# Claim-Wise Interaction Modeling with Hybrid Context and Domain-Adaptive Dictionary for Japanese Patent Retrieval

Zelin Zhang[1] Hiroki Tarutani[1] Toshiaki Nakazawa[2] Fei Cheng[1]
Chenhui Chu[1]
[1]Kyoto University [2]The University of Tokyo
{zelin,tarutani}@nlp.ist.i.kyoto-u.ac.jp {feicheng,chu}@i.kyoto-u.ac.jp
nakazawa@nlab.ci.i.u-tokyo.ac.jp

## Abstract

Retrieving Japanese patents presents a unique confluence of linguistic and structural challenges due to semantic compression and Japanese morphological ambiguity. To address this, we propose Claim-Wise Interaction Modeling with Hybrid Context and Domain-Adaptive Dictionary (CWIM-HCDAD). Our approach uniquely combines a domain-adaptive lexical retriever with a neural semantic retriever. We employ the Frequency-weighted Likelihood Ratio (FLR) to construct a technical term dictionary for BM25. Parallelly, we utilize a dual-vector architecture to mix claim-wise vectors on the hybrid semantic meaning of the abstract and main content. Evaluations on the GENIAC dataset demonstrate that CWIM-HCDAD outperforms baselines for retrieving claims containing complex technical compounds and variable identifiers.

## 1 Introduction

Unlike web search, where precision is often sufficient, patent retrieval is a high-recall task [1]. Missing a single relevant document can result in serious legal problems. This requirement asks retrieval systems to understand not just the broad theme of the patents, but the specific, granular technical elements defined in the patent claims. The analysis of Japanese patents is in Appendix A. We identified the following two challenges:

Firstly, most neural Information Retrieval (IR) systems are designed for document retrieval, treating the document as a sequence of sentences [1]. However, the legal force of a patent resides in its Claims. A claim is a single sentence that lists the individual elements of an invention. For a prior art document to anticipate a claim, it must disclose each and every element of that claim. Traditional architectures encode a single claim for performing a fuzzy match that captures the theme of the patent, but lose context [2].

Secondly, standard dense retrieval models achieve high recall on controlled test collections with small corpora but suffer drastic performance degradation when deployed on massive real-world corpora [2]. In a small corpus, theme alignment is often sufficient to isolate the patent. However, in a corpus of millions, thousands of patents share the same theme. The single-vector model cannot distinguish the true prior art from the theme matches.

To address these problems, we proposed Claim-Wise Interaction Modeling with Hybrid Context and Domain-Adaptive Dictionary (CWIM-HCDAD). Based on the architecture presented in Figure 1, our system includes three highlights:

- Domain-Adaptive Lexical Filtering: We address the "semantic drift" of dense vectors by incorporating an FLR [3][4] based compound term extraction for constructing a keyword dictionary, ensuring formulation queries of keyword retrieval using only technical terms critical for retrieval. Retrieval accuracy for similar patent documents improved by over 5.6% compared to dictionaries constructed using other methods.
- Hybrid Context (H-Context) in Embedding: We adopt hybrid contexts in the embedding stage, explicitly adding context claims that depend on the texts to be embedded. As shown in Table 1, this proposal improves the experiment results with 3.4% in Ax and 19.23% in Ay.
- Claim-level MaxSim operator [5]: We adopt a two-stage retriever on Ax Vectors and Ay Vectors seperately to extend the searching region and distinguish

