JAPAS: A Benchmark and Neural Approach for Japanese Patent Support Relation Extraction

Katsuki Chousa, Ryosuke Sugiura
NTT, inc.
{katsuki.chousa, ryosuke.sugiura}@ntt.com

Abstract

Efficient analysis of patent literature is crucial for technological development and protecting intellectual property. A key task is verifying the "support requirement," which mandates that the detailed description must fully describe the claimed invention. This requirement is fundamental to a patent's validity. Manual verification is a labor-intensive process that demands technical and legal expertise, making automation highly desirable. However, research on this task has been hampered by two key challenges: (1) the absence of a public benchmark, and (2) the reliance of prior work on lexical matching, which fails to capture semantic equivalence. To address these issues, we introduce JAPAS, the first public benchmark for this task, comprising over 2,000 instances manually annotated for Japanese patents. Each instance is labeled with a claim span, a supporting description paragraph, a relation type, and the annotator's confidence level. Using this benchmark, we also establish modern baselines that capture semantic similarity, such as embeddings and LLMs. Our experiments show that a fine-tuned Qwen3-14B model achieves an F1 score of 0.50, outperforming the conventional lexical-based baseline. This result, which demonstrates that the task is feasible yet challenging, highlights the utility of JAPAS as a research foundation and provides a performance target for future work.

Keywords: claim analysis, alignment, dataset

1. Introduction

Patent documents are a rich source of detailed information on the latest technological innovations, and their analysis is essential for technological development and protecting intellectual property. A patent application document is composed of patent claims and detailed descriptions. The claims define the legal scope of the protection, while the detailed descriptions provide exhaustive information about the invention. Under national and international patent law [1], the claim scope must be fully supported by the detailed descriptions: They should include some statements by which a person skilled in the art can understand the claimed invention and how it works; failure to meet this requirement can result in the patent being refused or later invalidated.

Consequently, it is necessary to verify whether every claim is adequately supported by the detailed descriptions. Figure 1 shows an example of the support relationships. However, this process is labor-intensive and requires deep domain knowledge of both the technology and intellectual property law: Each claim often comprises several statements describing the structure of the invention, which may be scattered throughout the detailed description at different levels of abstraction. Automating this task would offer significant benefits to various professionals. For patent examiners, it could lead to more efficient and consistent examinations. For applicants and patent practitioners, it would assist in drafting high-quality applications.

To address this challenge, several automatic support-relation extraction methods have been proposed. The existing studies formulate the task as an alignment problem between claims and detailed descriptions. Murata and Isahara (2002) proposed a global alignment method using the Unix `diff` command to identify correspondences between claims and detailed description paragraphs. Shinmori et al. (2004) introduced a local alignment approach that leverages the discourse structure of claims and lexical similarity. However, these methods rely on exact lexical matching and therefore miss support relations when the claim and detailed description use different lexical choices. Although recent pre-trained models can capture semantic similarity, they have not yet been applied to this task. Second, and more fundamentally, a public benchmark for advancing research in this area has been absent. Prior work either lacks quantitative evaluation or uses in-house datasets, leaving no publicly available benchmark for the support-relation extraction task and thus impeding systematic progress.

Here, we present JAPAS, a manually annotated dataset of 2,056 support-relation instances extracted from Japan Patent Office (JPO) applications. The annotation was conducted by annota-

[1] For example, this requirement is codified in Article 36 of the Japan Patent Act, 35 U.S.C. §112 in the United States, Article 84 EPC in the EU, and Article 6 of the Patent Cooperation Treaty (PCT)

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 4188–4198
11–16 May 2026.

