

Research Article

PQPS: Prior-Art Query-Based Patent Summarizer Using RBM and Bi-LSTM

Girthana Kumaravel  and **Swamynathan Sankaranarayanan** 

Department of Information Science and Technology, CEG Campus, Anna University, Guindy, Chennai 600025, India

Correspondence should be addressed to Girthana Kumaravel; keerthi3110@gmail.com

Received 24 August 2021; Accepted 18 November 2021; Published 28 December 2021

Academic Editor: L. J. García Villalba

Copyright © 2021 Girthana Kumaravel and Swamynathan Sankaranarayanan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A prior-art search on patents ascertains the patentability constraints of the invention through an organized review of prior-art document sources. This search technique poses challenges because of the inherent vocabulary mismatch problem. Manual processing of every retrieved relevant patent in its entirety is a tedious and time-consuming job that demands automated patent summarization for ease of access. This paper employs deep learning models for summarization as they take advantage of the massive dataset present in the patents to improve the summary coherence. This work presents a novel approach of patent summarization named PQPS: prior-art query-based patent summarizer using restricted Boltzmann machine (RBM) and bi-directional long short-term memory (Bi-LSTM) models. The PQPS also addresses the vocabulary mismatch problem through query expansion with knowledge bases such as domain ontology and WordNet. It further enhances the retrieval rate through topic modeling and bibliographic coupling of citations. The experiments analyze various interlinked smart device patent sample sets. The proposed PQPS demonstrates that retrievability increases both in extractive and abstractive summaries.

1. Introduction

The importance of innovative technology development has been well established in many industrial sectors. Also, enterprises assess their invention in terms of intellectual property rights (IPRs) primarily through their patents. With the rapid advancement of various technologies worldwide, patent search and analysis have become an essential task for both the government and the private sector [1]. Enterprises use this legal and technical document (patent) to gain state-of-the-art technology, reveal business trends, and inspire novel solutions [2–5]. These patent rights last for around 20 years and give the inventor the right to use the invention commercially. The patentable subject matter and patentability restrictions differ by region. The enterprise's patent attorneys, inventors, and researchers devote a significant amount of time and resources to finding the right patents to discover new technological developments and focus their research in that direction [6]. They also perform this prior-art search to prevent the infringement of the current

innovation with the established technology and intellectual property. It is usually performed to ensure the invention's originality. It is publicly available evidence that the invention already exists. This search technique is much effective in assessing the invention's novelty and non-obviousness, identifying the potential related and competing art, and finally determining the patented invention's strength and scope.

The majority of traditional prior-art search techniques are keyword-based. The patent examiners or the patent analyst usually frame the patent search query from the patent application document by considering the term frequency. The priority date and classification codes are typically included in this frequency-based keyword search technique. Because of the patent's ambiguous and non-standard language, the documents found by a keyword-based prior-art search are insufficient to invalidate the claims. The formulated query is expanded with terms or phrases from external resources such as International Patent Classification (IPC) code definitions [7], thesaurus [8], or

knowledge base [9] to boost the retrieval rate and to cope up with the term mismatch problem.

Patent citations, in addition to patent textual fields and classification code, have been shown to improve retrieval rates [10]. They represent the relationship among the patents. Citation links assist in the discovery of more critical and valuable documents by granting authority to a cited or citing text. Approaches based on citations [11] include bibliographic coupling (BC), co-citation, and direct citation. In co-citation, two documents are relevant if they are cited together by one or more documents, while in BC, the document pair is relevant if both cite together one or more related documents. The stronger the bibliographic pairing is, the more citations the bibliographically coupled text pair shares. BC is retrospective while co-citation is forward looking. This paper makes use of BC to enhance the retrieved prior-art patent search set consisting of thousands of documents. The result set has many irrelevant documents. To search through the entire set and to find the relevant ones is tedious and time consuming. So instead, ranking them based on relevance by incorporating the patent characteristics will do a better job and improve precision.

Furthermore, since the lexicon in these documents varies more, manual processing (reading and understanding) and identifying prominent information from each patent document in the retrieved prior-art search set will be more difficult. The development of text summarizers for technical and legal documents has been prioritised to address this problem. Summarization aims to create succinct and insightful summaries of retrieved patent document collections while preserving the document's sense. Automatically producing summaries from broad text corpora has long piqued the interest of researchers in information retrieval and natural language processing. These summaries produce a gist (condensed version) of the text that emphasizes only the most relevant points [12]. Automatic summarization is classified as extractive or abstractive depending on how the summary is produced. The most important sentences or paragraphs are selected and assembled to form a description in extractive summarization. On the contrary, abstractive summarization generates meaningful sentences.

The proposed PQPS focuses on generating effective summaries on prior-art search results. The prior-art search patent documents are based on the search query and obtained by expanding the initial query with information from the knowledge base. The prior-art patents obtained using this method lack some relevant documents and may include irrelevant documents. Topic modeling approaches and citation analysis are used to enhance further the prior-art result set. Extractive and abstractive summaries are generated from the resultant set. The PQPS encompasses both extractive and abstractive techniques as the patents are lengthy and are challenging to obtain the gist by retaining the information in its entirety.

The main contributions of this paper are as follows:

- (1) Filtering the base query processor patent resultant set as it contains more irrelevant documents through Latent Dirichlet Allocation (LDA).

- (2) Enhancing the filtered patent set through bibliographic coupling.
- (3) Ranking the retrieved prior-art search patent set based on the structural similarity.
- (4) Generating extractive summaries with stacked RBM.
- (5) Employing the Seq2Seq model with pretrained embeddings and attention for generating abstractive summaries.

The rest of the paper is laid out as follows. Section 2 portrays the existing works carried out on search query formulation, patent citation analysis, and patent summarization. Section 3 outlines the background of the models and techniques employed for text summarization. The detailed flow of the proposed system is portrayed in Section 4. Sections 5–7 discuss the methodology of the proposed system in detail. The experimental results carried out as part of this work are detailed in Section 8. Finally, Section 9 concludes this paper and discusses the future work.

2. Related Work

This section presents the challenges associated with prior-art search and presents it in three dimensions. First, we focus on query formulation and expansion techniques for prior-art search. Secondly, we consider the methods that improve the retrieval rate through citations, and finally, we present the techniques for summarizing the patent documents.

2.1. Prior-Art Search Query Processing. Prior-art search query formulation and expansion are the foci of research on enhancing prior-art search and retrieval. As a result, most of the previous search queries relied on patent terms from various textual areas [13–18]. Because of abstract or generic terms offered by patentees to optimize their protection scope, this keyword-based query formulation technique falls behind, and the vocabulary mismatch problem persists. This method frequently necessitates additional research on the patent application domain. To address this issue, the authors use external resources such as thesaurus and domain-independent knowledge bases (WordNet, Wikipedia, and Wiktionary) [9, 19, 20] and domain-dependent knowledge bases (IPC and domain ontology) [7, 9, 21] to expand queries. Expansion with the domain-independent knowledge bases improves precision but recall drops due to lack of contextual information. IPC definitions were also utilized to expand queries [7]. Although it enhances recall in the chemical area, the results were not consistent across topics. This system creates a domain ontology and expands the query with terms and phrases from the domain ontology to address the vocabulary mismatch problem.

2.2. Patent Retrieval through Citations. Patent citations are essential for establishing relationships between patents and demonstrating technical developments and evolutions [22]. In this work, as a source for key extraction, the authors employed in-text citations from both patent and non-patent

literature and additional metadata. Mahdabi and Crestani used a similar approach, expanding the prior-art search query with term distribution of publications from the citation network [23]. Fuji et al., on the other hand, used citation connections to re-rank patent publications [24]. The authors used textual data and citation linkages to score and rank the patents. This proposed PQPS varies from prior systems where it uses a bibliographic coupling network of patent citations to find missing relevant patents.

2.3. Patent Summarization. Advancements in machine learning and artificial intelligence have simplified many tasks. One of the major tasks made easy for human through these techniques was automatic text summarization. Several approaches for text summarization have been developed to date. These summarization systems need to produce a concise summary while representing the information presented in the source document. Based on the way the summaries are generated, summarization techniques fall in to two categories: extractive and abstractive. Extractive summarization [25] techniques select sentences from source document while abstractive technique [26] generates the summary like human-crafted one by considering the whole document.

The most common techniques used for extractive summary generation are statistics-based, topic-based, discourse-based, and graph-based methods. Statistical techniques use statistical features [27, 28] such as sentence location [29, 30], sentence centrality, word or proper noun frequency [31], title similarity, and sentence bushy direction. Individual sentence scores are computed based on assigned feature weights, and sentences with high scores are more likely to be included in the generated summary. On the other hand, topic-based methods identify the terms that characterize the document's topic and use signatures or templates to score the sentences. Sentences are represented as nodes in graph-based approaches [32, 33], and a linkage is formed if there is a relationship between them. Many machine learning techniques have been used for summarization, including latent semantic analysis (LSA), Bayesian models [34], topic models, and hidden Markov models (HMMs) [35]. External knowledge bases, such as Wikipedia [36] and ontologies [37, 38], were also used for text summarization to identify meaningful sentences by mapping them to concepts in the ontology. Recently, text summarization has grown fast with advances in profound learning technology like RBM [39, 40], recurrent neural network (RNN) [41], and convolutional neural network (CNN) [39, 42]. Some researchers viewed text summarization as a sequence labeling task [43] and generated the summary. The SummaRuNNer [41] proposed by Nallapatti et al. is a sequence labeling task where the author evaluated the probability of a sentence to be included in the summary and then included them until it reaches summary length.

The abstractive summarization task has recently received ample attention because of its ability to generate sound and verbally robust summaries as that of humans [44]. This task was mostly performed with many-to-many Seq2Seq model,

and it was first introduced by Cho et al. [45] and Sutskever et al. [46]. Rush et al. [47] proposed an abstractive sentence summarization model encompassing local attention-based encoder and neural network language model decoder. Chopra et al. [48] proposed a conditioned RNN model for a decoder with a convolutional attention-based encoder in line with sentence summarization. This model outperforms other state-of-the-art models on the Gigaword corpus dataset. We can see that these summarization models mainly focused on news articles or CNN mail datasets.

Even though text summarization has garnered much attention in recent years, the summaries generated for patent documents are far from human-derived summaries and only a few research studies [49–53] address the problem of patent text summarization. These works either rely on some metrics to retrieve sentences or paragraphs to be included in summary using ontology [49] or focus on the patent document's claim section [51]. They have used metrics of discourse summarization. These methods are insufficient because the patents contain many recurring abstract terms such as “apparatus,” “methods,” “means,” and “device.” Additionally, focusing just on the claim section results in the inclusion of embodiment of the invention in the generated summary. The proposed PQPS system is novel because it combines both extractive and abstractive techniques for generating patent summary through deep learning techniques mainly RBM and Bi-LSTM, respectively.