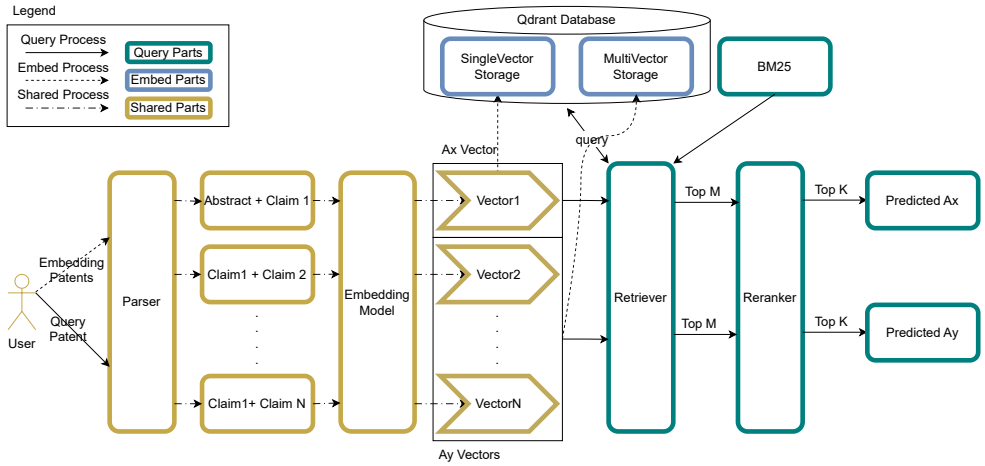


**Figure 1** The overview of architecture

the hard negatives. As shown in Table 1, this proposal improves the experiment results with 6.6% in Ax and 38.1% in Ay.

## 2 Task Definition

The evaluation is conducted on a large-scale corpus comprising approximately 4 million patent documents. For a given query patent, the retriever is tasked with retrieving the Top-100 most relevant prior art documents. The relevance targets are categorized into two specific sub-tasks:

- Task Ax (Primary Claim Match): Documents that are semantically most similar to Claim 1, the independent claim of the query patent.
- Task Ay (Secondary Claim Match): Documents that correspond to the dependent claims of the query patent. This match evaluates the system's ability to perform both broad conceptual retrieval based on the theme and specific retrieval based on technical details within a massive search space.

## 3 Architecture

### 3.1 The Hybrid Parser

The entry point is the Parser, which creates Hybrid Context Pairs.

- Abstract + Claim 1: We concatenate the Abstract (as context) with Claim 1 (legal scope) to create a Head representation that grounds abstract legal terms in technical reality.
- Claim context propagation: Dependent claims are explicitly paired with their parent independent claim to ensure the embedding model receives a self-contained semantic unit.

### 3.2 The Dual-Vector Encoding

The Embedding Model outputs two distinct types of vectors:

- Ax Vector (SingleVector Storage): Captures the global semantic theme. Used for coarse filtering.
- Ay Vectors (MultiVector Storage): Captures local interaction sites. Used for fine-grained MaxSim operations.

### 3.3 Domain-Adaptive Lexical Dictionary

We complement vector retrieval with Okapi BM25 [6] to ensure exact keyword matching. To address Japanese oversegmentation, we utilize an FLR-based domain dictionary [3][4] for recognizing technical compounds.

FLR quantifies the "unitariness" (coupling strength) of constituent words, distinguishing valid technical terms from frequent but loose collocations. The algorithm is defined as:

$$\mathrm{FLR(W)} = f(W) \times \left( \prod_{i=1}^{n} (LN(w_i) + RN(w_i) + 1) \right)^{\frac{1}{2n}}$$

where $f(W)$ denotes the frequency of the candidate compound term $W$ in the corpus, and $n$ is the number of constituent simple nouns $w_i$ in $W$. The constituent simple nouns are identified using the MeCab morphological analyzer [7]. The term $w_i$ represents the $i$-th constituent noun. $LN(w_i)$ and $RN(w_i)$ denote the total frequency of nouns

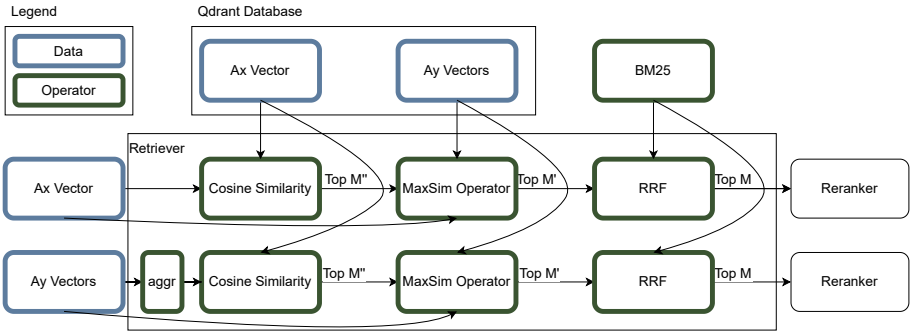


**Figure 2** The retriever architecture

appearing as left and right neighbors of the constituent word $w_i$, respectively.