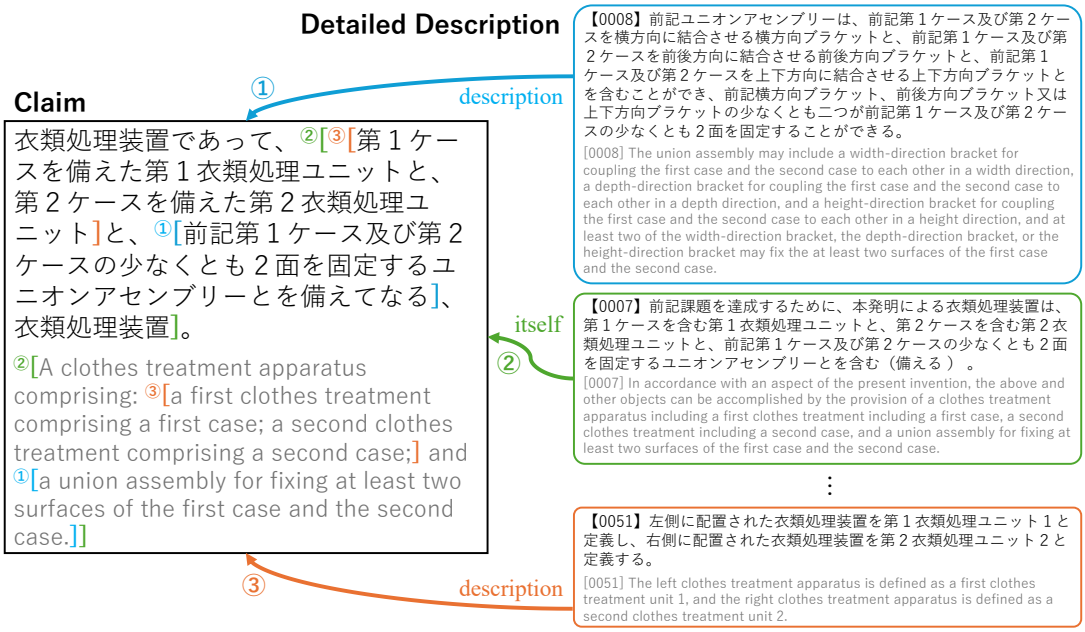

Figure 1: An example of support relationships within a patent document. Each colored passage from the detailed description supports the phrase in the claim enclosed in brackets of the same color. English translations are provided below the original Japanese text for reference.

tors with substantial experience in reading patent documents, following a detailed schema designed in consultation with patent experts. Each instance is annotated with 1) a claim span, corresponding to each component of the invention structure, 2) a supporting paragraph of a detailed description, 3) the type of support relation, and 4) a confidence label. With JAPAS, we have established the first public benchmark for the support-relation extraction task. We also propose novel support relation extraction models that utilize semantic similarity with text embeddings and large language models (LLMs) to establish strong baselines for future research. Our experiments show that fine-tuning a Qwen model on JAPAS achieved a +11 F1 point improvement over a conventional lexical-based baseline. This result demonstrates that semantic-aware representations markedly enhance automated patent claim support verification and underscores the value of our task-specific annotated dataset.[2]

2. Related Work

Several large-scale patent corpora have been released for tasks such as patent-topic classification, information retrieval, and abstractive summarization (Iwayama et al., 2003; Li et al., 2018; Sharma et al., 2019; Suzgun et al., 2023; Piroi et al., 2011). While these resources provide annotations for each textual segment in a patent, indicating its corresponding data field (e.g., title, abstract, claims, detailed description), they lack annotations for relations *between* these fields, such as support relations. As a result, there is currently no publicly available benchmark for support-relation extraction, and this has impeded progress on this task.

Early attempts at extracting support relations focused on aligning claims with the detailed description. Murata and Isahara (2002) proposed a global alignment method using the Unix `diff` command, while Shinmori et al. (2004) parsed claims into discourse structures to perform local alignment between key elements and sentences in the description. The primary limitation of these methods, however, is their reliance on lexical matching to assess text similarity, making them less effective at handling paraphrases.

Other approaches have also been proposed to evaluate the degree of support at the document level by calculating the similarity between the entire claims and description, using either sets of words (Mann and Underweiser, 2012) or topic distributions from LDA (Khachatryan and Muehlmann, 2020). However, these methods have a significant limitation: because they calculate a single similarity score for the entire document, they cannot identify which specific parts of a claim are unsupported. Furthermore, similar to the methods mentioned above, their datasets are not publicly available, which has made reproducible evaluation and comparison difficult.

[2]The dataset is available at https://www.kecl.ntt.co.jp/icl/lirg/japas/.

More generally in NLP, recent research has utilized semantic similarity based on text embeddings to overcome the limitations of lexical matching. E5 (Wang et al., 2024a) and Multilingual E5 (Wang et al., 2024b) are text embedding models for English and multilingual contexts, respectively. Both models have achieved state-of-the-art performance on various benchmarks, including information retrieval and the Semantic Textual Similarity (STS) task. In the case of Japanese, specialized embedding models such as Ruri (Tsukagoshi and Sasano, 2024) and Sarashina-Embeddings[3], which build on Japanese pre-trained language models, have demonstrated superior performance on comparable benchmarks. Nevertheless, the effectiveness of these methods on the support-relation extraction task remains to be investigated. Establishing a benchmark and evaluating such models in this context is therefore an essential step toward advancing automated patent analysis.

3. Dataset Construction

3.1. Resources

The data source for this study is the bulk-download service for patent information provided by the Japan Patent Office (JPO). The JPO offers this service free of charge[4]. We obtained the patent documents by sending a hard disk to the JPO, which they returned to us containing the requested data.

3.2. Data Selection and Preprocessing

We focused on the patents filed as international applications under the Patent Cooperation Treaty (PCT), rather than those filed directly with national patent offices. A PCT application is written in a unified format as a single international filing, and this structure is preserved during the subsequent national-phase procedures in each contracting state. This unified format allows any PCT application filed with one state to be readily transferred into applications for other member states. Therefore, in regard to future prospects, it is much easier to extend the dataset to include other languages if we use PCT applications. Additionally, PCT applications are more expensive than direct applications for each state, and thus, they are more likely to be carefully organized and technically reliable, thereby providing a high-quality source for annotation.

