

Multi-Label Delegation Target Identification in Japanese Law

Tsuyoshi Fujita, Yuya Sawada, Yusuke Sakai, Taro Watanabe

Nara Institute of Science and Technology (NAIST)

fujita.tsuyoshi.fy4@naist.ac.jp

{sawada.yuya.sr7, sakai.yusuke.sr9, taro}@is.naist.jp

Abstract

Legal systems often employ legislative drafting practices in which higher-level laws delegate detailed regulations to lower-level laws. In particular, when a single provision delegates authority to multiple lower-level laws, readers must cross-reference many related provisions scattered across the entire bodies of law, imposing a substantial cognitive burden. Reflecting this real-world problem, this work proposes a novel *multi-label delegation target identification* task to identify all the delegated provisions corresponding to an input provision, with the focus on Japanese law. We construct pipeline systems combining a candidate retriever and a candidate classifier, using language models with diverse architectures. Experimental results show that adopting a cross-encoder-based binary classifier as a candidate classifier improves performance at finer granularities, establishing a solid baseline for the task. Furthermore, our comprehensive analyses reveal the remaining challenges in achieving sufficient performance on instances with many delegated provisions and identify typical error types, providing a foundation for future improvement.

1 Introduction

Many legal systems rely on *delegation*, whereby higher-level laws authorize the lower-level laws to implement detailed provisions, as illustrated in Figure 1 (Yoshida, 2012; Whittington and Iuliano, 2017; Sim et al., 2024; Del Monte and Mañko, 2021). While identifying delegation is essential for understanding legal systems, reading laws that include delegation imposes a substantial burden on readers, who must repeatedly consult the delegated laws. Consequently, legal information providers have long developed support systems that visualize delegation for commercial and governmental use. However, effective deployment of such systems requires exhaustive annotation of delegation across

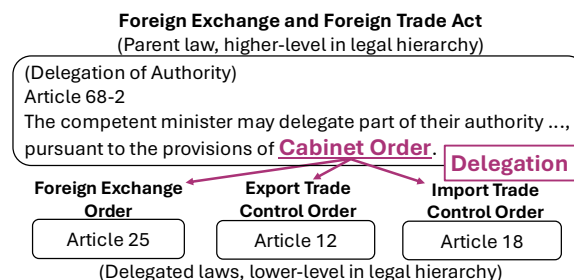


Figure 1: Example of delegation in Japanese laws. The original Japanese text is provided in Appendix E.

large and frequently updated bodies of law, making manual annotation time-consuming.

Delegation may have a one-to-one correspondence between a delegating provision (henceforth *delegation source provision*) and a delegated provision (henceforth *delegation target provision*), but there are also cases where a single delegation source provision corresponds to multiple delegation target provisions. Figure 1 illustrates the latter case in Japanese law, where Article 68-2 of the Foreign Exchange and Foreign Trade Act, a higher-level law, stipulates the delegation of authority, “pursuant to the provisions of Cabinet Order,” and the details of the authority to be delegated are specified in the three provisions of the lower-level Cabinet Orders, i.e., Article 25 of the Foreign Exchange Order, Article 12 of the Export Trade Control Order, and Article 18 of Import Trade Control Order. When multiple delegation targets exist, annotating delegation requires cross-referencing a larger number of relevant provisions scattered across the entire body of law, substantially increasing the annotation burden compared to the cases with a single delegation target.

To automate delegation extraction in Japanese law, Fujita et al. (2026) proposed a pipeline system combining a named entity recognition model for extracting keywords indicating delegation (Yamada et al., 2020; Devlin et al., 2019) with a similarity-

based entity disambiguation model for identifying the delegation targets indicated by each keyword (i.e., delegation target identification) (Orlando et al., 2024; Nakatani et al., 2025; Gillick et al., 2018; Wu et al., 2020). They reported an F1 score exceeding 95 for extracting the keywords and a Recall@10 exceeding 90 for identifying the delegation targets, achieving sufficient accuracy for supporting manual annotation. However, their experiments were conducted under a *single-label* setting, in which identifying any one of the delegation targets is regarded as a correct prediction, even when multiple targets exist. While this simplification is effective under restricted conditions, it is insufficient for practical applications. As illustrated in Figure 1, delegation can involve multiple target provisions, with the single-label setting representing a special case. Therefore, realistic legal understanding and downstream applications require a *multi-label* setting that identifies all relevant delegation targets.