To suppress generic high-frequency phrases captured by FLR, we introduce a composite score $S(W)$ weighted by Inverse Document Frequency (IDF):

$$S(W) = \text{FLR}(W) \times \left( \log_{10} \left( \frac{N}{df(W) + 1} \right) + 1 \right)$$

where $N$ is the corpus size and $df(W)$ is the document frequency. We extract the top-$k$ terms (length $n = 1$ to 8) based on $S(W)$ to construct the dictionary.

### 3.4 The Interaction of Encoding

This progress is shown in Figure 2. The final retrieval score is computed by fusing the semantic and lexical signals.

1. Global Filtering: The system computes cosine similarity between Ax Vectors to rapidly eliminate millions of irrelevant patents.
2. The MaxSim Operator: The MaxSim Operator applied to the Top candidates:

$$\text{S}_{\text{MaxSim}}(Q, D) = \sum_{q \in Q} \max_{d \in D} (q \cdot d)$$

   This equation measures the total semantic relevance between a query patent and a database patent by iterating through each individual claim vector $q$ in the query set $Q$, identifying its most similar counterpart $d$ in the candidate set $D$ via the highest similarity score, and then summing these maximum values to aggregate how well the entire scope of the query's technical claims is covered by the document in the database.
3. Hybrid Fusion (RRF): To robustly handle exact keyword matches that neural models might hallucinate, we merge the rankings from $\text{S}_{\text{MaxSim}}$ and the dictionary-based BM25 via Reciprocal Rank Fusion (RRF). RRF is a rank aggregation method that combines results from multiple retrievers without requiring score normalization. Since the score distributions of vector similarity and BM25 differ significantly, RRF provides a stable fusion mechanism. The RRF score for a document $d$ is calculated as:

$$\text{Score}_{\text{RRF}}(d) = \frac{w_{\text{neural}}}{\eta + r_{\text{neural}}(d)} + \frac{w_{\text{lexical}}}{\eta + r_{\text{lexical}}(d)}$$

   where $r_{\text{neural}}(d)$ and $r_{\text{lexical}}(d)$ denote the rank of document $d$ in the retrieval results sorted by $\text{S}_{\text{MaxSim}}$ and BM25 scores, respectively. $\eta$ is a smoothing constant (typically set to 60) used to mitigate the impact of high-ranking documents dominating the fusion score.

## 4 Experiments

### 4.1 Experimental Settings

For retriever experiments, we utilize a dataset partitioned into training, testing, and validation sets with sizes of 72,069, 13,514, and 4,504, respectively. This corresponds to an approximate split ratio of 80:15:5. To evaluate the effectiveness of the retrieval system, we report Recall@100, measuring the proportion of ground-truth documents present in the top-100 retrieved candidates. We employ BAAI/bge-multilingual-gemma2 as the backbone model for the dense retriever and BAAI/bge-reranker-v2.5-gemma2-lightweight for the reranking stage. The details of training are in Appendix B. The experimental environment is in Appendix C.

For the BM25 component, we compare three methods of constructing the dictionary: (1) a dictionary where keywords are derived by segmenting category names from the IPC classification table [8] using symbols; (2) a dictionary comprising keywords extracted from the abstract and

**Table 1** The results of Claim-wise 2-stage filtering. The values in parentheses denote the relative improvement over the Baseline.

| Method | Ax-Recall@100 | Ay-Recall@100 |
|---|---|---|
| Baseline | 42.22 | 22.16 |
| H-Context | 43.66 (+3.4%) | 26.44 (+19.3%) |
| *2-Stage Filtering* | | |
| 200:100 | **45.01** (+6.6%) | **30.61** (+38.1%) |
| 300:100 | 44.81 (+6.1%) | 30.60 (+38.1%) |
| 500:100 | **45.01** (+6.6%) | **30.61** (+38.1%) |