From the PCT applications published by the JPO in 2019, we randomly selected 62 patents. For each patent, we extracted the *Detailed Description* and *Claims* sections from the provided XML files. Due to budget constraints, we limited our annotation to the first four claims of each patent. This is a reasonable simplification, as the initial claims typically define the broadest and most fundamental inventive concepts.

3.3. Annotation Schema

We constructed the dataset by defining a *support relation* between a text span in a claim and a paragraph in the detailed description. The annotation guidelines were established through discussions with patent experts. The schema comprises four elements:

Claim Span The start and end character offsets of a contiguous textual span within a claim. The span typically corresponds to a clause containing a predicate.

Description ID The paragraph number of the description that supports the Claim Span.

Relation Type A three-way categorization of the support relation: (a) `itself`: The paragraph restates the claim text verbatim or with minor paraphrasing. (b) `description`: The paragraph provides definitions, elaborations, or detailed explanations of the claim text. (c) `example`: The paragraph presents embodiments, examples, or variations corresponding to the claim text.

Confidence The annotator's self-assessed certainty about the assigned label. We defined three confidence tiers: `high` for cases where the annotator is more than 80% certain; `medium` for certainty between 50-80%; and `low` for certainty below 50%.

These annotated elements are stored in a single JSON file for each patent. Figure 2 shows an example of our data format.

Note that a single Claim Span can be supported by multiple description paragraphs, resulting in a one-to-many relationship.

3.4. Annotation Procedure

The annotation was conducted by annotators with substantial experience in reading patent documents and in document annotation. First, annotators carefully read a claim and identified textual spans at the clause level that require supporting evidence. Next, for each identified span, the annotator treated it as a query and searched the entire description to find paragraphs that provide support. Finally, for each identified claim-description

[3] https://huggingface.co/sbintuitions/sarashina-embedding-v1-1b

[4] https://www.jpo.go.jp/system/laws/sesaku/data/download.html

```
{
"claims": [{
    "id": "1",
    "text": "複数の通信装置を管理する管理装置で
    あって、...",
    "segments": [{
        "start": 0,
        "end": 20,
        "description_id": "0021",
        "relation_type": "description",
        "confidence": "high"
    }, ...]
  }, ...],
  "descriptions": [{
      "id": "0001",
      "text": "本発明は、通信システムに関する。"
  }, ...]
}
```


Figure 2: An Annotated Data in JSON Format

Subset	Train	Dev	Test
Patents	40	11	11
Claims /patent	4	4	4
Char. length /claim	175.6	210.6	232.6
Descriptions /patent	108.6	102.8	111.4
Char. length /desc.	228.4	209.1	224.6
Supports	1335	347	406
Supports /claim	8.34	7.89	9.23

Table 1: Statistics of the constructed dataset. "Char." and "fesc." stand for "Character" and "description," respectively.

pair, the annotator assigned one of the three Relation Types and recorded their confidence level. This process was repeated until almost all claim spans were linked to at least one supporting paragraph. After annotation, we split the dataset into training, development, and test sets in an approximate 4:1:1 ratio.

Table 1 and Table 2 present the statistics of the constructed dataset and the proportion of each label in the dataset. The training set contains 1,335 annotated instances, which is sufficient for fine-tuning LLMs. The development and test sets are also large enough for robust evaluation.

As shown in Table 1, we found an average of 8-9 support relations per claim. Considering that a typical patent application at the JPO in 2019 has an average of 11.0 claims (IP5 Offices, 2020) and that patents in our dataset have roughly 100 description paragraphs on average, this annotation number is plausible.

4. Proposed Methods

Verifying the support relation can be viewed as the task of aligning constituent parts of the claim with their supporting evidence in the description. A claim consists of multiple clauses describing different elements of an invention, and the evidence for these elements is often scattered across various paragraphs in the description. Therefore, we decided to formalize the task as a problem of aligning each paragraph in the description to the specific claim spans it supports. To tackle this problem, we propose and evaluate two approaches leveraging pre-trained models: an embedding-based method and an LLM-based one.

Relation	Prop.	Confidence	Prop.
itself	10.4 %	high	54.9 %
description	66.4 %	medium	30.2 %
example	23.2 %	low	14.9 %

Table 2: Distribution of relation-type and confidence levels in the dataset. Prop. indicates proportion of the whole dataset, expressed as a percentage.

4.1. Embedding-based Method

Prior studies in the field of information retrieval and on sequence alignment have demonstrated the usefulness of using text embeddings (Thompson and Koehn, 2019; Jalili Sabet et al., 2020; Dou and Neubig, 2021; Reimers and Gurevych, 2019). Inspired by these studies, we compute the similarity between the embedding of a claim span and that of a description paragraph to decide whether a support relation exists between them.