3. Encoder-Decoder Architecture

This section provides an overview of the many deep learning-based models used for abstractive summarization techniques such as RNN, LSTM, and GRU. The encoder-decoder architecture is based on the sequence-to-sequence model [46]. Text summarization is a many-to-many sequence problem, where the input sequence (paragraph or document) is mapped to another similar sequence (summary) of varying length. The encoder and decoder are the two primary components of this approach. They are stacks of recurrent neural network units. The encoder reads the entire input sequence and generates context vector as an internal representation. At each timestep, the decoder reads the context vector and generates the output summary. In the following section, we will look at how different deep learning models can be integrated with this framework to generate abstractive summaries.

3.1. Recurrent Neural Network. The input text is processed in a sequential order by RNN through feedback loops. These loops distribute data among the various nodes and make predictions based on the gathered information. Thus, the RNN preserves the order of input words in the sequence. Whenever a new input is received, prediction is made by considering the output of the preceding states. During training, RNN computes gradients at each timestep using backpropagation through time (BPTT) algorithm. This network performs well with shorter sequences. With a lengthy input sequence, it suffers from vanishing gradient

problem [54, 55] during backpropagation as the gradient becomes smaller and smaller so that update becomes insignificant. Another major issue with larger sequence is training and evaluation due to the computation and memory constraints [56].

3.2. Gated RNN. Long short-term memory (LSTM) and gated recurrent unit (GRU) handle the problem of vanishing gradients using their gates. They have control over the information passing between the hidden states. These two networks are essentially the RNN variants with independent hidden and cell states. Figure 1 depicts the variances of the two networks, RNN and LSTM. The LSTM has three gates as shown in the diagram: forget, input, and output gates. The forget gate (equation (1)) is a single-layered architecture with sigmoid activation. This activation function in the forget gate assists in determining whether to preserve the information or discard it.

$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_f). \quad (1)$$

With the information available, the input gate attempts to learn new information (equations (2) and (3)) and quantifies the significance of the information carried (equation (4)). Based on the significance, the information is stored in the cell state.

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i), \quad (2)$$

$$\tilde{C}_t = \tanh(w_c[h_{t-1}, x_t] + b_c), \quad (3)$$

$$i_t = f_t * C_{t-1} + i_t * \tilde{C}_t. \quad (4)$$

The information is passed from current timestamp to the next through the output gate, and the same is given by equation (5). As stated in these equations, the value of the hidden state is determined by passing through the sigmoid and tanh functions. This hidden state (h_t) (equation (6)) is used for prediction. GRU is quite similar to LSTM; however, it lacks memory unit. Also, it is less complex with only two gates, namely, reset and update gates.

$$\tilde{o}_t = \sigma(w_o[h_{t-1}, x_t] + b_o), \quad (5)$$

$$h_t = o_t * \tanh(C_t). \quad (6)$$

3.3. Bidirectional RNN. A unidirectional RNN during prediction considers only the previous sequences, and there are possibilities of having noise. As a result, the future predictions suffer, lowering the quality of the summary. To address this issue, bidirectional RNN processes the input sequence in both forward and backward directions, i.e., the input sequence is fed in normal time order for one network and in reversed order for another network. At each time step, the output of the

two networks is concatenated and transmitted to the next level. Thus, the network will carry information about both preceding and next sequences to construct a summary. Bidirectional RNN enhances the quality of the summary generated.

3.4. Network with Attention. The performance of the Seq2Seq models can be improved with better network structure. A single context vector is passed as input from encoder to decoder in the encoder-decoder network. However, if the input sequence is lengthy, this alone will not capture the complete essence. As a result, various context vectors are derived in order to focus on certain parts of the input sequence [57]. Local attention and global attention were distinguished by Luong et al. [58]. Local attention considers only a few hidden states of the encoder when determining the attended context vector, whereas global attention considers all hidden states.

3.5. Beam Search. Beam search techniques are frequently employed in conjunction with the decoder in tasks such as multiple language generation, text summarization, and machine translation [59, 60]. Decoding the sequences entails searching over all the potential sequences and ranking them based on their likelihood. Because the vocabulary in these tasks often consists of dozens or millions of words, this search becomes intractable (NP-complete). As the size of the input rises, the heuristic approaches offer one or more approximate output sequences, which may or may not be sufficient. These algorithms decode sequences using probability and greedy or beam search. In greedy search, the best candidate for an input sequence is chosen at each time step based on likelihood. However, producing only one top candidate may result in a suboptimal solution. In contrast, beam search analyzes many candidates for an input sequence at each timestep.

4. PQPS: Prior-Art Query-Based Patent Summarizer

The functioning of the proposed PQPS is described in Figure 2. The query processor retrieves the initial set of patents based on the query built with knowledge bases (domain ontology and WordNet) and the novel patent application document. Though this retrieved set retrieves multiple relevant documents, it can have irrelevant documents and miss some relevant documents due to information overload. The PQPS system filters the irrelevant documents using LDA and uses a bibliographic coupling network on citations to improve the retrieval efficiency. The resultant document set is then ranked using the structural similarity metrics. The PQPS then addresses the high workload of the patent analyst by summarizing the ranked patents in both extractive and abstractive manner using deep learning techniques. A detailed explanation of each module is given in the following sections.

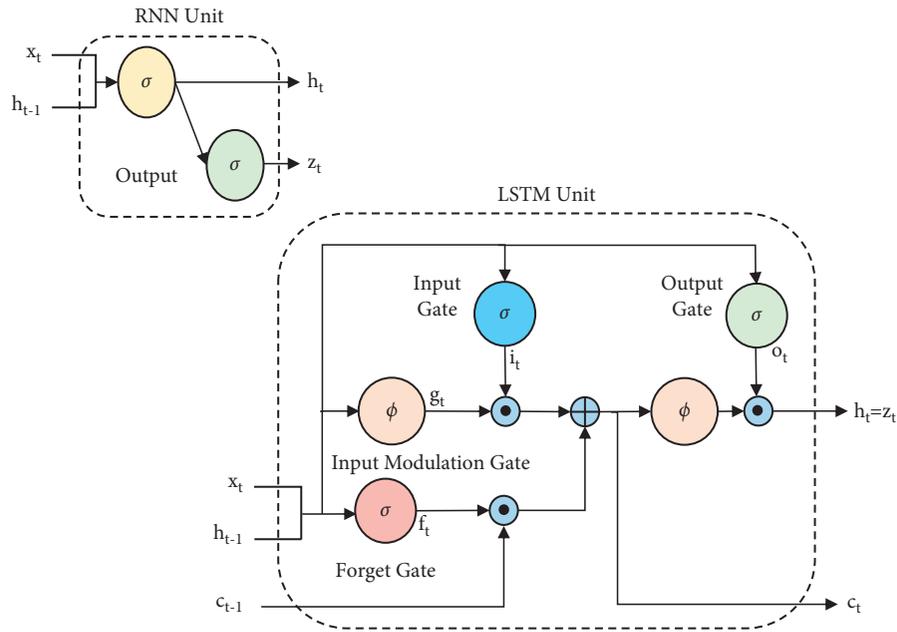


FIGURE 1: RNN vs. LSTM.

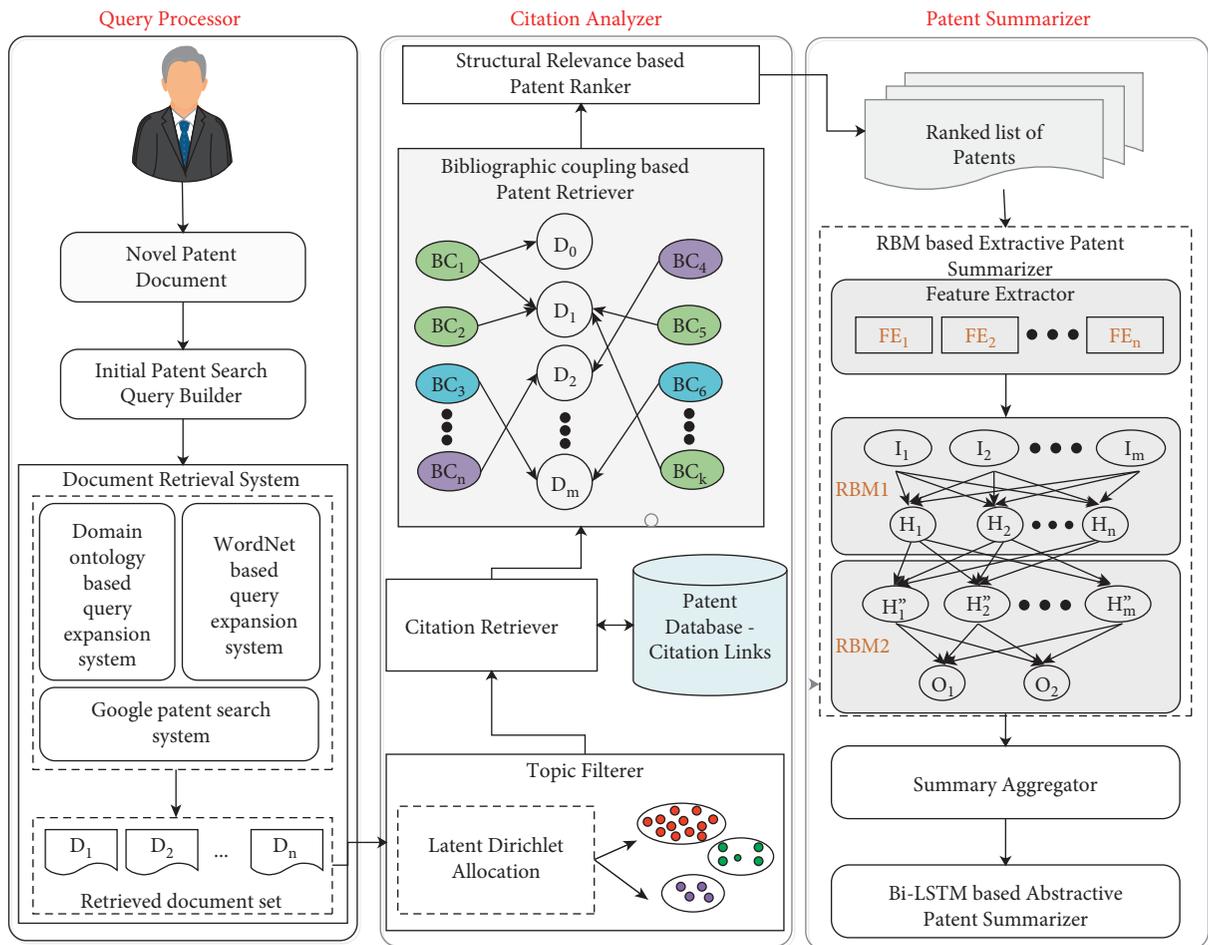


FIGURE 2: Prior-art query-based patent summarizer using RBM and Bi-LSTM.