**Table 2** BM25 with different dictionary

| Method | Ax-Recall@100 | Ay-Recall@100 |
|---|---|---|
| category-csv | 14.3 | 2.1 |
| juman++ wikipedia | 11.0 | 8.8 |
| FLR dict | **19.9** | **14.6** |

claims that are tagged as "extracted from Wikipedia" by Juman++ [9][10] to target rare terms and exclude general vocabulary; and (3) the dictionary constructed using the method described in Section 3.3. We formulate queries using terms present in both the query patent's abstract/claims and the respective dictionary to perform BM25 retrieval, measuring the proportion of ground-truth documents present in the top-100 retrieved candidates.

## 4.2 Retriever

As shown in Table 1, the proposed method significantly outperforms the baseline. While the inclusion of H-Context (Hybrid Context, which is combining claim 1 to complement the information of claim i) provides a moderate improvement, the Claim-wise 2-stage filtering strategy yields the most substantial performance gains. Specifically, under the 200:100 setting, the method achieves relative improvements of 6.6% on Ax-Recall@100 and 38.1% on Ay-Recall@100. Notably, the gain on Ay-Recall@100 is considerably larger than that on Ax-Recall@100, suggesting that the filtering strategy is particularly effective for the more challenging Ay subset. Furthermore, the results are highly consistent across different initial pool sizes ($N = 200, 300, 500$).

## 4.3 BM25

As shown in Table 2, the FLR-based dictionary achieved the highest performance. While baseline methods relying on simple category segmentation or heuristic tags for rare words proved to be too coarse, our results demonstrate that explicitly quantifying "term technicality" for dictionary construction significantly enhances retrieval accuracy.

**Table 3** Comparison of Recall@100 among BM25, Vector Search, and Hybrid Search (RRF).

| Method | Ax-Recall@100 | Ay-Recall@100 |
|---|---|---|
| BM25 | 18.2 | 15.7 |
| Vector Search | 41.8 | 36.9 |
| **RRF (Hybrid)** | **42.0** | **36.9** |
| **Candidate Pool (Top-200)** | **44.7** | **42.0** |

## 4.4 Impact of Hybrid Search

We evaluated a Hybrid Search combining BM25 and our dense retriever via Reciprocal Rank Fusion (RRF). Results are reported on the common subset where both Vector Search and BM25 outputs are available, leading to different figures from earlier experiments. Merging the top-200 candidates with fusion parameters $k = 59$, $w_{\text{lex}} = 0.42$, and $w_{\text{neu}} = 1.0$, Table 3 shows that Vector Search significantly outperforms BM25. Hybrid Search yields slight gains on $A_x$ (42.0%) but maintains performance on $A_y$. However, the high recall of the top-200 candidate pool ($A_x$: 44.7%, $A_y$: 42.0%) indicates that while relevant documents are retrieved, the current fusion strategy fails to promote them to the top-100. This suggests that further refinement in the ranking mechanism is required to fully capitalize on the retrieved candidates.

# 5 Conclusion

In this paper, we proposed CWIM-HCSD, a novel framework designed to address the linguistic and structural complexities of Japanese patent retrieval. By synergizing domain-adaptive lexical filtering based on FLR with a dual-vector neural architecture, our approach effectively resolves the trade-off between semantic generalization and token-level precision. Experimental evaluations on a large-scale dataset of approximately 4 million documents demonstrated that CWIM-HCSD outperforms standard dense retrieval baselines. These results confirm that explicit interaction modeling and hybrid context preservation are essential for high-recall tasks in the patent domain. Furthermore, our analysis of hybrid search revealed that while the candidate pool achieves high recall, standard fusion methods like RRF do not fully capitalize on this potential. Future work will focus on developing more sophisticated ranking mechanisms to bridge the gap between candidate generation and final retrieval performance.

## Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP23K28144.

## References

[1] Julian Risch, Nicolas Alder, Christoph Hewel, and Ralf Krestel. Patentmatch: A dataset for matching patent claims prior art, 2020.