In this paper, we introduce a novel **multi-label delegation target identification** task, which explicitly addresses scenarios where a single provision delegates authority to multiple targets, with a focus on Japanese law. We adopt a pipeline that combines (i) a *candidate retriever* that narrows down delegation target candidates by computing embedding-based similarity between delegation keywords and candidate provisions, considering multiple possible targets, and (ii) a *candidate classifier* that predicts true delegation targets from among the candidates fetched by the candidate retriever. This pipeline can be incorporated into an annotation support tool that provides human annotators with a ranked list of candidate provisions for verification. The list is generated by the candidate retriever, with candidates predicted as valid by the candidate classifier placed at the top. Experimental results using language models with diverse architectures show that this pipeline can improve performance in identifying all delegation targets at finer granularities, such as paragraph and item-levels, compared to using the candidate retriever alone, establishing a solid baseline for future research of the multi-label delegation target identification task.

To summarize, our contributions are as follows:

- We propose the novel and practical multi-label delegation target identification task in Japanese law. It moves beyond the previous single-label simplification and more accurately reflects the real-world complexity of legal delegation.

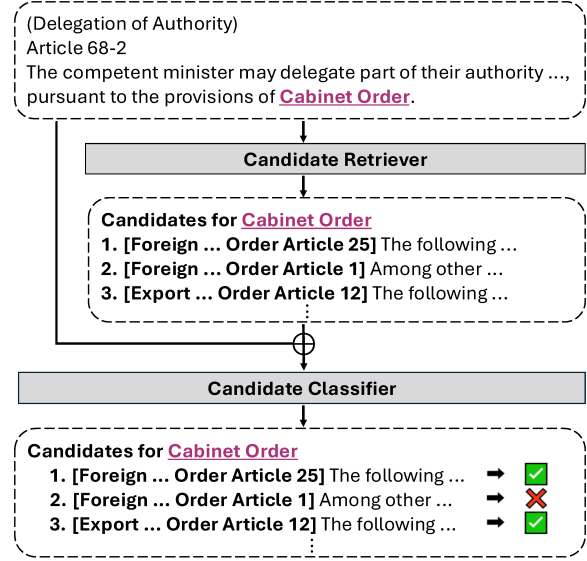


Figure 2: Data processing flow of multi-label delegation target identification

- We construct straightforward and interpretable pipeline systems combining a candidate retriever with a candidate classifier, using language models with diverse architectures. Experimental results show that using a cross-encoder-based binary classifier as a candidate classifier improves performance at finer granularities. This finding leads to a practical utility and establishes a solid baseline for the task.
- We conduct comprehensive analyses of the model behavior. Quantitative analysis reveals the remaining challenges to achieve sufficient performance on instances with a larger number of gold delegation target provisions, and qualitative analysis identifies typical error types, providing a foundation for future improvement.

2 Method

2.1 Task Definition: Multi-Label Delegation Target Identification

The multi-label delegation target identification task takes as input a provision from a higher-level law and the position of a delegation keyword within it, and aims to identify all delegation target provisions indicated by the keyword¹. For example, in Figure 1, given Article 68-2 of the Foreign Exchange and Foreign Trade Act and the delegation keyword “Cabinet Order,” the task is to identify Article 25 of the Foreign Exchange Order, Article

¹Appendix 6.1 compares the multi-label delegation target identification task with prior work.

12 of the Export Trade Control Order, and Article 18 of the Import Trade Control Order as the delegation target provisions. As shown in Figure 2, we address this task using a pipeline that combines a candidate retriever for narrowing down delegation target candidates with a candidate classifier that predicts true delegation targets from among the candidates fetched by the candidate retriever.

2.2 Candidate Retriever

We extend the candidate retriever of Fujita et al. (2026) to support the multi-label setting. Their candidate retriever embeds an input provision x , with the positional information of a delegation keyword, and a delegation target candidate y into vector representations $\mathbf{h}_x$ and $\mathbf{h}_y$ using a text embedding model (Wang et al., 2024b,c). The inner product $s(x, y) = \mathbf{h}_x^\top \mathbf{h}_y$ serves as the similarity score between x and y , and provisions with high similarity scores are retrieved. The retriever is trained with the following loss function that maximizes similarity to one gold delegation target:

$$\mathcal{L}_{\text{single}} = -\log \left(\frac{\exp(s(x, y^+))}{\sum_{y' \in \mathcal{Y}} \exp(s(x, y'))} \right). \quad (1)$$

Here, y^+ denotes a gold delegation target provision. When multiple gold targets exist, one is selected as y^+ . y' denotes a candidate provision in the candidate set $\mathcal{Y}$ ($y' \in \mathcal{Y}$), where $\mathcal{Y}$ comprises a set of candidate provisions within a mini-batch $\mathcal{Y}_C$ and a set of hard-negative samples $\mathcal{Y}_{\text{hard}}$ for y^+ ($\mathcal{Y} = \mathcal{Y}_C \cup \mathcal{Y}_{\text{hard}}$) (Wu et al., 2020).