Let $C \coloneqq (c_1, \ldots, c_N)$ be the token sequence of a claim and $D \coloneqq (d_1, \ldots, d_M)$ be that of a description paragraph. A claim is segmented into the smallest units for which we wish to determine a support relationship: $S \coloneqq \{(s_1, e_1), \ldots, (s_L, e_L)\}$, where $s_\ell \in \mathbb{N}$ and $e_\ell \in \mathbb{N}$ are the start and end token indices of the ℓ-th span. Here, we simply use linguistic units as span units, such as an entire sentence or clauses segmented by punctuation.

First, we obtain token-level embeddings for the entire claim using a pretrained text-embedding model Emb:

$$(\mathbf{u}^c_1, \ldots, \mathbf{u}^c_N) = \mathrm{Emb}(c_1, \ldots, c_N), \qquad (1)$$

where $\mathbf{u}^c_n$ is the embedding vector of c_n. To capture contextual information, we encode the entire claim at once. Alternatively, for a context-free representation, we encode only the tokens within each individual span.

Next, each span (s_ℓ, e_ℓ) is represented by a single vector $\mathbf{v}^C_\ell$:

$$\mathbf{v}^C_\ell = \mathrm{pooling}\big(\mathbf{u}^c_k \mid s_\ell \le k \le e_\ell\big), \qquad (2)$$

where the pooling function follows the strategy of the embedding model (e.g., mean, or max). The

embedding for the description paragraph $\mathbf{v}^D$ is obtained in the same manner:

$$\mathbf{v}^D = \text{pooling}(\text{Emb}(d_1, \ldots, d_M)). \quad (3)$$

Finally, we compute a similarity score between the span embedding $\mathbf{v}_\ell^C$ and the description paragraph embedding $\mathbf{v}^D$:

$$\text{score}_\ell = \text{sim}(\mathbf{v}_\ell^C, \mathbf{v}^D), \quad (4)$$

where the similarity function also follows the setting used by the embedding model (e.g., cosine similarity). If the score exceeds a threshold τ, we classify the span as supported by the description paragraph. The threshold τ is determined by maximizing the F1 score on a development set.

4.2. LLM-based Method

In order to leverage large language models (LLMs) in this task, we use a structured prompt template. The prompt consists of system instructions and user input. The system instructions outline the task and specify the required output format, while the user input provides a claim and a description paragraph. While the prompts were developed in Japanese based on preliminary experiments, we present their English translations in Figure 3 for clarity and wider understanding.[5]

The LLM is expected to generate a JSON-formatted string containing the extracted claim span text, support type, and confidence level. Note that the model must generate the actual text of the span, not its character or token offsets. If no support relation is found, the model is instructed to return an empty list.

We have explored three settings for this method: (1) **Zero-shot**, where the model performs the task based only on the instructions; (2) **Few-shot**, where we include three input-output examples in the prompt to provide in-context learning; (3) **Fine-tuning**, where we adapt the model to our task by training it on our annotated dataset using the same prompt-response format. For fine-tuning, we use the Low-Rank Adaptation (LoRA) technique (Hu et al., 2022) to efficiently train the model, given the relatively small size of our dataset.

5. Experiments

We conducted two sets of experiments. First, we evaluated the proposed methods on the primary task of support relation extraction. Second, we assessed the best-performing model's ability to classify the relation type and confidence level of the extracted relations.

[5]The original Japanese prompt is shown in Appendix A.

```
You are an analyst who is well versed in
checking the support requirement for patent
specifications. For the claim and the detailed
description paragraph below, extract the spans
that supported by the description paragraph, and
 determine the relation type of support and the
confidence level.

# Extraction Items
1. Supported Span (Which spans of the claim are
supported by the description)
2. Relation Type
  - itself: The exact wording of the claim
  apperas in the specification.
  - description: The description provides a
  definition or detailed explanation of the
  claim langauge.
  - example: The description gives an
  embodiement, example, or variation that
  corresponds to the claim language.
3. Confidence (high/medium/low)

* When sevelal spans are supported within the
same claim, list them all.
* If no span is found, return an emply list.

# Output Format (JSON)
[{
  "claim_span": "<term or phrase within the
  claim>",
  "type": "itself | description | example",
  "confidence": "high | medium | low"
}, ...]

Claim: {claim}
Description: {description paragraph}
```

Figure 3: The English translation of the prompt used in the LLM-based model.

5.1. Support Relation Extraction

In this experiment, we evaluated the models' ability to identify support relations between claim spans and description paragraphs using our dataset.

5.1.1. Model Settings

We evaluated the embedding-based method and the LLM-based one as follows.