[2] Orion Weller, Michael Boratko, Iftekhar Naim, and Jinhyuk Lee. On the theoretical limitations of embedding-based retrieval, 2025.

[3] Hiroshi Nakagawa, Hiroaki Yumoto, and Tatsunori Mori. 出現頻度と連接頻度に基づく専門用語抽出. 自然言語処理, Vol. 10, No. 1, pp. 27–45, 2003.

[4] Hiroshi Nakagawa and Tatsunori Mori. A simple but powerful automatic term extraction method. pp. 1–7, 2002.

[5] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert, 2020.

[6] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In **Proceedings of the Third Text REtrieval Conference (TREC-3)**, pp. 109–126. NIST, 1995.

[7] Kaoru Yamamoto Taku Kudo and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In **Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing**, pp. 230–237, Barcelona, Spain, 2004. Association for Computational Linguistics.

[8] Japan Patent Office. Ipc classification table and update information (japanese version). https://www.jpo.go.jp/system/patent/gaiyo/bunrui/ipc/ipc8wk.html. Accessed: 2026-01-08.

[9] Daisuke Kawahara Hajime Morita and Sadao Kurohashi. Morphological analysis for unsegmented languages using recurrent neural network language model. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 2292–2297, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[10] 森田一, 黒橋禎夫. RNN 言語モデルを用いた日本語形態素解析の実用化. 情報処理学会 第 78 回全国大会講演論文集, 横浜 (慶應義塾大学), 3 2016. 情報処理学会. 第 78 回全国大会.

[11] Matthew Henderson, Rami Al-Rfou, Brian Strope, Yunhsuan Sung, Laszlo Lukacs, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. Efficient natural language response suggestion for smart reply, 2017.

[12] Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering, 2021.

[13] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

# A Structure of Japanese Patent Documents

The Japanese patent documents (JPO XML) are structured hierarchically. For the task of prior art retrieval and novelty detection, we specifically focus on two critical sections: the Abstract and the Claims.

## A.1 Abstract

The <abstract> serves as a condensed semantic representation of the document, often used as the target unit for initial retrieval or relevance scoring.

- Content: It summarizes the technical problem (<tech-problem>) and the solution (<tech-solution>) in a compact form. Comparing the query claim against the abstracts of potential prior art documents is a standard approach to filter candidates efficiently before full-text analysis.

## A.2 Claims

The <claims> section defines the legal scope of the invention and serves as the primary query for validity verification.

- Structure: This section consists of a sequence of <claim> elements. Each element is identified by a num attribute (e.g., <claim num="1">).
- Content: Inside each <claim>, the technical features of the invention are described in <claim-text>. In novelty detection tasks, the independent claim (typically Claim 1) is of utmost importance as it encompasses the broadest scope of the invention. Dependent claims refer back to preceding claims to add specific limitations.

# B Training

## B.1 Training of Retriever

Retriever is trained using a composite objective combining Multiple Negatives Ranking Loss (MNRL) [11] and Hard Sample Mining (HSM) distillation [12].

1. MNRL: We first employ standard MNRL to optimize the embeddings using in-batch negatives, ensuring coarse-grained semantic alignment between the query claim and the document.
2. HSM (Distillation): To capture fine-grained interactions, we apply a Score Distillation objective. The Retriever mines Hard Negatives with high vector similarity but low ground-truth relevance. These hard samples are scored by an independent eeranker. We use the Reranker's continuous relevance score as the target for a Margin-MSE loss, forcing the Student to reproduce the Teacher's precise margin between the positive and the hard negative, rather than treating all negatives as equal zeros.

## B.2 Training of Reranker

The training data for the Reranker is composed of Ground Truth (GT) positive pairs augmented with the HSM Data. The specific hard negatives are identified by the Retriever during its training phase. By explicitly training on the Retriever's confusion set (HSM), the Reranker learns to correct specific structural hallucinations.

# C Experimental Environment

All retriever experiments are conducted on a computational environment equipped with 4 NVIDIA A100 GPUs with LoRA [13] to save GPU memory.