analysis of performance metric variation rates, we determined α values within the 0.2 range.

Table 2. Margin Loss ratio

α	EM score	MRR	D_p	D_n	$\lvert D_p - D_n \rvert$
0.0	**60.935**	0.5856	0.653	0.006	0.647
0.2	60.545	0.7417	0.733	-0.044	0.777
0.4	57.231	0.7965	0.754	-0.052	0.806
0.5	56.735	**0.8124**	**0.768**	**-0.292**	**1.058**

3.7. Using Cosine Distance for Loss Function

The way of setting the distance in the margin loss function can consider Euclidean distance and cosine distance. Because the similarity of the document is based on the cosine distance, we can consider using cosine distance for margin distance calculations. In Table 3, we evaluate experiments by selecting a method that uses cosine distances in addition to margin distances to anchors and positive distances when using Euclidean distances and cosine distances. Table 3 shows that classification performance is best when using Euclidean distances, and similarity distances are best when using cosine distances. Applying distance using Euclidean distance in classification methods is a more efficient way to reflect the margin between classifications. On the other hand, the ability to distinguish similar documents is best shown in methods that reflect cosine distances between reference and prior art, apparently because margin distances only take into account the distances between preceding and non-similar documents.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 February 2025 doi:10.20944/preprints202502.1615.v1

Table 3. Cosine distance for loss function

	MRR	D_p	D_n	$\|D_p - D_n\|$
Euclidean	0.7417	0.713	-0.040	0.753
Cosine	0.6865	**0.820**	**-0.071**	**0.89**
Euc + Cos	0.7618	0.553	0.005	0.548
Euc + a-p Cos	**0.7627**	0.708	-0.037	0.745

3.8. Episodic Training

We perform performance evaluation as a way to regulate the learning dataset. In this regard, Table 4 compares PAI-NET with PAI-NET using the underlying dataset, a set of documents with only the same label compared to the baseline document, and PAI-NET$_n$ with the same label without considering similarity. We reveal that pairs of similar document sets with only the same label have lower classification performance or similarity than baseline datasets. They were insufficient to be written as appropriate learning sets for pairs of similar document sets with different labels.

Table 4. Comparisons with episodic training

	MRR	D_p	D_n	$\|D_p - D_n\|$
PAI-NET	0.7417	**0.713**	**-0.040**	**0.753**
PAI-NET$_p$	**0.7499**	0.744	0.016	0.728
PAI-NET$_n$	0.5625	0.619	0.003	0.616

On the other hand, classification performance was best when using different datasets with the same label without considering similarity, which can be seen as reflected in the classification feature even if it is not classified as data written in the existing self-supervision method.

3.9. Comparisons with Pretrained Model

To evaluate the performance enhancement of our proposed model, we conducted comprehensive experiments comparing PAI-NET with its pretrained baseline model through cosine similarity distances and Mean Reciprocal Rank (MRR) analysis. Our experimental findings demonstrate substantial improvements achieved by PAI-NET.

Specifically examining the cosine similarity distributions shown in Figure 6, the pretrained model exhibits minimal discriminative capability, with nearly indistinguishable similarity scores between positive (0.998 ± 0.001) and negative document pairs (0.997 ± 0.001). In contrast, PAI-NET demonstrates significantly enhanced discriminative power by establishing a clear demarcation between positive pairs (0.708 ± 0.199) and negative pairs (0.095 ± 0.216). This pronounced separation in similarity metrics indicates PAI-NET's superior ability to effectively differentiate between semantically similar and dissimilar patent documents.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 February 2025 doi:10.20944/preprints202502.1615.v1