Embedding-based method We used two Japanese text embedding models: Ruri-v3-310m (Tsukagoshi and Sasano, 2024) and Sarashina-Embeddings-v1[6], both implemented using the `sentence-transformers` library [7]. For each model, we used its default pooling function (`mean-pooling` for Ruri, `last-token-pooling` for Sarashina) and cosine similarity. We experimented with two types of claim units for generating embeddings: the entire claim treated as a single sentence, and clauses segmented by commas[8]. For the clause-level

[6]https://huggingface.co/sbintuitions/sarashina-embedding-v1-1b

[7]https://sbert.net/

[8]We used three types of commas for Japanese texts: "、, ,"

units, we further evaluated two distinct encoding strategies: a context-free approach where each clause is encoded individually, and a contextual approach where the embedding for each clause is derived from the full claim text, thereby retaining context from surrounding clauses. In total, we compared six configurations for this method: 2 models × 3 embedding strategies.

LLM-based method We used Qwen3-14B[9] as our base LLM. We used unsloth [10] for fine-tuning[11] and vllm (Kwon et al., 2023) for inference.

Baseline As a baseline using only surface-level information, we used cosine similarity between sentence-level TF-IDF vectors. The TF-IDF vectors were computed for each patent individually, treating all its claims and descriptions as a single document for vocabulary construction. While other surface-based methods have been proposed in prior work, we chose this TF-IDF baseline due to the difficulty of reproducing earlier methods, whose implementations are not publicly available.

5.1.2. Evaluation Metrics

To evaluate model performance, we use two granularities: claim level and span level. The claim-level evaluation is the more fundamental metric, assessing the model's core ability to identify the correct supporting paragraph for a claim. In contrast, the span-level evaluation is a stricter metric that assesses the model's precision in pinpointing the exact text fragment being supported.

For the claim-level evaluation, we measured the F1-score, which is widely employed in sequence alignment tasks. To do so, we first aggregated our span-level annotations to the claim level. Specifically, if at least one ground-truth support relation exists between any span within a claim and a given description paragraph, we considered the entire claim and that paragraph to be a positive pair for evaluation. The claim-level precision (P_c), recall (R_c), and F1-score (F_c) are calculated as:

$$F_c = \frac{2 \times P_c \times R_c}{P_c + R_c} \tag{5}$$

$$P_c = \frac{|T_c \cap G_c|}{|G_c|} \tag{6}$$

$$R_c = \frac{|T_c \cap G_c|}{|T_c|} \tag{7}$$

where T_c is the set of predicted support relation pairs (claim, description paragraph), G_c is the set of ground-truth support relation pairs, and $|\cdot|$ denotes the size of the set.

For the span-level evaluation, we used the micro-averaged token-level F1-score, a metric used in span extraction tasks such as SQuAD v2 (Rajpurkar et al., 2018). Since the ground-truth spans are not pre-tokenized and different models use different token units, a unified tokenizer is required for a fair evaluation. In our experiments, we used fugashi (McCann, 2020).[12] The span-level F1-score (F_s) is the average F1-score over all claim-paragraph pairs in the test set, calculated on the basis of the overlap between predicted and ground-truth tokens within the spans.

$$F_s = \frac{1}{N} \sum_{i=1}^{N} \frac{2 \times P_s^{(i)} \times R_s^{(i)}}{P_s^{(i)} + R_s^{(i)}} \tag{8}$$

$$P_s^{(i)} = \frac{|T_s^{(i)} \cap G_s^{(i)}|}{|G_s^{(i)}|} \tag{9}$$

$$R_s^{(i)} = \frac{|T_s^{(i)} \cap G_s^{(i)}|}{|T_s^{(i)}|} \tag{10}$$

where N is the total number of pairs, and $T_s^{(i)}$ and $G_s^{(i)}$ are the multisets of tokens for the predicted and ground-truth claim spans for the i-th pair, respectively.

We report two variants of this metric: one that includes the performance for pairs with no support relation (`inc. NS`) and one that excludes them (`exc. NS`). When calculating `inc. NS`, if both the prediction and the ground truth for a given pair are "No Support", its F1-score is treated as 1. These metrics require caution because, as the statistics in Table 1 suggest, support relations are relatively sparse. Consequently, a model that always predicts "No Support" would achieve a deceptively high score on `exc. NS`, while a model that always predicts a pair as supportive would score deceptively high on `exc. NS`. Thus, there is a trade-off between these two metrics, making it crucial to improve both for a comprehensive assessment of model performance.

5.1.3. Results

The experimental results are presented in Table 3. Overall, most of our proposed methods surpassed the performance of the TF-IDF baseline. Furthermore, our results show a clear performance hier-

[9] We used the 4-bit quantized model from Unsloth: https://huggingface.co/unsloth/Qwen3-14B-unsloth-bnb-4bit

[10] https://unsloth.ai/

[11] For reproducibility, we show the hyperparameters for fine-tuning in Appendix B.