5. Query Processor

The query processor builds an initial query from the patent application document issued by the patent analyst. The initial query is built by extracting noun phrases from different textual fields title, abstract, technical field, and description. The candidate noun phrases are selected to build an initial query based on the term frequency-inverse field frequency (TF-IFF) scoring. The patent document is lengthy and verbose, and each patent has its lexicon; therefore, vocabulary mismatch arises. To rectify this mismatch, the PQPS document retrieval system uses knowledge bases such as domain ontology and WordNet to enrich the initial query with semantically related concepts and terms. The domain ontology-based query expansion uses smart device domain ontology to expand the domain-related concepts while the WordNet-based query expansion system relies on WordNet, a lexical database for English. The document retrieval system's Google patent search employs Google prior-art search API to retrieve patents from the initial query. The citation analyzer module is used for further processing in all the documents obtained by these three systems. More details about this query processor were detailed in our previous work [9].

6. Citation Analyzer

Patent analysis conducted on the query processor module reveals that irrelevant documents were retrieved, in addition to relevant documents. Some relevant documents were uncovered from the retrieval due to the prevailing vocabulary mismatch problem. This citation analyzer module adopts a filtering mechanism through LDA and bibliographic coupling methods to reduce the irrelevant patent retrieval and further enhances the relevant document retrieval.

6.1. Topic Filterer. The topic filterer finds the abstract topics using LDA, an unsupervised model from topic modeling. The central intuition behind LDA for document filtering is that it groups each document based on its words, and further related documents are clustered to form a topic. It is based on the assumption that each document in the collection is a mixture of topics, and therefore, the document belongs to the topic whose strength is vital. This filterer analyzes title, abstract, and description of the relevant patent set. The fields are preprocessed, and LDA with collapsed Gibbs sampling [61] is employed. It is a Markov chain Monte Carlo approach where the model parameters are drawn from the posterior distribution for each iteration.

6.1.1. Identification of Number of Topics. The number of topics is usually determined based on the statistical measure perplexity [62]. It determines the predictive quality of the model. The low perplexity value indicates better performance. But according to Chang et al. [63], perplexity is not correlated to human judgments. Therefore, the PQPS incorporates a trial-and-error approach with different values

for number of topics based on the coherence value. The topics obtained along with their main keywords and manually generated category names are detailed in Table 1 for a sample patent application entitled "Bluetooth beacon attendance system based on smartphone and application method." For this sample from the patents retrieved, through a trial-and-error approach, the number of topics is decided as 45.

6.1.2. Relevance with Novel Patent Application. The filterer uses the topic probability distribution of each document to filter the irrelevant retrieved patent documents. LDavis [64], an interactive tool, is used to interpret and visualize their distributions, and the same is shown in Figure 3. Here the topics are represented as circles where their centres are determined by computing the distance between the topics. The size of the circle depicts the topic prevalence in the corpus. The intertopic distance is computed using the Jensen-Shannon divergence, a symmetric similarity measure. Based on the intertopic distance, the closely related clusters with the sample patent application topic clusters are chosen as relevant documents and the remaining clusters are filtered out.

Figure 3 represents the intertopic distance using the LDavis tool for the sample patent application titled "Bluetooth beacon attendance system based on smartphone and application method." This figure focuses on the 44th topic and its closeness with other related topics. As their closeness represents their similarity, closely linked topics are only taken into account for further processing. In this case, the documents which belong to the topics highlighted by a red box are only chosen as relevant document clusters.

6.2. Bibliographic Coupling-Based Patent Retriever. After filtering, the citations for each relevant patent are obtained through Open Patent Services (OPS), a European Patent Office (EPO) web service. This process allows access to EPO's raw data through the XML interface. This web service extracts all the citations links for each filtered patent set and is stored in a database. With the data available, we build a citation graph where each patent document acts as a vertex and between the vertices, and there is a directed edge if one patent document cites or is cited by another. Bibliographic coupling helps to retrieve relevant documents that have not been previously retrieved because of information overload. BC groups the patent documents in this citation graph referring to the same set of cited patent documents. The fundamental idea is that if a document d_1 is cited by another document d_2 , it means that d_1 is in some way related and essential to d_2 . This relatedness helps to identify missing relevant documents for the patent application document. The BC strength represents the number of common citations. For each patent document and application document pair, this BC strength is computed and patents with BC strength greater than a threshold are identified as missing relevant patents and included to the newly retrieved set.

TABLE 1: Topics with prominent keywords for a sample patent application.

Topics with their id	Prominent keywords	No. of documents in each topic
45: attendance	Terminal, smart, server, attendance, transmit	33
4: smartphone – based attendance	Bluetooth, phone, attendance, transmission, processing	8
37: digital information transformation	Mobile, telephone, cable, multiperformance, response	15
23: digital information transformation using wireless devices	Camera, capture, synchronize, infra red, selection	13
9: smartphone + wireless	Mobile terminal, notification, extract, receive, response	32
7: wireless – based access control for applications	Network, transfer, social, connection, client	11

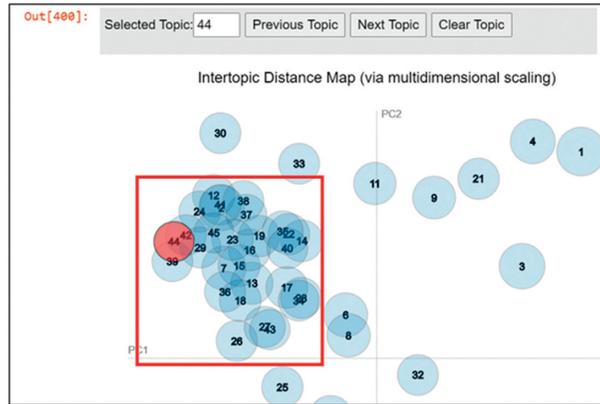


FIGURE 3: Intertopic distance graph.

Since a patent encompasses numerous subject areas, it may cite another document for any of these topics or subject areas. Therefore, the newly retrieved set has the possibility of having few irrelevant documents. All these references and topics need not be relevant to the patent application document. Thus, the newly retrieved patent set is filtered based on the cosine similarity with the patent application document and a threshold value.

6.3. Structural Relevance-Based Patent Ranker. The documents are ranked based on relevance with the search query terms. In a prior-art search, since the entire patent application is compressed as a query and because of the verbose nature of the patent documents, relevance metrics alone will not be sufficient to order the patents. The patent inherent feature structural similarity is incorporated in the relevance evaluation. Our analysis on the importance of different textual fields (title, abstract, background, and description) in our previous work [9] found that different fields have varying influences. The terms from the description field have more similarities than the abstract and title fields. This phenomenon occurs because the description field contains technical terminologies. Consequently, their similarity with the source document is given different weights. The relevance estimator assigns the field weights in the following order: $w_{\text{title}} < w_{\text{abs}} < w_{\text{desc}}$. Here w_{title} denotes the weight of terms from title field, w_{abs} denotes the weight of abstract field, and w_{desc} denotes the weights of words from the description section. The structural relevance score calculated with these textual fields is given by the following equation:

$$\begin{aligned}
 \text{SR}(Q_m, d_i) = & \frac{w_{\text{title}} * \text{sim}_{\text{title}}(Q_m, d_i)}{\sqrt{l(\text{title}_{d_i})}} + \frac{w_{\text{abs}} * \text{sim}_{\text{abs}}(Q_m, d_i)}{\sqrt{l(\text{abs}_{d_i})}} \\
 & + \frac{w_{\text{desc}} * \text{sim}_{\text{desc}}(Q_m, d_i)}{\sqrt{l(\text{desc}_{d_i})}}.
 \end{aligned}
 \tag{7}$$

Here, $\text{SR}(Q_m, d_i)$ is the structural relevance score, and $\text{sim}_{\text{title}}(Q_m, d_i)$ is the similarity between the semantically enriched query, Q_m , and the document title. Similarly, $\text{sim}_{\text{abs}}(Q_m, d_i)$ and $\text{sim}_{\text{desc}}(Q_m, d_i)$ represent the similarity between the semantically enriched query Q_m and the document abstract and description, respectively. $l(\text{title}_{d_i})$, $l(\text{abs}_{d_i})$, and $l(\text{desc}_{d_i})$ represent the length of the document title, abstract, and description, respectively.

7. Patent Summarizer

Patent summarizer creates summary through a unified model by combining both state-of-the-art extractive and abstractive approaches. It composes two neural network modules, i.e., summary extractor and abstractive summary generator. The summary extractor encodes each document, extracts the sentences from them, and clusters the individual summaries, and the abstractive summary generator paraphrases each summary clusters.

7.1. RBM-Based Extractive Patent Summarizer. The RBM-based extractive patent summarizer (RBM-EPS) inputs a document set D with multiple related patent documents

$d_1, d_2, d_3, \dots, d_i$. A single document d_i in the document set consists of multiple sentences $Si = si_1, si_2, si_3, \dots, si_n$. For each patent document in D , RBM-EPS creates a new document summary \bar{d}_i by selecting sentences from Si , and it aggregates them into three groups based on the degree of semantic and syntactic relation with the source document. RBM-EPS encompasses three submodules and will delve into each of them in detail.

7.2. Patent Feature Extractor. The first step towards extractive summarization is through identifying prominent features for the sentence selection. Patent feature extractor relies on the hand-crafted features that correspond to syntactic and semantic information of patent document sentences. Many of these features are widely used by summarizers for sentence selection [38, 40, 65–67], and their measures are normalized in the range of 0 to 1 for practical usage. The features that are extracted in this module are detailed in Tables 2 and 3.

7.3. Stacked RBM. This system makes use of a restricted Boltzmann machine, a non-deterministic generative model, to extract the salient sentences. RBM is a two-layer network with an input layer of visible nodes (m nodes) and an output layer of hidden nodes (n nodes). The two layers of a single unit of RBM form a fully bipartite graph, as seen in the workflow of PQPS (Figure 2). Here, the connections exist only between the nodes of two layers and not among the nodes within a layer where the i^{th} input node (I_i) is connected to h^{th} hidden node (H_h) by a weight (w_{ih}). Furthermore, all the nodes (visible and hidden) have a constant bias represented as a_i and b_h for visible and hidden layers accordingly. This system stacks the RBM to create a deep layer architecture. The first unit is a Gaussian–Bernoulli RBM [67], and the second is Bernoulli–Bernoulli RBM.