To adapt to multiple targets, we introduce two multi-label losses and compare their performance:

$$\begin{aligned} \mathcal{L}_{\text{multi 1}} &= -\log \left(\frac{\sum_{y^+ \in \mathcal{Y}_P} \exp(s(x, y^+))}{\sum_{y' \in \mathcal{Y}_P \cup \mathcal{Y}_N} \exp(s(x, y'))} \right), \\ \mathcal{L}_{\text{multi 2}} &= -\sum_{y^+ \in \mathcal{Y}_P} \log \left(\frac{\exp(s(x, y^+))}{\sum_{y' \in \mathcal{Y}_N \cup \{y^+\}} \exp(s(x, y'))} \right), \end{aligned} \quad (2)$$

where $\mathcal{Y}_P$ and $\mathcal{Y}_N$ denote the sets of positive and negative candidates, respectively, within the candidate set $\mathcal{Y} = \mathcal{Y}_P \cup \mathcal{Y}_N$. $\mathcal{L}_{\text{multi 1}}$ extends the single-label loss $\mathcal{L}_{\text{single}}$ by maximizing the sum of all gold targets' scores, while $\mathcal{L}_{\text{multi 2}}$, following Zhang et al. (2022), treats each gold target independently against the negatives. Experimenting with all three losses enables the multifaceted comparison between single-label and multi-label training.

2.3 Candidate Classifier

We implement three types of candidate classifiers to reflect representative approaches commonly adopted in relevant fields such as entity linking and multi-label classification: (1) *label-embedding classifier*, (2) *binary classifier*, and (3) *LLM classifier*. This allows us to examine the effectiveness of these widely used approaches as baselines for our task.

Label-Embedding Classifier Following Févry et al. (2020), we employ a label-embedding classifier that scores delegation target candidates by computing similarity between the representation of the delegation keyword and learned label embeddings. Concretely, the model comprises a transformer-based language model (E_{emb}) followed by a fully-connected layer ($\text{Linear}_{\text{emb}}$). The probability that a candidate provision y is a correct delegation target of x is defined as:

$$P_{\text{emb}}(y | x) = \sigma(z_{\text{emb}}), \quad (4)$$

$$z_{\text{emb}} = \mathbf{u}_x^\top \mathbf{e}_y + b_y, \quad (5)$$

$$\mathbf{u}_x = \text{Linear}_{\text{emb}}(\text{cls}(E_{\text{emb}}(\tau_x))). \quad (6)$$

Here, $\mathbf{e}_y$ and b_y denote the embedding and bias of candidate y , respectively. The function $\text{cls}(\cdot)$ extracts the representation of [CLS] token from encoder output, and $\sigma(\cdot)$ is the sigmoid function. τ_x is the representation of a delegation keyword and its context, constructed as in Wu et al. (2020). The model is trained with binary cross-entropy loss:

$$\begin{aligned} \mathcal{L}_{\text{BCE}} &= -[\mathbb{I}(y) \log P_{\text{emb}}(y | x) \\ &\quad + (1 - \mathbb{I}(y)) \log(1 - P_{\text{emb}}(y | x))], \end{aligned} \quad (7)$$

where $\mathbb{I}(y) = 1$ if y is a gold delegation target and 0 otherwise. At inference, candidates with $P_{\text{emb}}(y | x)$ above a predefined threshold are predicted as delegation targets of x .

Binary Classifier The binary classifier adapts a cross-encoder (Wu et al., 2020) to a binary classification task to determine whether delegation exists between a delegation source and a delegation target candidate. The model comprises a transformer-based language model (E_{bin}) followed by a fully-connected layer ($\text{Linear}_{\text{bin}}$), as in the label-embedding classifier. The probability that a candidate provision y is a correct delegation target of a delegation keyword x is computed as:

$$P_{\text{bin}}(y | x) = \sigma(z_{\text{bin}}), \quad (8)$$

$$z_{\text{bin}} = \text{Linear}_{\text{bin}}(\text{cls}(E_{\text{bin}}(\tau_{x,y}))). \quad (9)$$

# of Delegation Targets	1	2–10	11–100	101–
# of Delegation Keywords	17,890	2,508	320	5
Ratio(%)	86.3%	12.1%	1.54%	0.02%

Table 1: Number of gold delegation target provisions

Here, $\tau_{x,y}$ is a token sequence concatenation of x and y as in Wu et al. (2020). The model is trained with binary cross-entropy loss, defined by replacing $P_{\text{emb}}(y | x)$ in Equation 7 with $P_{\text{bin}}(y | x)$. At inference, candidates with $P_{\text{bin}}(y | x)$ above a predefined threshold are predicted as delegation targets of x .