[12] We followed the Japanese SQuAD evaluation script in Stability-AI/lm-evaluation-harness: https://github.com/Stability-AI/lm-evaluation-harness/blob/jp-stable/lm_eval/jasquad/evaluate.py

Systems	Claim-level Precision	Recall	F1-score	Span-level F1-score inc. NS	exc. NS
TF-IDF	.39	.19	.25	.91	.16
SentEmb Ruri	.41	.35	.38	.91	.26
SentEmb Sarashina	.39	.39	.39	.90	.29
Context-free ClauseEmb Ruri	.30	.37	.33	.87	.24
Context-free ClauseEmb Sarashina	.07	1.0	.14	.03	.70
Contextual ClauseEmb Ruri	.45	.34	.39	.91	.25
Contextual ClauseEmb Sarashina	.28	.43	.34	.85	.23
Qwen3-14B 0-shot	.16	.53	.26	.74	.24
Qwen3-14B 3-shots	.16	.53	.26	.74	.24
Qwen3-14B Fine-tuning	.51	.48	.50	.91	.37

Table 3: Comparison of accuracies extracting support relations.

archy: the fine-tuned LLM achieved the best performance, followed by the embedding-based methods and the few-shot LLM approaches.

The best-performing model, Qwen3-14B Fine-tuning, obtained a claim-level F1-score of 0.50. While this represents a substantial improvement of +0.25 points over the TF-IDF baseline and +0.11 points over the best embedding-based method, SentEmb Sarashina, the absolute score of 0.50 suggests that the task is feasible for automated approaches yet remains a significant challenge. Notably, the fine-tuned LLM achieved a superior trade-off between the `inc. NS` and `exc. NS` metrics. It maintained a high `inc. NS` of 0.91, comparable to the TF-IDF baseline, demonstrating its ability to correctly identify the majority of non-supported pairs. At the same time, it boosted the `exc. NS` to 0.37, significantly outperforming all other methods, which indicates a strong ability to precisely identify correct text spans in supported pairs. This well-balanced performance means the model can accurately extract true positive spans while avoiding false positives.

In contrast, Context-free ClauseEmb Sarashina showed an anomalously high `exc. NS` of 0.70. This can be explained by its extremely low precision (0.07) and perfect recall (1.0) on the claim-level metric, indicating that the model classified nearly every pair as having a support relation. Consequently, its high `exc. NS` score is misleading, and its extremely low `inc. NS` score (0.03) confirms that it is not a practically useful model.

This clear performance hierarchy among our methods strongly suggests that while modern LLMs have strong general-purpose capabilities, specialized tasks like patent support verification require task-specific fine-tuning. This highlights the value of our newly constructed JAPAS dataset, which enables such fine-tuning and helps advance research in this area.

5.2. Relation Type and Confidence Classification

Next, we chose the best-performing model from the previous experiment, Qwen3-14B Fine-tuning and evaluated its ability to classify relation types and annotator confidence levels.

5.2.1. Settings

To evaluate the relation type classification, we defined a metric called claim-level labeled F1-score. This is an extension of the claim-level F1-score from Section 5.1.2. A prediction is counted as correct only if the predicted (claim, description paragraph) pair is correct and the predicted relation type label (e.g., `description`) exactly matches the ground-truth label. If a ground-truth pair has multiple relation type labels, the model must predict all of them correctly.

As for the confidence level classification, we measured the accuracy for each predicted label (`high`, `medium`, `low`). This allowed us to assess whether the model's confidence predictions align with the annotators' certainty.

5.2.2. Results

The classification results for relation types and confidence levels are presented in Table 4 and Table 5, respectively.

As for the relation type classification, the model achieved an overall labeled F1-score of 0.40, indicating a reasonable capability to identify relation types. Performance varied significantly across types. The `itself` relation, which involves near-verbatim repetition, was the easiest to classify, achieving a high F1-score of 0.58. The

Support Type	Labeled P/R/F	#pred
Overall	.38 / .41 / .40	377
- itself	.80 / .46 / .58	25
- description	.40 / .48 / .43	282
- example	.18 / .10 / .13	70

Table 4: Classification performance for relation types. Labeled P/R/F refers to the Precision, Recall, and F1-score for the claim-level labeled F1-score metric.

Confidence	Accuracy	#Pred
Overall	51.40%	377
- high	58.67%	236
- medium	39.52%	132
- low	33.33%	9

Table 5: Classification accuracy for confidence.

`description` relation was also identified with moderate success, achieving a F1-score of 0.43. In contrast, performance on the `example` relation was notably lower, with an F1-score of just 0.13. The difficulty here may stem from two factors: the wide range of topics and styles in `example`, and the smaller proportion of `example` instances in the training data.