7.4. Summary Aggregator. The summarized documents are classified into three groups as strongly related, mediumly related, and weakly related based on Word Mover’s Distance (WMD) [71] score. WMD (equation (8)) measures the dissimilarity between the documents using word embedding and also takes into account the bag of words representation.

$$\text{Agg}(\bar{d}_i) = \begin{cases} \text{strong,} & 0 < \text{dis}_i < 0.4, \\ \text{medium,} & 0.4 < \text{dis}_i < 0.7, \\ \text{weak,} & \text{otherwise.} \end{cases} \quad (8)$$

where \bar{d}_i represents the extractive summary of i^{th} patent and dis_i indicates the WMD scoring between the search query or the source document and extractive summary. It uses pretrained word2vec embedding [27].

7.5. Bi-LSTM-Based Abstractive Patent Summarizer. Abstractive patent summary generation uses sequence-to-sequence (Seq2Seq) network [46], an encoder-decoder architecture. In this many-to-many sequence problem, the encoder parses the input sequence $x = (x_1, x_2, \dots, x_M)$ and creates a hidden sequence $h^e = (h_1^e, h_2^e, \dots, h_T^e)$ and forwards

to the decoder. The decoder makes use of this hidden representation as context information and generates the summary sequence $y = (y_1, y_2, \dots, y_S)$. Here M and S represent the number of encoder tokens (input document length) and number of decoder tokens (summary length), respectively. For encoding, the PQPS makes use of Bi-LSTM as it understands the context better by preserving the information in both directions backward (past) and forward (future). The structure of Bi-LSTM-based abstractive patent summarizer is represented in Figure 4. Here, three-layered bidirectional long short-term memory (stacked Bi-LSTM) forms the encoder and a single-layered LSTM is used as a decoder along with an embedding layer. In addition to this basic structure, it encompasses the attention mechanism for effective summarization, and we will explore each in detail.

7.5.1. Encoder. Though LSTM and gated recurrent unit (GRU) attempt to solve the vanishing gradient problem and have comparatively good performance, this work uses LSTM because of easy tuning and training time. As in the bidirectional setting, the encoder processes the input sequence in both forward \vec{h}^e and backward direction \overleftarrow{h}^e .

7.5.2. Beam Search Decoder Network. The decoder (single-layered LSTM) makes use of this encoder hidden state and previous output of the decoder and updates the decoder to a new hidden state (h_t^d) and selects a new token as this step’s decoder output. This decoder incorporates beam search for target word prediction instead of the usual greedy technique. At each time step t , the beam search retains top scoring B sequences based on the B sequences at the previous time step, where B is determined by beam width or beam size. Beam width determines the number of sequences to be kept in memory at each t . The target word (w) for time step t is predicted based on probability scores.

7.5.3. Embedding. ConceptNet NumberBatch [30] pretrained embeddings are opted for two reasons in the embedding layer. Firstly, it is built on other pretrained embeddings such as Glove [72] and word2vec [73], and secondly, it combines embeddings with knowledge bases such as WordNet and DBpedia.

7.5.4. Attention Mechanism. In a simple Seq2Seq model, the encoder usually returns a fixed length context vector that will not retain important information mainly if the input sequence is very long as in the case of the patent document. To deal with this, Bahdanau et al. [57] developed an alignment mechanism where at each time step, it focuses on the crucial parts of the text and generates a context vector (c_t). The context vector is achieved by computing attention distribution ($\alpha_{t,M}^e$) over the entire sequence of tokens given hidden encoder state (h_M^e) and decoder state (h_t^d) at time step t . The alignment scores ($s_{t,M}^e = s(h_M^e, h_t^d)$) are computed using additive attention. Additive attention linearly combines encoder and decoder hidden states and is given by

TABLE 2: Patent features for RBM-EPS.

Feature with description
<p>Title and search query similarity: patent title usually reflects the innovativeness or the main theme. This similarity feature helps to retain the sentences that are relevant and related to the patent title and search query provided by patent analyst. To compute them, both co-occurrences-based [40] and similarity-based features are incorporated.</p> <p>$FE_1 = T \cap \text{sen}_i + \text{SQ} \cap \text{sen}_i / \log T + \log \text{SQ}$, $FE_2 = \text{sim}(T, \text{sen}_i) + \text{sim}(\text{SQ}, \text{sen}_i) / \log T + \log \text{SQ}$</p> <p>$T$—set of words in title, SQ—set of words in search query, sen_i—set of words in i^{th} sentence, T—title length, SQ—search query length, and $\text{sim}(T, \text{sen}_i)$ and $\text{sim}(\text{SQ}, \text{sen}_i)$ are computed using cosine similarity.</p>
<p>Sentence field position: the first and last sentences in a paragraph or section provide meaningful and prominent information to the reader [68] and therefore it is likely to be part of the summary. $FE_3 = \begin{cases} 0.25, & \text{if } \text{sen}_i \text{ is at the beginning,} \\ 0.75, & \text{if } \text{sen}_i \text{ is at the end,} \\ \max[i^{-1}, (m-i+1)^{-1}], & \text{otherwise.} \end{cases}$ $FE_4 = 1 - (i/m)$</p>
<p>Term frequency-inverse field frequency (TF-IFF): this measure identifies sentences that have important noun phrases of the form $(\langle JJ \rangle * \langle NN \rangle * + \langle IN \rangle) ? \langle JJ \rangle * \langle NN \rangle * +$ at a particular field in the document. Here JJ denotes an adjective, NN represents noun phrases, and IN points to preposition or subordinating conjunction.</p> <p>$FE_5 = \text{TF} - \text{IFF}_{t,f,i} = \sum_{t \in \text{sen}_i} \text{TF}_{t,f} * \text{IFF}_{t,f}$, $\text{TF}_{t,f} = n_{t,f} / \sum_k n_{k,f}$, $\text{TF}_{t,f}$—term frequency of term t in the field f, $\text{IFF}_{t,f}$—inverse field frequency of the field f, $n_{t,f}$—No. of occurrences of term t in the field f, $\sum_k n_{k,f}$ is the size of the field f, F is the number of fields in the document, and $1 + j: t_i \in f_j$ represents the field frequency.</p>
<p>Term frequency-inverse concept frequency (TF-ICF): this is more like TF-IFF where it measures importance of noun phrases to the concepts in smart device ontology hierarchy rather than field wise. There are chances where the term in the corpus does not belong to any concepts. In that case, its ICF value is assigned to 1.</p> <p>$\text{ICF}_{t,c} = \log C / (1 + c: t_i \in C)$, $FE_6 = \text{TF} - \text{ICF}_{t,c,i} = \sum_{t \in \text{sen}_i} \text{TF}_{t,i} * \text{ICF}_{t,c}$ where C is the total number of concepts in the smart device ontology is the total number of concepts in the smart device ontology and $1 + c: t_i \in C$ presents the concept frequency with respect to term.</p>
<p>Sentence length: generally in patents, the sentences are long. To discard very short sentences and very long sentences from the summary, this feature is included. FE_7 retrieves sentences that are close to the mean length of sentences in the document.</p> <p>$FE_7 = t * \bar{t}$, $FE_8 = \ln(\bar{t} - t - \text{sen}_i / \sigma)$.</p> <p>$t(\text{sen}_i)$ refers to number of terms in sentence sen_i, \bar{t} denotes the mean of all sentences in a document, and σ is the standard deviation.</p>
<p>Sentence centrality: this measure retains sentences that are close to each other. This sentence centrality is computed by considering unigrams and bigrams and average score is treated as feature score.</p> <p>$SC_{ij} = \sum \text{sim}(\text{sen}_i)(\text{sen}_j) / \max(\sum \text{sim}(\text{sen}_i, \text{sen}_j))$, FE_9 – Cosine similarity measure, FE_{10} – Jaccard similarity measure</p>
<p>Thematic words (FE_{11}): thematic words are the related words with the topic of the document and their frequency will be higher. Sentences with these words indicate that they are informative. The top 20 most frequent phrases are chosen as thematic words and the sentence score is calculated accordingly.</p>

TABLE 3: Patent features for RBM-EPS.

Feature with description
<p>TextRank (FE_{12}): this graph-based model scores the sentences by applying PageRank. It [32] computes the score based on the number of overlapping words between sentences.</p>
<p>Cue phrases: these phrases such as “in particular,” “in summary,” “as a result,” “as a consequence,” and so on are usually followed by important information, and so they are good indicators for estimating the sentence salience [69].</p> <p>$FE_{13} = \text{SCP}_i = \text{cp}_i / \text{tcp}$. Here, cp_i denotes the number of cue phrases in sentence i and tcp indicates the total number of cue phrases in the document.</p>
<p>Sentence semantic relatedness score (SSR): this score computes the relatedness between the search query and the phrases in a sentence with respect to smart device ontology using Lin’s measure [70]. Higher value on SSR score denotes they are more related and if there is no semantic match, SSR is assigned to 0.</p> <p>$FE_{14} = \text{SSR}_i = \sum_p \text{sim}_{\text{lin}}(\text{SQ}, p_{j,i}) / \text{tp}_i \text{sim}_{\text{lin}}(c_1, c_2) = (2 * \text{IC}(\text{LCS}(c_1, c_2))) / (\text{IC}(c_1) + \text{IC}(c_2))$, $\text{IC}(c) = 1 - \log(\text{hyponyms}(c) + 1) / \log(\text{max_nodes})$</p> <p>$\text{sim}_{\text{lin}}(\text{SQ}, p_{j,i})$ represents the summation of Lin’s similarity between search query and phrase j in sentence i and tp_i denotes the total number of phrases in sentence i. c_1 and c_2 are two ontologically related concepts, $\text{LCS}(c_1, c_2)$ represents their most common ancestor, $\text{hyponyms}(c)$ denotes the number of subsumers of a concept c, and max_nodes denotes the maximum number of nodes (concepts) in the ontology.</p>