LLM Classifier The LLM classifier formulates candidate classification as a multi-label multiple-choice task using a chat-based large language model (LLM) to efficiently conduct inference, inspired by the zero-shot prompting paradigm of ChatEL (Ding et al., 2024). Given an input provision x and a candidate set $\mathcal{C}$, the model is prompted to select all correct delegation targets corresponding to x from a list of candidates. To handle large candidate sets, we partition $\mathcal{C}$ into subsets of candidates per prompt. Each prompt presents candidates as enumerated choices, and the model is instructed to output the indices of all correct options in a comma-separated format, allowing multiple selections. Predictions from all subsets are then aggregated to produce the final output. The prompt templates for the LLM classifier are shown in Appendix A.

3 Experimental Setup

Dataset We use the *provision database* of Japanese laws, and the *delegation extraction dataset* with manually annotated delegation keywords and targets, both constructed by Fujita et al. (2026). Fujita et al. (2026) adopt a relaxed single-label setting that treats predicting any one delegation target as correct. In contrast, we introduce a multi-label setting that requires identifying all targets, better reflecting practical delegation annotation. Table 1 shows the distribution of gold delegation targets per keyword in the delegation extraction dataset: about 14% of the keywords have multiple targets, mostly between 2 and 10. Although relatively infrequent, these cases impose a particularly heavy burden in the manual delegation annotation (see Section 1). Therefore, handling cases with multiple delegation targets is an important practical issue. In the multi-label delegation target identification, all provisions in the database

are treated as candidates, and the task is to predict all gold targets for each delegation keyword. Following Fujita et al. (2026), we evaluate the performance of the candidate retrievers and classifiers using the delegation extraction dataset at four granularities: entire law, article, paragraph, and item. Evaluation is conducted via five-fold cross-validation, which uses 20% of the sentences as evaluation data, and further splits the remaining 80% into training and development sets at a ratio of 9:1. Further details on the dataset, model training, and evaluation are provided in Appendix B.

Candidate Retriever We use multilingual E5 (Wang et al., 2024c) as the encoder and compare the performance of three candidate retrievers trained with different losses: Ret-Single, trained with $\mathcal{L}_{\text{single}}$; Ret-Multi1, trained with $\mathcal{L}_{\text{multi1}}$; and Ret-Multi2, trained with $\mathcal{L}_{\text{multi2}}$. Since most delegation keywords have at most 10 targets (Table 1), each retriever outputs the top 30 candidates at inference, which is sufficient to cover the majority of the gold targets. Performance is evaluated using Recall@30 (R@30), measuring the proportion of gold targets among the top-ranked candidates.

Candidate Classifier We compare three types of candidate classifiers: *label-embedding classifier*, *binary classifier*, and *LLM classifier*. For the **label-embedding classifier**, we use BERT (Devlin et al., 2019) and LUKE (Yamada et al., 2020) as encoders and finetune them for the task, denoted Emb_{bert} and Emb_{luke} , respectively. For the **binary classifier**, we finetune BERT in the same manner, denoted as Bin. For the **LLM classifier**, we use some of the latest LLMs, namely Qwen3 (Yang et al., 2025), Qwen3 Swallow², gpt-oss (OpenAI et al., 2025), and gpt-oss Swallow³, and prompt them to select the correct delegation targets from candidate provisions. These models are denoted as LLM_{qwen} , $\text{LLM}_{\text{qwen-s}}$, LLM_{gpt} , and $\text{LLM}_{\text{gpt-s}}$, respectively. For each method and base model, we test three variants depending on the candidate set used as input. The *-Single variant uses candidates retrieved by Ret-Single, the *-Multi1 variant uses candidates from Ret-Multi1, and the *-Multi2 variant uses candidates from Ret-Multi2. For all variants, we use the top 30 retrieved candidates. For example, $\text{Emb}_{\text{bert}}\text{-Multi1}$ denotes the label-embedding classifier based on BERT that takes as

²<https://swallow-llm.github.io/qwen3-swallow.en.html>

³<https://swallow-llm.github.io/gptoss-swallow.en.html>

Granularity	Ret-Single	Ret-Multi1	Ret-Multi2
Law	95.2 (± 0.91)	91.8 (± 1.13)	91.8 (± 1.00)
Article	72.2 (± 2.16)	63.6 (± 2.26)	65.4 (± 1.31)
Paragraph	69.5 (± 2.43)	60.0 (± 2.28)	62.0 (± 1.13)
Item	68.5 (± 2.58)	59.3 (± 2.33)	61.1 (± 1.24)

Table 2: R@30 of candidate retrievers

input the top 30 candidates retrieved by Ret-Multi1.