As for the confidence level classification, the model's predictions showed a strong correlation with its empirical accuracy. The accuracy for predictions the model labeled as `high` (58.67%) was substantially higher than for those labeled `medium` (39.52%) or `low` (33.33%). This suggests that the model learned to estimate the reliability of its own predictions by learning from human-annotated confidence scores. However, the results also reveal a need for better calibration, as the model's high-confidence predictions were still incorrect over 40% of the time. This finding highlights the potential for future work on confidence calibration techniques and the need for larger-scale training data.

5.3. Analysis of Relation Type Classification

In the previous experiment, the model performed support relation extraction and relation type classification simultaneously. Consequently, the classification performance was potentially affected by the success or failure of the extraction step. Thus, we conducted a further analysis to evaluate the model's classification ability under an oracle setting, assuming that the support relations are correctly identified.

Specifically, we used the fine-tuned Qwen3-14B model with the 406 ground-truth support relation pairs (claim span and description paragraph) from the test set and had it predict only the relation type (`itself`, `description`, or `example`). The ground-truth claim span text was supplied to the model using forced decoding. We used accuracy as the evaluation metric.

Label	Correct / Total	Acc. (%)
`itself`	22 / 24	91.7
`description`	212 / 278	76.3
`example`	66 / 104	63.5
Overall	**300 / 406**	**73.9**

Table 6: Accuracy of relation type classification under an oracle setting.

The results are presented in Table 6. The overall accuracy was 73.9%. By label, `itself`, which has a high degree of lexical overlap, achieved the highest accuracy at 91.7%, followed by `description` at 76.3%. In contrast, `example`, which describes embodiments or variations of the invention, was the most difficult category to classify, with an accuracy of 63.5%. As noted in the previous section, this difficulty may stem from two factors: the wide range of topics and styles in `example`, and the smaller proportion of `example` instances in the training data. Furthermore, it may be attributed to the inherent ambiguity of the task since the difference between `description` and `example` is often a matter of abstraction. The stark disparity between the labeled F1-score for `example` in Section 5.2 (0.13) and the accuracy under this oracle setting (63.5%) strongly suggests that the primary cause of performance degradation in the end-to-end task is the difficulty of the relation "extraction" stage.

6. Conclusion

In this work, we addressed the task of extracting support relations between patent claims and their detailed descriptions. This area has been limited by a lack of public benchmarks and a reliance on surface-level lexical matching in prior methods.

To bridge this gap, we introduced JAPAS, a manually annotated dataset. Using JAPAS, we established the first public benchmark for this task. We also evaluated novel methods that utilize semantic similarity with text embeddings and LLMs to establish strong baselines for future research. Our experiments demonstrated that Qwen3-14B fine-tuned on JAPAS achieved a claim-level F1-score of 0.50, outperforming a TF-IDF baseline by a large margin of +0.25 F1 points. This result highlights that JAPAS is an effective, high-quality foun-

dation for solving this complex task, while the F1 score of 0.50 suggests that the task itself is feasible yet remains challenging.

Our dataset and benchmark will facilitate future research in automated patent analysis. Future work will include expanding the scale and linguistic diversity of the dataset.

7. Limitations

While JAPAS, as constructed in this work, provides a solid foundation for future research, it has several limitations.

First, although JAPAS is a significant contribution to the field, it is the first public benchmark for the patent support relation extraction task. That is, as it is our initial effort, it targets Patent Cooperation Treaty (PCT) applications, which offer high uniformity and are suitable for future expansion to other languages, but only in Japanese. Consequently, the scope of the current dataset is limited to Japanese. An important direction for future work is to build upon this foundation by expanding the dataset to other languages, such as English.

Second, JAPAS contains 2,056 manually annotated instances, which is a significant contribution to this field and whose scale is sufficient for evaluating the performance of various models and establishing strong baselines; however, the data was extracted from only 62 patents and thus does not completely cover the linguistic diversity across all technical fields. Future research could therefore aim to improve the dataset's scale and diversity by incorporating patents from a wider range of technical fields, which would further enhance the generalizability of models trained on it.

Third is the Inter-Annotator Agreement (IAA). In our annotation process, a single annotation is determined through discussion among multiple annotators. Maintaining this level of annotation quality inherently incurs high costs because of the expertise required. Due to budget constraints in this study, we assigned only a single annotation team to each data instance. Therefore, an IAA score, which quantitatively measures the consistency and objective reproducibility of the annotation process, has not been calculated. A future task is to conduct multiple annotations on a subset of the dataset to calculate and report a reliability coefficient, such as Cohen's Kappa, to further validate the annotation quality.

Fourth concerns the evaluation of practical utility. It is not yet clear what level of performance, as demonstrated in this study, is required to be useful to patent practitioners and to what extent it can improve their efficiency. Furthermore, practical benefits depend not only on model accuracy but also heavily on the design of the user interface that presents the model's output. Therefore, future work should include user studies with patent practitioners to identify practical performance targets and discuss the design of an effective interface to improve workflow efficiency.