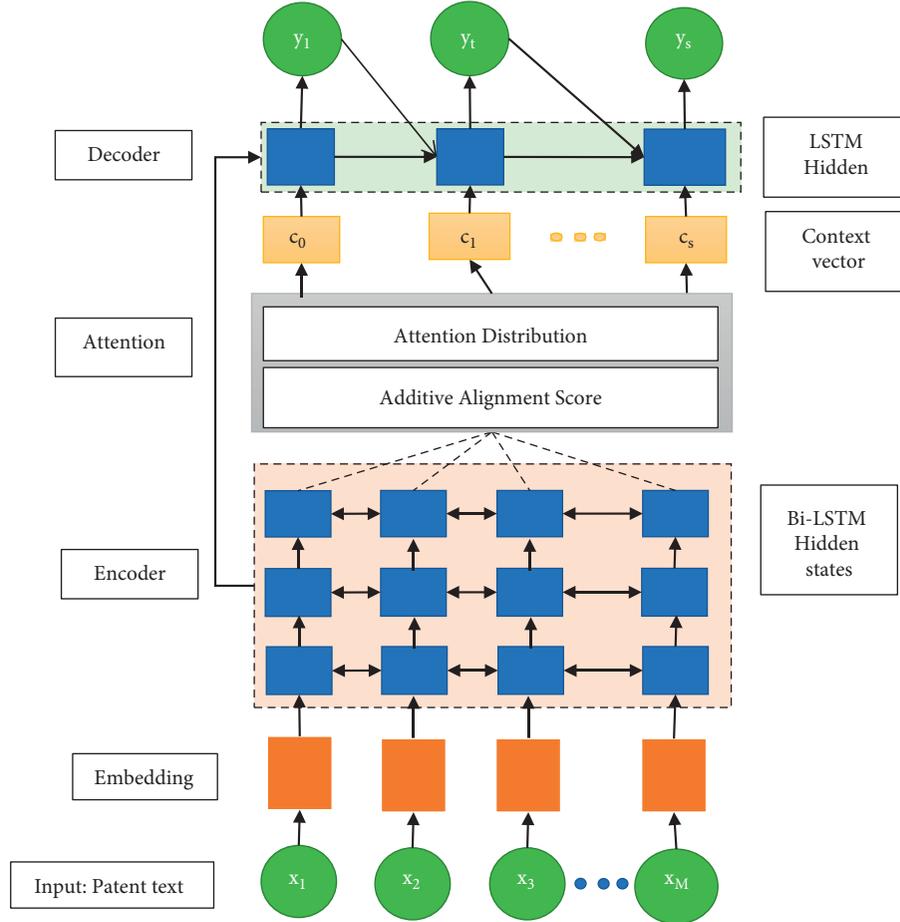


FIGURE 4: Bi-LSTM-based abstractive patent summarizer with attention.

$$s_{t,M}^e = v_{\text{align}}^T \tanh(w_{\text{align}}(h_M^e \oplus h_t^d)), \quad (9)$$

where both v_{align}^T and w_{align} are weight matrices.

8. Experimental Results

8.1. Query Processor. Experiments with query processor are carried out with the textual fields of smart device patents collected through the Google Patents search engine. The dataset for this experimentation includes 753 smartphone patents, 478 smartwatch patents, and 421 smarthome patents. The query processor analyzes the following aspects of PQPS:

- (i) Influence of patent textual fields in a prior-art search query: the system examines the patent textual fields (title, abstract, background, technical field, summary, description, and claims) and finds their impact on each of them on the prior-art search query. The result shows that the terms from the description field in the prior-art search query yield better results than other fields.
- (ii) Document retrieval system: it encompasses domain ontology-based query expansion system (DOQES), WordNet-based query expansion system (WQES),

and Google Patent Search System (GPSS) for retrieving patents.

The prior-art search query for DOQES and WQES is built through query expansion of initial query with smart device domain ontology and WordNet. GPSS automatically constructs a prior-art search query. The retrieval efficiency of the three subsystems in terms of mean average precision (MAP) and recall is portrayed in Table 4. The table results show that DOQES performs better than WQES which is better than GPSS. This difference in retrieval performance was due to the number and quality of search terms. A more detailed analysis of these two aspects of the query processor system was presented in [9].

8.2. Citation Analyzer. The PQPS citation analyzer focuses on three aspects of its submodules: topic-based filtering of the retrieved patent document set, missing relevant patent documents identification through BC, and patent ranking based on structural relevance. Each of these aspects is delved in the following subsections.

8.3. Topic-Based Filtering of Retrieved Patent Set. The topic filterer processes around 1000 patents each time as retrieved by the document retrieval system. Only the title and abstract

TABLE 4: Performance of document retrieval system across three different subdomains.

Search system/subdomain	Google Patent Search System		WordNet-based query expansion system		Domain ontology-based query expansion system	
	MAP	Recall	MAP	Recall	MAP	Recall
Smartphone	0.142	0.195	0.163	0.431	0.332	0.763
Smartwatch	0.151	0.234	0.173	0.421	0.253	0.734
Smarthome	0.138	0.252	0.169	0.416	0.271	0.741

TABLE 5: Overview of the document retrieval system results for sample patent applications.

Sample patent applications	Number of patents retrieved	Mean text length	Number of tokens
Bluetooth beacon attendance system based on smartphone and application method	1,642	140.79	2,31,180
Electronic watch having a compass function	1,387	128.36	1,78,038
Location-based messaging	1498	180.54	2,70,450

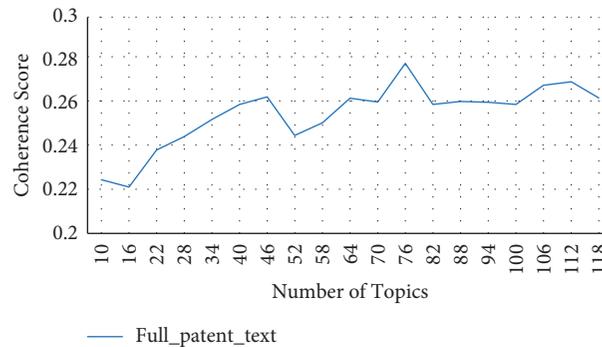


FIGURE 5: Deciding the number of topics for the sample patent application “Bluetooth beacon attendance system based on smartphone and application method” based on coherence score.

fields are considered for filtering. Even though all these patents were retrieved in response to a specific query, the patents retrieved covered a wide range of topics. The same can be observed from the token and vocabulary frequencies depicted in Table 5 for the patent set retrieved for various queries.

The number of topics for each retrieved patent set must be selected before generating the topic model to filter out irrelevant documents. The filterer computes the coherence score to determine the number of topics. It employs a trial-and-error approach to discover the best model by constructing multiple LDA models with topics ranging from 10 to 120. After comparing the coherence scores of multiple models, the model with the optimum coherence score is picked. The coherence scores of multiple LDA models for the sample prior-art search query after iterating 5 times and their mean scoring for different topics are shown in Figure 5. Here, the coherence score varies from 0.22 to 0.28, increasing as the number of topics increases. The optimal coherence score is chosen as the model has highest coherence score before a significant drop or flattening. The optimal coherence score is attained when Num – Topics = 45. Based on the intertopic distance between the topics, the topics are chosen. The topic clusters and their documents

that are close to the principal relevant topic are considered as relevant.

IPC codes were used to inspect the patents filtered out to see if any relevant patent documents were included. This filtered-out set does not have any relevant documents. Furthermore, the filtered-out documents significantly reduce the dataset size for further processing. Table 6 shows the retrieved patent set size statistics before and after filtering for sample patent applications. This table also enlists a sample set of IPC codes for manually investigated patents that had been filtered in and filtered out. For instance, the IPC codes of filtered in patents for patent application “Bluetooth beacon attendance system on smartphone and application method” are G07C1, H04L29, H04B5, H04M1, G06Q50, G06Q10, and so on. These IPC codes, in turn, are assigned to patents that specify “time or attendance registers’ registration or indication or recording,” “arrangements related to the transmission of digital information,” “near-field transmission system,” “telephonic communication-substation equipment,” “data processing systems for specific business sectors,” and “administration and management of data processing systems,” respectively. These topics are much relevant to the patent application. The IPC codes of filtered out retrieved patents, on the other hand, specify cardboard

TABLE 6: Document size before and after filtering for sample patent applications.

Sample patent application	No. of patents before filtering	No. of patents after filtering	Sample IPC codes of filtered retrieved patents	Sample IPC codes of filtered-out retrieved patents
Bluetooth beacon attendance system based on smartphone and application method	1642	1074	G07C1, H04L29, H04B5, H04M1, G06Q50, G06Q10	A63F9, A61B5, G01N33, G06F21
Electronic watch having a compass function	1387	741	G04B47, G04G21, A61H39, G01C1, A44C5, A63B71	A47G19, A47C3, A47C7, A41D19, B25F1, F21V33
Location-based messaging	1498	906	H04L12, H04L29, H04M1, H04W4	G08G1, G16H20, B60R16

or indoor games, measuring the diagnostic devices and exploring or analyzing the materials through specific methods, security arrangements for protecting computers, and so on. This result confirms that the patents related to these topics are irrelevant and can be filtered out.

8.4. Identifying Missing Relevant Patents through Bibliographic Coupling. The citing and cited patents are retrieved for the topic filtered patents. The date range considered for this citation data collection is from the priority date of the sample patent application to 2020/11/31. Because of abundant patent applications and granted patents, relevant patents can be overlooked in citations. As a result, the BC strength between patent pairs is examined to ensure that no relevant patents are excluded from processing. The BC strength, as previously stated, represents the correlation between the patents. For instance, there are 6,265 patents connected through the citation for the sample patent application “Bluetooth beacon attendance system based on smartphone and application method.” Among 6,265 patent citations, 3,494 bibliographic coupled patent pairs were identified. The pairs with low BC strength are excluded. The mean BC strength is computed and set as the threshold value. Around 263 patents with BC strength greater than threshold (3) were retrieved as relevant patents, and overall, both the query processor module and citation analyzer yield 1337 patents (1074 + 263). The same along with the statistics of other patent applications are tabulated in Table 7.

8.5. Ranking Patents Based on Structural Relevance-Based Patent Ranker. The ranker computes patent similarity based on inherent patent characteristic, precisely structural similarity. All the patents filed with USPTO and the WIPO have a defined structure that includes the necessary textual fields title, abstract, description, and background. During our previous work’s experiment of query processor module [9], we found that various textual fields have variable effects on the generation and retrieval of prior-art search queries. As a result, weights are applied to the field description, abstract, and title in decreasing order, with values of 0.75, 0.5, and 0.25, respectively.

8.6. Patent Summarizer

8.6.1. Dataset. Experiments for extractive summarization methods are carried out with a smart device patent

document set. This patent document set encompasses patents from smarthome, smartwatch, and smartphone domains and is retrieved as part of the query processor module and citation analyzer module in Figure 2. These patent documents are collected through Google search Application Programming Interface (API) using the expanded search query and citation analysis. Here, the detailed description of the patent is used as input, and the summary field acts as the reference summary. This document set consists of 500 documents for each search query.

Abstractive summarization models are trained using BIGPATENT [28] dataset. This dataset comprises 1.3 million patent documents grouped under nine categories based on Co-operative Patent Classification (CPC). Each patent embodiment is used as the input, and the abstract written by the applicant can be used as the gold standard summary. The average length of the gold standard summary is around 100 words. It is difficult to retain such long sequences in memory and generate a summary of this length. Therefore, this abstractive summarization uses only the first two from the abstract and use them as gold standard summary. Random patents under the technology categories “g” and “h” are chosen for training and validation.