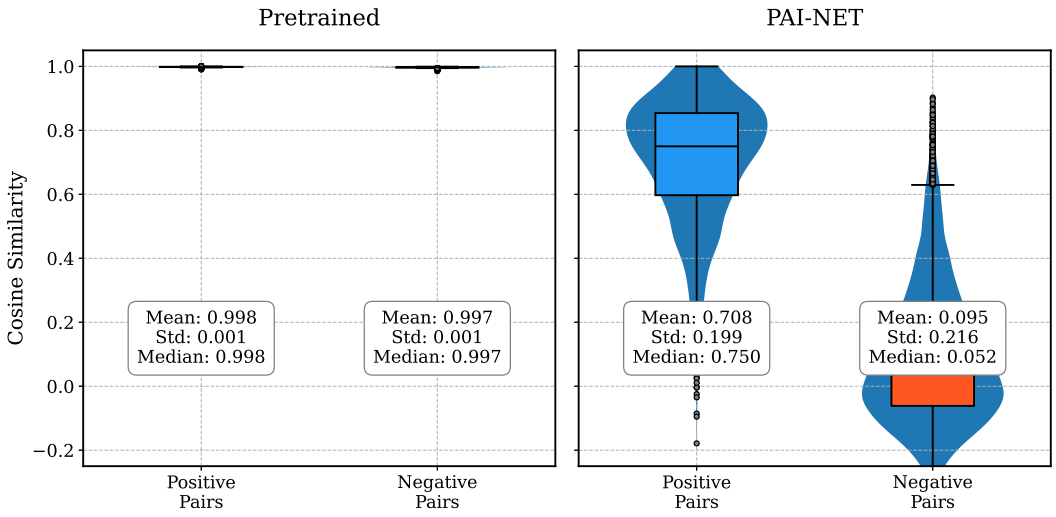

Figure 6. Cosine Similarity Distribution Comparison with Pretrained model

The quantitative assessment through MRR metrics, as presented in Table 5, further validates PAI-NET's enhanced performance. Our model achieves an MRR score of 0.954, representing a 9.8% improvement over the pretrained model's score of 0.869. This improvement in MRR metrics substantiates PAI-NET's enhanced capability in accurately positioning relevant documents at higher ranks during retrieval operations, thereby delivering more precise and reliable search results.

Table 5. Comparison of Pretrained and Finetuned Models

Model	Positive Pairs	Negative Pairs	MRR
Pretrained	0.998 ± 0.001	0.997 ± 0.001	0.869
Finetuned	0.708 ± 0.199	0.095 ± 0.216	**0.954**

3.10. Comparisons with State-of-the-Arts

We measure classification performance and cosine similarity distances together with recent good performance studies [20,30,31] (SOTA, state-of-the-arts). Evaluating the degree to which the patent document classification and similarity distance performance are satisfied simultaneously, the performance of this study, as observed in 6, shows superior results compared to existing studies. Among the relevant studies, CL+SCL [31] adds sampling considering distribution at the dataset stage because the model is constructed on the premise of self-supervised, which makes it challenging to create a suitable candidate group for classification labels within batch sizes. In PAI-NET using triple pair document datasets that did not consider label values, the similarity distance between similar and non-similar documents was the highest, indicating that the distance from similar documents became relatively close. We observe that our proposed method shows over a 15% improvement in MRR performance over the related SOTA methods.

Table 6. Comparisons with State-of-the-Arts

	Baseline			PAI-NET					
	patentBert [20]	SentenceBert (cls) [30]	CL+SCL [31]	*Euc*	*Cos*	*Euc* + *Cos*	*Euc* + *apCos*	*positive*	*negative*
R	0.5856	0.5617	0.5763	0.7417	0.6865	0.7618	**0.7627**	0.7499	0.5810

 doi:10.20944/preprints202502.1615.v1

4. Conclusions

We have presented PAI-NET, a novel framework for improving similarity search performance in Retrieval Augmented Generation (RAG) systems for patent knowledge management. Our approach effectively addresses the unique challenges of patent document similarity search and makes several significant contributions to patent information systems and knowledge management.

The core strength of PAI-NET lies in its ability to incorporate expert domain knowledge through prior art relationships, resulting in superior document recommendation performance. Our experimental results demonstrate a 15% improvement in similarity-based retrieval performance compared to state-of-the-art methods in the patent domain. This substantial enhancement in retrieval accuracy provides particular value for prior art search processes where identifying relevant patent information is crucial

A key innovation of our approach is the novel solution to knowledge representation in expert systems through the leveraging of prior art information rather than traditional ontological structures. This approach not only provides superior performance but also significantly reduces the costs associated with maintaining expert knowledge bases. The framework demonstrates sophisticated capabilities in understanding and utilizing complex relationships between patent documents, enabling more nuanced and accurate knowledge retrieval. The computational efficiency of PAI-NET's architecture is noteworthy, as it handles similarity learning tasks through fine-tuning without significant overhead. This efficiency is particularly valuable in practical applications, enabling robust service deployment even in environments with continuously accumulating patent document information. The implications of this work extend beyond patent systems to the broader field of expert knowledge systems. The methodology we've developed for incorporating domain expertise through document relationships shows promise for adaptation to other specialized fields where similar expert-verified relationships exist. This approach represents a significant step forward in reducing the human effort required for constructing and maintaining domain-specific knowledge bases.