8. Ethics Statement

The patent documents used in this research were obtained through the official bulk data service provided by the Japan Patent Office (JPO). All data is public information, and there are no privacy concerns.

The annotation for this dataset was commissioned to a company specializing in data construction for the Natural Language Processing (NLP) field. Annotators were notified in advance that the work would involve patent documents, and their informed consent was obtained. Patent documents are publicly available technical literature, and the likelihood of them containing offensive content or personally identifiable information is extremely low. The annotation was conducted by the company's expert staff, who have extensive experience in reading patent documents, and they were fairly compensated for their expertise .

The objective of this research is to assist patent examiners and practitioners with verifying support requirements. The developed technology is intended to assist professionals, and we anticipate a low risk of misuse that could cause negative societal impacts.

9. Bibliographical References

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

IP5 Offices. 2020. IP5 Statistics Report 2020 Edition. Accessed: 2025-09-24.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational*

Linguistics: EMNLP 2020, pages 1627–1643, Online. Association for Computational Linguistics.

Davit Khachatryan and Brigitte Muehlmann. 2020. Measuring the drafting alignment of patent documents using text mining. *PLOS ONE*, 15(7):1–20.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Ronald J. Mann and Marian Underweiser. 2012. A new look at patent quality: Relating patent prosecution to validity. *Journal of Empirical Legal Studies*, 9(1):1–32.

Paul McCann. 2020. fugashi, a tool for tokenizing Japanese in python. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 44–51, Online. Association for Computational Linguistics.

Masaki Murata and Hitoshi Isahara. 2002. Using the diff command in patent documents. In *Third NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Akihiro Shinmori, Manabu Okumura, and Y Marukawa. 2004. Aligning patent claims with detailed descriptions for readability. In *Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization*.

Brian Thompson and Philipp Koehn. 2019. Vecalign: Improved sentence alignment in linear time and space. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Hayato Tsukagoshi and Ryohei Sasano. 2024. Ruri: Japanese general text embeddings.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024a. Text embeddings by weakly-supervised contrastive pre-training.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024b. Multilingual e5 text embeddings: A technical report.

10. Language Resource References

Makoto Iwayama, Atsushi Fujii, Noriko Kando, and Akihiko Takano. 2003. Overview of patent retrieval task at NTCIR-3. In *Proceedings of the ACL-2003 Workshop on Patent Corpus Processing*, pages 24–32, ,. Association for Computational Linguistics.

Shaobo Li, Jie Hu, Yuxin Cui, and Jianjun Hu. 2018. Deeppatent: patent classification with convolutional neural networks and word embedding. *Scientometrics*, 117(2):721–744.

Florina Piroi, Mihai Lupu, Allan Hanbury, and Veronika Zenz. 2011. Clef-ip 2011: Retrieval in the intellectual property domain. In *CLEF (notebook papers/labs/workshop)*.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Mirac Suzgun, Luke Melas-Kyriazi, Suproteem K Sarkar, Scott Kominers, and Stuart Shieber. 2023. The harvard USPTO patent dataset: A large-scale, well-structured, and multi-purpose corpus of patent applications. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

A. Original Japanese Prompt

The original Japanese prompt used in the LLM-based model can be seen as follows:

Prompt Template

```
あなたは特許明細書のサポート要件チェックに詳しいアナリストです。以下の請求項と明細書の段落について、サポート要件を満たす部分（スパン）を抽出し、その説明の種類と確信度を判定してください。

# 抽出項目
1. 対応スパン（請求項のどの語句・文言が、明細書にサポートされているか）
2. 説明の種類（itself/description/example）
  - itself: 請求項の文言そのものがそのまま書かれている部分
  - description: クレームの文言の定義や詳細説明が書かれている部分
  - example: クレームの文言に対応する実施例やバリエーションが書かれている部分
3. 確信度（high/medium/low）

* 同じ請求項内に複数箇所が該当する場合は、全て列挙してください。
* 該当箇所がなければ、空のリストを返してください。

# 出力フォーマット (JSON)
[{
  "claim_span": "<請求項内の語句・文言>",
  "type": "itself | description | example",
  "confidence": "high | medium | low"
},…]

請求項: {請求項}
明細書: {明細書段落}
```

B. LLM Training Settings

The hyperparameters for fine-tuning Qwen3-14B are detailed in Table 7.

LoRA Rank	64
LoRA Alpha	64
LoRA Dropout	0
LoRA Target	MLP & Attention layers
Optimizer	AdamW (Loshchilov and Hutter, 2019)
Learning Rate	2e-5
Learning Rate Scheduler	Linear
Warmup Ratio	0.1
Weight Decay	0.01
Epochs	2
Batch Size	8

Table 7: List of hyperparameters for fine-tuning LLM