All the patent documents considered were preprocessed to remove digits and special symbols, and the texts were converted to lowercase. Among the 1.3 million documents, the models were trained with 17,743 documents and validated with 7605 patents. On an average, the documents chosen for training and validation have an average of 100 sentences and 45 words per sentence. So, the patent documents with less than 50 sentences are not considered during training and validation. The statistics of this dataset are summarized in Table 8. The average extractive length for training and validation was 756 and 687, respectively. Similarly, the average human-crafted summary length was 40 words for both training and validation. These summarization models were tested with our summary extractor module results.

8.7. Evaluation Metric for Summarization. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [74], a recall-based measure, is used for verifying the performance of the extractive and abstractive summarization model. ROUGE is based on the overlapping n-grams (content overlap) between the predicted and gold standard summary. Among the variants of ROUGE, the most common measures such as

TABLE 7: Patent citation statistics.

Sample patent applications	No. of patents after filtering	Total no. of patent citations	BC patent pairs	Mean BC strength	No. of patents retrieved through BC	Total no. of patents retrieved
Bluetooth beacon attendance system based on smartphone and application method	1,074	6,265	3,494	3.4	263	1,337
Electronic watch having a compass function	741	3,896	2,125	2.5	129	879
Location-based messaging	906	4,987	3,892	1.5	114	1020

TABLE 8: Statistics of the dataset for patent summarization.

Statistics	Training	Validation
Total no. of documents	17,743	7,605
Average description length (words)	1,706	1,688
Average extractive summary length	756	687

TABLE 9: Extractive summary generated by the models.

Stacked RBM summary (our model)

- (1) The invention relates to a Bluetooth beacon attendance system based on a smartphone.
- (2) Components are writing tool, a Bluetooth beacon, a smartphone mobile attendance application system, a smartphone Bluetooth 4.0 chip, a smartphone Bluetooth 4.0 recognition system, a smartphone information transmitting system, a remote attendance recording and information storage system, and a remote attendance management server system.
- (3) According to the Bluetooth beacon attendance system, based on the Bluetooth beacon sensing technique, an attendance scanning device is completed by virtue of the co-operation of a smartphone with the Bluetooth 4.0 during employee attendance, the information which is confirmed to be qualified is transmitted to the remote attendance management server system to be stored and to generate an attendance result and attendance abnormality in real time, the attendance abnormality prompting and attendance record inquiry can be acquired by an employee from the smartphone mobile attendance application system in real time, so that the appealing statement can be made by the employee.

LSA summary (state-of-the-art method)

- (1) The work attendance relevant information that instrument writes encryption in blank Bluetooth beacon.
- (2) Employee can obtain turn out for work abnormality prompt and attendance record inquiry in real time in smart mobile phone movable attendance checking application system and carries out complaint explanation.
- (3) Employee can obtain turn out for work abnormality prompt and attendance record inquiry in real time in smart mobile phone movable attendance checking application system and carries out complaint explanation.

TextRank summary (state-of-the-art method)

- (1) And write instrument, Bluetooth beacon, smart mobile phone movable attendance checking application system, intelligent mobile phone Bluetooth.
- (2) And write instrument, Bluetooth beacon, smart mobile phone movable attendance checking application system, intelligent mobile phone Bluetooth.
- (3) The work attendance relevant information that instrument writes encryption in blank Bluetooth beacon, step user opens the smart mobile phone movable attendance checking application system on mobile phone, input customer id, log in after user name and password, select "the Bluetooth attendance checking function" in "movable attendance checking," by the Bluetooth beacon that mobile phone is pressed close to make, call the signal content that intelligent mobile phone Bluetooth.

Free Summarizer (state-of-the-art method)

- (1) Recognition system, smart mobile phone information transmitting system, long-range attendance record information storage system, and long-range attendance management server system, wherein Bluetooth of mobile phone.
- (2) Recognition system connects smart mobile phone information transmitting system via smart mobile phone movable attendance checking application system, attendance record is exported to long-range attendance record information storage system by smart mobile phone information transmitting system, preserve attendance record by attendance record information storage system and result exported to long-range attendance management server system, long-range attendance management server system generates checking-in result, and by the inquiry of this checking-in result input smart mobile phone movable attendance checking application system reminding user.
- (3) Step user opens the smart mobile phone movable attendance checking application system on mobile phone, input customer id, log in after user name and password, select "the Bluetooth attendance checking function" in "movable attendance checking," by the Bluetooth beacon that mobile phone is pressed close to make, call the signal content that intelligent mobile phone Bluetooth.

ROUGE-1 (unigram), ROUGE -2 (bigrams), and ROUGE-L (longest common subsequence) are used here for evaluation. For these metrics, it presents results in terms of precision, recall, and F-score.

8.8. Effect of Extractive Summarizer. Our summary extractor module comprises two layers of RBM with 14 perceptrons in the input layer. The hidden layer's size is twice that of the input as it helps in discovering the latent factors. The final hidden layer is a softmax layer with 2 neurons. The 2 neurons represent the classes where the sentences are to be included in the summary or not. The learning rate for this model is fixed to 0.1. The quality of summary generated by our summary extractor module is tested by applying it on the smart device patent document set and their results are compared with the state-of-the-art methods latent semantic analysis (LSA) and TextRank. The results are compared with a summarization tool Free Summarizer (<http://freesummarizer.com/>). Excerpts (first 3 sentences) of the summary generated by the extractive models stacked RBM, LSA, TextRank, and Free Summarizer for a test document are shown in Table 9.

As observed in Table 9, the extractive summary generated using stacked RBM is consistent and well organized. On other hand, LSA and TextRank have redundant entries. In LSA, sentences 2 and 3 are redundant.

Similarly, in TextRank, redundant behavior is observed in sentences 1 and 2. Though we eliminate redundant entry and read through them, they are not consistent and difficult to find out the ultimate objective or topic of the document. Free summarizer on the other hand produces non-redundant summaries but the sentences are systematically organized.

Table 10 reveals the ROUGE scores obtained by different extractive summarization models and tools. Among these models, LSA, TextRank, and Free Summarizer have a 50% compression rate, while stacked RBM had an average compression rate of around 60%. As seen from Table 10, our summary extractor module using stacked RBM achieves much better results in terms of ROUGE-2 scoring than other metrics (ROUGE-1 and ROUGE-LCS). Stacked RBM outperforms all other methods and tools for extractive summarization because of the following two reasons.

The foremost reason is the feature extractor. The features are extracted by considering semantics, sentence saliency, redundancy, and coherence with the source document and prior-art search query. Semantic importance is computed with smart device domain ontology and sentences with concepts related to domain ontology are given priority. Redundancy is eliminated through similarity computation among sentence pairs. Also, coherence with prior-art search is achieved with title and search query similarity computation feature. Secondly, RBM discovers more latent factors than other methods. As observed from Table 10, the precision score for all ROUGE metrics is in a higher range than recall. This is because the obtained summaries are shorter than the gold standard extractive summary. A possible solution to address this issue is to have summary generation from the input text with a limit on the number of sentences

to be generated rather than salient sentence extraction based on features.

Although based on the metrics, stacked RBM performs better and can extract relevant and prominent sentences from the patents, it is essential to evaluate the candidate summaries qualitatively through domain experts. The readability part of the generated summary is qualitatively analyzed by the patent analyst and domain experts from academia. Two patent analysts and five computer science domain experts assessed 50 candidate summaries independently regarding gold standard truth and input patent text. The assessors evaluated by concentrating on informativeness, readability, and validity aspects of generated summary. Informativeness evaluates whether the generated summary is relevant or not. It also checks whether the overall content of the input text is conveyed in the generated summary or not. Readability checks for uniformity or coherence or understandable nature of the summary. Finally, validity evaluates whether the generated summary can be used as such or not. These factors are evaluated by measuring the score in the range of 0 to 5 with 5 being more coherent, readable, and informative, and 1 being unreasonable and not effective replacement of summary. In Table 11, the average scoring of the three factors (informativeness, readability, and validity) for the extractive models is tabulated. The results tabulated in Table 11 show that stacked RBM attains good scoring in all three focus points compared to other methods.

The average execution time of each algorithm is presented in Figure 6. It can be noticed from the figure that the execution time of stacked RBM is on higher side than other models as it involves the number of internal parameter evaluation. On the other hand, with Free Summarizer, the average execution time is static as it is web-based, and it does not consume much time. Though it consumes time, the summary generated proves that it is worth the time consumed.

8.9. Effect of Abstractive Summarizer. All the experiments with LSTM and Bi-LSTM are carried out with 512 latent dimensions and 128 embedding sizes. The Seq2Seq model with single-layered LSTM learns embeddings from the training documents while the Seq2Seq model with stacked LSTM (2 layers) and attention and our Seq2Seq model with stacked Bi-LSTM (3 layers) make use of pretrained embedding ConceptNet NumberBatch and attention. To avoid overfitting and to further improve the performance of the model, dropout is employed. LSTM and Bi-LSTM layers in the encoder had a dropout of 0.3 while dropout of 0.2 is employed at the decoder. 0.3 or 0.2 dropout means that 30% or 20% of neurons can be dropped during training. Adam [75] with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e - 8$ with learning rate $\eta = 0.001$ was used in all abstractive summarization experiments for optimization. Adam was chosen as it combines the properties of other stochastic gradient optimization algorithms such as RMSProp and AdaGrad. The LSTM models were trained for 50 epochs while the Bi-LSTM model was trained for 100 epochs. For all the models, early stopping was set up when the loss for validation data does not

TABLE 10: Performance comparison of summary extractor and other models.

Metric	ROUGE-1			ROUGE-2			ROUGE-LCS		
	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-score
Models									
LSA	0.52	0.34	0.35	0.62	0.43	0.62	0.51	0.34	0.42
TextRank	0.51	0.36	0.39	0.67	0.46	0.67	0.54	0.38	0.46
Free Summarizer	0.39	0.31	0.29	0.71	0.63	0.85	0.51	0.43	0.43
Stacked RBM	0.64	0.46	0.55	0.89	0.68	0.95	0.59	0.46	0.53

TABLE 11: Qualitative evaluation scoring.

Factors/models	Informativeness	Readability	Validity
LSA	2.84	2.56	2.47
TextRank	2.99	2.78	2.36
Free Summarizer	2.40	2.89	2.76
Stacked RBM	3.42	3.18	2.96

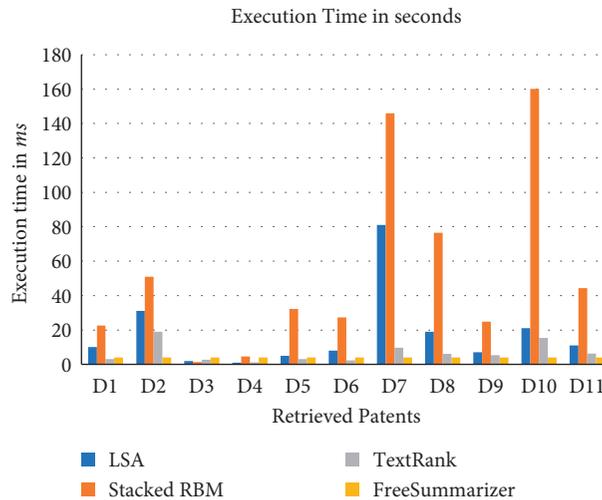


FIGURE 6: Execution time of extractive summary models.