Looking ahead, our research opens several promising avenues for future investigation. These include the potential adaptation of our framework to general knowledge management systems and the development of more flexible neural network architectures capable of accommodating evolving domain knowledge. Furthermore, we envision extending our approach to other types of expert systems where document relationships play a crucial role in knowledge organization and retrieval.

Author Contributions: Conceptualization, K.L. and J.B.; methodology, K.L. and J.B.; software, J.B.; validation, K.L. ; formal analysis, K.L.; investigation, K.L. and J.B.; resources, K.L. and J.B.; data curation, K.L. and J.B.; writing—original draft preparation, K.L. and J.B.; writing—review and editing, K.L. and J.B.; visualization, J.B.; supervision, J.B.; project administration, J.B.; funding acquisition, J.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Hankuk University of Foreign Studies Research Fund Of 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Siriwardhana, S.; Weerasekera, R.; Wen, E.; Kaluarachchi, T.; Rana, R.; Nanayakkara, S. Improving the Domain Adaptation of Retrieval Augmented Generation (RAG) Models for Open Domain Question Answering. *Transactions of the Association for Computational Linguistics* **2022**, *11*, 1–17. https://doi.org/10.1162/tacl_a_00530.
2. Wang, Y.; Li, X.; Wang, B.; Zhou, Y.; Ji, H.; Chen, H.; Zhang, J.; Yu, F.; Zhao, Z.; Jin, S.; et al. PEER: Expertizing Domain-Specific Tasks with a Multi-Agent Framework and Tuning Methods. *ArXiv* **2024**, *abs/2407.06985*. https://doi.org/10.48550/arXiv.2407.06985.
3. Liang, L.; Sun, M.; Gui, Z.; Zhu, Z.; Jiang, Z.; Zhong, L.; Qu, Y.; Zhao, P.; Bo, Z.; Yang, J.; et al. KAG: Boosting LLMs in Professional Domains via Knowledge Augmented Generation. *ArXiv* **2024**, *abs/2409.13731*. https://doi.org/10.48550/arXiv.2409.13731.

 doi:10.20944/preprints202502.1615.v1

4. Kang, B.; Kim, J.; Yun, T.R.; Kim, C.E. Prompt-RAG: Pioneering Vector Embedding-Free Retrieval-Augmented Generation in Niche Domains, Exemplified by Korean Medicine. *ArXiv* **2024**, *abs/2401.11246*. https://doi.org/10.48550/arXiv.2401.11246.
5. Khanna, S.; Subedi, S. Tabular Embedding Model (TEM): Finetuning Embedding Models For Tabular RAG Applications. *ArXiv* **2024**, *abs/2405.01585*. https://doi.org/10.48550/arXiv.2405.01585.
6. Nguyen, Z.; Annunziata, A.; Luong, V.; Dinh, S.; Le, Q.; Ha, A.H.; Le, C.; Phan, H.A.; Raghavan, S.; Nguyen, C. Enhancing Q&A with Domain-Specific Fine-Tuning and Iterative Reasoning: A Comparative Study. *ArXiv* **2024**, *abs/2404.11792*. https://doi.org/10.48550/arXiv.2404.11792.
7. Chu, J.M.; Lo, H.C.; Hsiang, J.; Cho, C.C. Patent Response System Optimised for Faithfulness: Procedural Knowledge Embodiment with Knowledge Graph and Retrieval Augmented Generation. In Proceedings of the Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024), 2024, pp. 146–155.
8. Wang, S.; Yin, X.; Wang, M.; Guo, R.; Nan, K. EvoPat: A Multi-LLM-based Patents Summarization and Analysis Agent. *arXiv preprint arXiv:2412.18100* **2024**.
9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in neural information processing systems, 2017, pp. 5998–6008.
10. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
11. Dai, Z.; Yang, Z.; Yang, Y.; Carbonell, J.G.; Le, Q.; Salakhutdinov, R. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In Proceedings of the Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 2978–2988.
12. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of the Advances in neural information processing systems, 2019, pp. 5753–5763.
13. Beltagy, I.; Peters, M.E.; Cohan, A. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150* **2020**.
14. Roudsari, A.H.; Afshar, J.; Lee, C.C.; Lee, W. Multi-label Patent Classification using Attention-Aware Deep Learning Model. In Proceedings of the IEEE International Conference on Big Data and Smart Computing (BigComp), 2020, pp. 558–559.
15. Li, S.; Hu, J.; Cui, Y.; Hu, J. DeepPatent: patent classification with convolutional neural networks and word embedding. *Scientometrics* **2018**, *117*, 721–744.
16. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* **2013**.
17. Kim, Y. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* **2014**.
18. Risch, J.; Krestel, R. Domain-specific word embeddings for patent classification. *Data Technologies and Applications* **2019**.
19. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing* **1997**, *45*, 2673–2681.
20. Lee, J.S.; Hsiang, J. PatentBERT: Patent Classification with Fine-Tuning a pre-trained BERT Model. *arXiv preprint arXiv:1906.02124* **2019**.
21. Hu, J.; Li, S.; Hu, J.; Yang, G. A Hierarchical Feature Extraction Model for Multi-Label Mechanical Patent Classification. *Sustainability* **2018**, *10*, 219.
22. Bai, J.; Shim, I.; Park, S. MEXN: Multi-Stage Extraction Network for Patent Document Classification. *Applied Sciences* **2020**, *10*, 6229.
23. Song, G.; Huang, X.; Cao, G.; Liu, W.; Zhang, J.; Yang, L. Enhanced deep feature representation for patent image classification. In Proceedings of the Tenth International Conference on Graphics and Image Processing (ICGIP 2018). International Society for Optics and Photonics, 2019, Vol. 11069, p. 110690P.
24. Jiang, S.; Luo, J.; Pava, G.R.; Hu, J.; Magee, C.L. A CNN-based Patent Image Retrieval Method for Design Ideation. *arXiv preprint arXiv:2003.08741* **2020**.
25. Csurka, G. Document image classification, with a specific view on applications of patent images. In *Current Challenges in Patent Information Retrieval*; Springer, 2017; pp. 325–350.

 doi:10.20944/preprints202502.1615.v1

26. Landauer, T.K.; Foltz, P.W.; Laham, D. An introduction to latent semantic analysis. *Discourse processes* **1998**, *25*, 259–284.
27. Kim, B.T.S.; Hyun, E.J. Mapping the Landscape of Blockchain Technology Knowledge: A Patent Co-Citation and Semantic Similarity Approach. *Systems* **2023**, *11*. https://doi.org/10.3390/systems11030111.
28. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International conference on machine learning. PMLR, 2020, pp. 1597–1607.
29. Fang, H.; Xie, P. Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766* **2020**.
30. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3973–3983.
31. Gunel, B.; Du, J.; Conneau, A.; Stoyanov, V. Supervised Contrastive Learning for Pre-trained Language Model Fine-tuning. In Proceedings of the International Conference on Learning Representations, 2021.
32. Li, Z.; Zhou, L.; Yang, X.; Jia, H.; Li, W.; Zhang, J. User Sentiment Analysis of COVID-19 via Adversarial Training Based on the BERT-FGM-BiGRU Model. *Systems* **2023**, *11*. https://doi.org/10.3390/systems11030129.
33. Lin, W.; Yu, W.; Xiao, R. Measuring Patent Similarity Based on Text Mining and Image Recognition. *Systems* **2023**, *11*. https://doi.org/10.3390/systems11060294.
34. Young, T.; Hazarika, D.; Poria, S.; Cambria, E. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine* **2018**, *13*, 55–75.
35. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 1701–1708.
36. Caruana, R. Multitask learning. *Machine learning* **1997**, *28*, 41–75.
37. Ruder, S. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* **2017**.
38. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.