TABLE 12: Performance comparison of abstractive summarization models.

Metric/models	ROUGE-1	ROUGE-2	ROUGE-LCS
	Recall	Recall	Recall
Stacked LSTM + ConceptNet + attention	0.342	0.216	0.31
LSTM	0.299	0.20	0.29
Stacked Bi-LSTM + ConceptNet + attention	0.399	0.252	0.35

improve after 5 epochs (patience=5). Also, to avoid exploding gradients problem, the gradient clipping technique with a threshold of 5 is applied. The beam width of 10 was used in the models which means it considers at most 10 words

at each time step while generating the target word. All these abstractive summarizations models were trained in Google Colab Notebook with Tesla T4 GPU environment. These models were run 10 times as they are stochastic by initializing

TABLE 13: Sample summary generated by abstractive summarization models.

Extractive summary (input): the invention relates to a Bluetooth beacon attendance system based on a smartphone. Components are writing tool, a Bluetooth beacon, a smartphone mobile attendance application system, a smartphone Bluetooth 4.0 chip, a smartphone Bluetooth 4.0 recognition system, a smartphone information transmitting system, a remote attendance recording and information storage system, and a remote attendance management server system. According to the Bluetooth beacon attendance system, based on the Bluetooth beacon sensing technique, an attendance scanning device is completed by virtue of the co-operation of a smartphone with the Bluetooth 4.0 during employee attendance, the information which is confirmed to be qualified is transmitted to the remote attendance management server system to be stored and to generate an attendance result and attendance abnormality in real time, the attendance abnormality prompting and attendance record inquiry can be acquired by an employee from the smartphone mobile attendance application system in real time, so that the appealing statement can be made by the employee. Above-mentioned technical matters of the present invention is mainly solved by following technical proposals: based on the Bluetooth Beacon attendance checking system of smart mobile phone, comprise Bluetooth of mobile phone 4.0 and write instrument, Bluetooth Beacon, smart mobile phone movable attendance checking application system, intelligent mobile phone Bluetooth 4.0 chip, intelligent mobile phone Bluetooth 4.0 recognition system, smart mobile phone information transmitting system, long-range attendance record information storage system and long-range attendance management server system, wherein Bluetooth of mobile phone 4.0 writes instrument by the data of turning out for work of encryption write Bluetooth Beacon, this information is inputted intelligent mobile phone Bluetooth 4.0 chip via Bluetooth signal by Bluetooth Beacon, the output terminal of intelligent mobile phone Bluetooth 4.0 chip connects intelligent mobile phone Bluetooth 4.0.

Reference summary: Bluetooth beacon attendance system based on a smartphone. The Bluetooth beacon attendance system comprises a mobile phone Bluetooth4.0 writing tool, a Bluetooth beacon, a smartphone mobile attendance application system, a smartphone Bluetooth 4.0 chip, a smartphone Bluetooth 4.0 recognition system, a smartphone information transmitting system, a remote attendance recording and information storage system, and a remote attendance management server system.

Stacked LSTM + attention: Bluetooth for smartphone attendance Bluetooth chip information system Bluetooth storage LSTM Bluetooth attendance for system Bluetooth Bluetooth attendance for for

Stacked Bi-LSTM + attention: it is a Bluetooth beacon system for attendance using phone. It has Bluetooth beacon and transmitter server and generates attendance.

the models with these parameters, and their average scores are presented in Table 12.

As observed from Table 12, the Seq2Seq model using stacked Bi-LSTM with attention and ConceptNet pretrained embedding achieves better performance than other models. This can be seen from the ROUGE scores where stacked Bi-LSTM improves stacked LSTM by 5.7% on ROUGE-1, 3.6% on ROUGE-2, and 4% on ROUGE-LCS. The sample summary generated by these models is represented in Table 13. In the table, the text is discussing about Bluetooth-based attendance management systems using a smartphone. It discusses the components involved and the working model of the system. The findings in Table 13 show that the Seq2Seq model using LSTM lacks main keywords and has repetitive common keywords while the Seq2Seq model with stacked LSTM has a better summary than that of LSTM. Also, the summary of stacked LSTM has more vocabulary representation than LSTM. This improvement is due to ConceptNet embedding. On other hand, the Seq2Seq model with LSTM learns embeddings from the training data available which are lesser when compared to the pretrained embedding dataset size. Comparing these two models, the summary generated by the Seq2Seq model using Bi-LSTM, attention, and ConceptNet embedding was much logical. Though the summary does not represent all the keywords present in the reference summary such as “recognition system,” “application system,” “chip,” and so on, it is understandable and has main concepts related to the text.

9. Conclusion

In this paper, we presented PQPS, an extractive and abstractive summarizer for patents. This summarizer is search

query based where it extracts prominent terms from the patent application document and expands them with a domain-dependent and domain-independent knowledge base. PQPS filters irrelevant documents using LDA-based topic modeling and enhances the relevant patent retrieval through bibliographic coupling to further improve the retrieval efficiency. The PQPS proposes a ranking model that ranks the resultant retrieval set by providing weightage to different fields of the patent. Finally, it uses deep learning models stacked RBM and Bi-LSTM to summarize the ranked set of patents extractively and abstractively.

Evaluation results of PQPS modules support the effectiveness of the proposed approach. Around 1600 patent applications from the smartphone, smartwatch, and smart-home domains have been tested with the PQPS system. The PQPS query processor module uses domain-dependent and domain-independent ontologies, thereby retrieving roughly 1000 prior-art patents for prior-art search query generation and expansion. The retrieval efficiency of the query processor system’s submodules was evaluated, and it was discovered that queries expanded with domain ontology improve relevant document retrieval in terms of recall by around 28% and 56%, respectively, over WordNet-based query expansion system and Google prior-art search system. LDA-based patent document filtering excludes extraneous documents using coherence score and intertopic distance map. The results are manually reviewed using IPC, and the patents that may be missed due to information overload are retrieved using BC. The resultant patent set is extractively summarized with stacked RBM. The average ROUGE-1, ROUGE-2, and ROUGE-LCS recall scores for stacked RBM were 0.46, 0.68, and 0.46, respectively, which were better than those of other state-of-the-art models like LSA,

TextRank, and the Free Summarizer tool. Abstractive patent summary generation using seq-seq Bi-LSTM with Num-Batch embedding and attention surpasses other models with an average recall of 0.399, 0.252, and 0.35 for ROUGE-1, ROUGE-2, and ROUGE-LCS, respectively. As part of future work, we intend to update the summarization model with more sentences.

Data Availability

The patent data used to support the findings of this study were collected through Google patent search API and Open Patent Services. They can be crawled and retrieved.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was sponsored by the Visvesvaraya PhD Scheme for Electronics & IT Proceedings (grant no. 3408/PD6/DeitY/2015).

References

- [1] L. Morrison and P. Germeraad, *Intellectual Asset Management at Avery Dennison, Profiting from Intellectual Capital: Extracting Value from Innovation*, Wiley, New York, NY, USA, 1998.
- [2] R. Paci, "More similar and less equal: economic growth in the European regions," *Review of World Economics*, vol. 133, no. 4, pp. 609–634, 1997.
- [3] J.-H. Kim, J.-X. Huang, H.-Y. Jung, and K.-S. Choi, "Patent document retrieval and classification at KAIST," in *Proceedings of the NTCIR Workshop Meeting, Serial. 5. NTCIR*, Tokyo, Japan, December 2005.
- [4] M.-J. Shih, D.-R. Liu, and M.-L. Hsu, "Discovering competitive intelligence by mining changes in patent trends," *Expert Systems with Applications*, vol. 37, no. 4, pp. 2882–2890, 2010.
- [5] J. C. Amy, C. V. Trappey, T.-A. Chiang, and Y.-H. Huang, "Ontology-based neural network for patent knowledge management in design collaboration," *International Journal of Production Research*, vol. 51, no. 7, pp. 1992–2005, 2013.
- [6] A. J. C. Trappey, F.-C. Hsu, C. V. Trappey, and C.-I. Lin, "Development of a patent document classification and search platform using a back-propagation network," *Expert Systems with Applications*, vol. 31, no. 4, pp. 755–765, 2006.
- [7] P. Mahdabi and F. Crestani, "Patent query formulation by synthesizing multiple sources of relevance evidence," *ACM Transactions on Information Systems*, vol. 32, no. 4, pp. 1–30, 2014.
- [8] H. K. Azad and A. Deepak, "A new approach for query expansion using Wikipedia and WordNet," *Information Sciences*, vol. 492, pp. 147–163, 2019.
- [9] K. Girithana and S. Swamynathan, "Domain ontology-based query expansion system (DOQES) for prior-art search," Unpublished.
- [10] P. M. Lee, "Term and citation retrieval: a field study," *Information Processing & Management*, vol. 29, no. 1, pp. 95–112, 1993.
- [11] Y.-W. Chang, M.-H. Huang, and C.-W. Lin, "Evolution of research subjects in library and information science based on keyword, bibliographical coupling, and co-citation analyses," *Scientometrics*, vol. 105, no. 3, pp. 2071–2087, 2015.
- [12] H. Khanpour, *Sentence extraction for summarization and notetaking*, PhD Thesis, University of Malaya, Kuala Lumpur, Malaysia, 2009.
- [13] W. Magdy and G. J. F. Jones, "Applying the kiss principle for the Clef-IP 2010 prior art candidate patent search task," 2010.
- [14] H. Mase, T. Matsubayashi, Y. Ogawa, M. Iwayama, and T. Oshio, "Proposal of two-stage patent retrieval method considering the claim structure," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 4, no. 2, pp. 190–206, 2005.
- [15] K. Konishi, "Query terms extraction from patent document for invalidity search," in *Proceedings of the NTCIR*, Citeseer, Tokyo, Japan, December 2005.
- [16] X. Xue and W. B. Croft, "Automatic query generation for patent search," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 2037–2040, Beijing, China, November 2009.
- [17] L. Helmers, F. Horn, F. Biegler, T. Oppermann, and K.-R. Müller, "Automating the search for a patent's prior art with a full text similarity search," *PLoS One*, vol. 14, no. 3, Article ID e0212103, 2019.
- [18] J. Risch, N. Alder, C. Hewel, and R. Krestel, "Patentmatch: a dataset for matching patent claims & prior art," 2020, <https://arxiv.org/abs/2012.13919>.
- [19] B. Al-Shboul and S.-H. Myaeng, "Wikipedia-based query phrase expansion in patent class search," *Information Retrieval*, vol. 17, no. 5–6, pp. 430–451, 2014.
- [20] P. Sharma, R. Tripathi, and R. C. Tripathi, "Finding similar patents through semantic query expansion," *Procedia Computer Science*, vol. 54, pp. 390–395, 2015.
- [21] F. Wang and L. Lin, "Domain lexicon-based query expansion for patent retrieval," in *Proceedings of the 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 1543–1547, IEEE, Changsha, China, August 2016.
- [22] P. Lopez and L. Romary, "Experiments with citation mining and key-term extraction for prior art search," in *Proceedings of the CLEF 2010-conference on Multilingual and Multimodal Information Access Evaluation*, Padua, Italy, September 2010.
- [23] P. Mahdabi and F. Crestani, "The effect of citation analysis on query expansion for patent retrieval," *Information Retrieval*, vol. 17, no. 5–6, pp. 412–429, 2014.
- [24] A. Fuji, M. Iwayama, and N. Kando, "Introduction to the special issue on patent processing," *Information Processing & Management*, vol. 43, no. 5, pp. 1149–1153, 2007.
- [25] A. Sinha, A. Yadav, and A. Gahlot, "Extractive text summarization using neural networks," 2018, <https://arxiv.org/abs/1802.10137>.
- [26] Y.-C. Chen and M. Bansal, "Fast abstractive summarization with reinforce-selected sentence rewriting," 2018, <https://arxiv.org/abs/1805.11080>.
- [27] M. Allahyari, S. Pouriyeh, M. Assefi et al., "Text summarization techniques: a brief survey," 2017, <https://arxiv.org/abs/1707.02268>.
- [28] A. Kanapala, S. Pal, and R. Pamula, "Text summarization from legal documents: a survey," *Artificial Intelligence Review*, vol. 51, no. 3, pp. 371–402, 2019.
- [29] H. P. Edmundson, "New methods in automatic abstraction," *ACM Journal*, vol. 10, pp. 321510–321519, 1969.
- [30] P. Ren, Z. Chen, Z. Ren, F. Wei, J. Ma, and M. de Rijke, "Leveraging contextual sentence relations for extractive summarization using a neural attention model," in

- Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 95–104, Tokyo, Japan, August 2017.
- [31] H. P. Luhn, “A statistical approach to mechanized encoding and searching of literary information,” *IBM Journal of Research and Development*, vol. 1, no. 4, pp. 309–317, 1957.
- [32] R. Mihalcea and T. Paul, “Textrank: bringing order into text,” in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pp. 404–411, Barcelona, Spain, July 2004.
- [33] X. Wan and J. Yang, “Improved affinity graph based multi-document summarization,” in *Proceedings of the Human Language Technology Conference of the NAACL*, pp. 181–184, New York, NY, USA, June 2006.
- [34] J. Kupiec, J. Pedersen, and F. Chen, “A trainable document summarizer,” in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 68–73, Seattle, WA, USA, July 1995.
- [35] J. M. Conroy and D. P. O’leary, “Text summarization via hidden Markov models,” in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 406–407, New Orleans, LO, USA, September 2001.
- [36] Y. Sankarasubramaniam, K. Ramanathan, and S. Ghosh, “Text summarization using wikipedia,” *Information Processing & Management*, vol. 50, no. 3, pp. 443–461, 2014.
- [37] M. S. Fabian, K. Gjergji, and W. Gerhard, “Yago: a core of semantic knowledge unifying Wordnet and Wikipedia,” in *Proceedings of the 16th International World Wide Web Conference, WWW*, pp. 697–706, Alberta, Canada, May 2007.
- [38] K. Girithana and S. Swamynathan, “Semantic query-based patent summarization system (SQPSS),” in *Proceedings of the International Conference on Intelligent Information Technologies*, pp. 169–179, Springer, Sochi, Russia, September 2018.
- [39] Z. Cao, F. Wei, L. Dong, S. Li, and M. Zhou, “Ranking with recursive neural networks and its application to multi-document summarization,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.
- [40] K. Girithana and S. Swamynathan, “Query oriented extractive-abstractive summarization system (QEASS),” in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pp. 301–305, Kolkata, India, January 2019.
- [41] R. Nallapati, F. Zhai, and B. Zhou, “Summarunner: a recurrent neural network based sequence model for extractive summarization of documents,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2017.
- [42] Y. Zhang, M. J. Er, R. Zhao, and M. Pratama, “Multiview convolutional neural networks for multidocument extractive summarization,” *IEEE transactions on cybernetics*, vol. 47, no. 10, pp. 3230–3242, 2016.
- [43] J. Cheng and M. Lapata, “Neural summarization by extracting sentences and words,” 2016, <https://arxiv.org/abs/1603.07252>.
- [44] N. Ramesh, B. Zhou, C. Gulcehre, and B. Xiang, “Abstractive text summarization using sequence-to-sequence RNNs and beyond,” 2016, <https://arxiv.org/abs/1602.06023>.
- [45] K. Cho, B. Van Merriënboer, C. Gulcehre et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” 2014, <https://arxiv.org/abs/1406.1078>.
- [46] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in *Advances in Neural Information Processing Systems*, pp. 3104–3112, MIT Press, Cambridge, MA, USA, 2014.
- [47] A. M. Rush, S. Chopra, and J. Weston, “A neural attention model for abstractive sentence summarization,” 2015, <https://arxiv.org/abs/1509.00685>.
- [48] S. Chopra, M. Auli, and A. M. Rush, “Abstractive sentence summarization with attentive recurrent neural networks,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 93–98, San Diego, CA, USA, June 2016.
- [49] A. J. C. Trappey, C. V. Trappey, and C.-Y. Wu, “Automatic patent document summarization for collaborative knowledge systems and services,” *Journal of Systems Science and Systems Engineering*, vol. 18, no. 1, pp. 71–94, 2009.
- [50] A. J. C. Trappey and C. V. Trappey, “An R&D knowledge management method for patent document summarization,” *Industrial Management & Data Systems*, vol. 108, no. 2, pp. 245–257, 2008.
- [51] N. Bouayad-Agha, G. Casamayor, G. Ferraro, S. Mille, V. Vidal, and L. Wanner, “Improving the comprehension of legal documentation: the case of patent claims,” in *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, pp. 78–87, Barcelona, Spain, June 2009.
- [52] Y.-H. Tseng, C.-J. Lin, and Y.-I. Lin, “Text mining techniques for patent analysis,” *Information Processing & Management*, vol. 43, no. 5, pp. 1216–1247, 2007.
- [53] E. Y. Igde, S. Aydogan, F. E. Boran, and D. Akay, “Linguistic summarization of structured patent data,” *International Journal of Computer and Information Engineering*, vol. 11, no. 9, pp. 1062–1065, 2017.
- [54] N. B. Bynagari, “The difficulty of learning long-term dependencies with gradient flow in recurrent nets,” *Engineering International*, vol. 8, no. 2, pp. 127–138, 2020.
- [55] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [56] A. Vaswani, N. Shazeer, N. Parmar et al., “Attention is all you need,” in *Advances in Neural Information Processing Systems*, pp. 5998–6008, MIT Press, Cambridge, MA, USA, 2017.
- [57] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” 2014, <https://arxiv.org/abs/1409.0473>.
- [58] M.-T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” 2015, <https://arxiv.org/abs/1508.04025>.
- [59] A. See, P. J. Liu, and C. D. Manning, “Get to the point: summarization with pointer-generator networks,” 2017, <https://arxiv.org/abs/1704.04368>.
- [60] Y. Wu, M. Schuster, Z. Chen et al., “Google’s neural machine translation system: bridging the gap between human and machine translation,” 2016, <https://arxiv.org/abs/1609.08144>.
- [61] T. L. Griffiths and M. Steyvers, “Finding scientific topics,” *Proceedings of the National Academy of Sciences*, vol. 101, no. 1, pp. 5228–5235, 2004.
- [62] W. Zhang, R. A. J. Clark, Y. Wang, and W. Li, “Unsupervised language identification based on latent Dirichlet allocation,” *Computer Speech & Language*, vol. 39, pp. 47–66, 2016.
- [63] J. Chang, J. Boyd-Graber, S. Gerrish, C. Wang, and D. M. Blei, “Reading tea leaves: How humans interpret topic models,” in *Proceedings of the Advances in Neural Information Processing Systems*, Vancouver, Canada, 2009.
- [64] C. Sievert and K. Shirley, “LDAvis: a method for visualizing and interpreting topics,” in *Proceedings of the Workshop on*

- Interactive Language Learning, Visualization, and Interfaces*, pp. 63–70, Baltimore, MD, USA, June 2014.
- [65] Z. Khaleghi, M. Fakhredanesh, and M. Hourali, “MSCSO: extractive multi-document summarization based on a new criterion of sentences overlapping,” *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 45, no. 1, pp. 195–205, 2021.
- [66] P. Verma and H. Om, “MCRMR: maximum coverage and relevancy with minimal redundancy based multi-document summarization,” *Expert Systems with Applications*, vol. 120, pp. 43–56, 2019.
- [67] W. M. Wang, Z. Li, J. W. Wang, and Z. H. Zheng, “How far we can go with extractive text summarization? heuristic methods to obtain near upper bounds,” *Expert Systems with Applications*, vol. 90, pp. 439–463, 2017.
- [68] S. Teufel, “Sentence extraction as a classification task,” in *Proceedings of the Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain, July 1997.
- [69] R. Ferreira, L. de Souza Cabral, R. D. Lins et al., “Assessing sentence scoring techniques for extractive text summarization,” *Expert Systems with Applications*, vol. 40, no. 14, pp. 5755–5764, 2013.
- [70] E. Agirre, C. Banea, D. Cer et al., “SemEval-2016 task 1: semantic textual similarity, monolingual and cross-lingual evaluation,” in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 497–511, ACL (Association for Computational Linguistics), San Diego, CA, USA, June 2016.
- [71] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From word embeddings to document distances,” in *Proceedings of the International Conference on Machine Learning (PMLR)*, pp. 957–966, Lille, France, July 2015.
- [72] J. Pennington, R. Socher, and C. D. Manning, “Glove: global vectors for word representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014.
- [73] X. Rong, “Word2vec parameter learning explained,” 2014, <https://arxiv.org/abs/1411.2738>.
- [74] C.-Y. Lin, “Rouge: a package for automatic evaluation of summaries,” in *Proceedings of the Workshop on Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004.
- [75] Kingma, P. Diederik, and B. Jimmy, “Adam: a method for stochastic optimization,” 2014, <https://arxiv.org/abs/1412.6980>.