Pipeline System We evaluate pipeline configurations combining each retriever with a corresponding classifier variant. Each pipeline is denoted as Pipe-[Classifier]-[Retriever], where [Classifier] indicates the candidate classifier type and base model (e.g., Emb_{bert}), and [Retriever] is one of Single, Multi1, or Multi2, indicating the candidate retriever variant. For example, Pipe-Emb_{bert}-Single denotes the pipeline comprising Ret-Single and Emb_{bert}-Single. Additionally, we include Naive-Top1, which directly outputs the highest-scoring candidate retrieved by Ret-Single as the predicted delegation target. Each pipeline is evaluated using precision, recall, and F1 based on the predicted versus gold delegation targets.

4 Results and Discussions

4.1 Candidate Retriever

Table 2 reports R@30 of the candidate retrievers. Ret-Single, trained with the single-label loss $\mathcal{L}_{\text{single}}$, outperforms Ret-Multi1 and Ret-Multi2, trained with the multi-label losses, across all granularities. This may stem from two factors.

First, high semantic similarity among gold targets: all gold delegation targets corresponding to a given delegation source share the common property of specifying detailed matters related to that delegation source, and thus often exhibit high semantic similarity to one another. Consequently, even when only one gold provision among multiple gold targets is used for training under $\mathcal{L}_{\text{single}}$, the model may still learn to assign high similarity scores to all gold delegation targets, including those not used during training.

Second, less discriminative representation learning in multi-label training: under $\mathcal{L}_{\text{multi1}}$, the probability assigned to positives is optimized jointly, which can reduce the relative training signal for each gold target compared with $\mathcal{L}_{\text{single}}$. On the other hand, $\mathcal{L}_{\text{multi2}}$ independently optimizes multiple positive pairs for the same source provision. Since the gold targets are often semantically simi-

Model		Granularity	Precision	Recall	F1
Naive-Top1		Law	92.6 ± 0.89	85.3 ± 1.15	88.8 ± 0.93
		Article	78.9 ± 1.84	47.3 ± 2.52	59.1 ± 2.35
		Paragraph	56.3 ± 6.64	33.3 ± 3.40	41.8 ± 4.45
		Item	56.0 ± 6.70	33.1 ± 3.44	41.6 ± 4.50
Emb _{luke}	-Single	Law	44.4 ± 2.64	75.4 ± 1.09	55.8 ± 1.83
		Article	15.0 ± 0.92	31.2 ± 1.51	20.2 ± 0.57
		Paragraph	7.35 ± 0.69	17.7 ± 0.72	10.4 ± 0.67
		Item	6.29 ± 0.65	16.9 ± 0.68	9.14 ± 0.68
	-Multi1	Law	30.4 ± 1.11	78.7 ± 0.47	43.8 ± 1.16
		Article	8.91 ± 0.36	30.3 ± 1.57	13.8 ± 0.53
		Paragraph	4.02 ± 0.14	15.8 ± 0.75	6.40 ± 0.22
		Item	3.25 ± 0.16	14.5 ± 0.75	5.32 ± 0.25
	-Multi2	Law	29.6 ± 1.44	76.7 ± 0.71	42.7 ± 1.50
		Article	8.36 ± 0.37	27.5 ± 1.03	12.8 ± 0.51
		Paragraph	3.86 ± 0.17	14.5 ± 1.00	6.10 ± 0.29
		Item	3.16 ± 0.13	13.5 ± 0.96	5.12 ± 0.23
Bin	-Single	Law	83.1 ± 3.58	78.1 ± 4.57	80.5 ± 3.97
		Article	64.3 ± 9.67	53.2 ± 4.70	57.9 ± 5.66
		Paragraph	56.2 ± 10.5	48.9 ± 5.61	51.8 ± 6.61
		Item	56.2 ± 10.5	48.9 ± 5.61	51.7 ± 6.62
	-Multi1	Law	87.0 ± 4.06	72.9 ± 8.32	79.2 ± 6.58
		Article	71.6 ± 6.27	48.6 ± 8.05	57.8 ± 7.77
		Paragraph	60.8 ± 10.2	45.3 ± 8.09	51.9 ± 9.00
		Item	60.7 ± 10.2	45.3 ± 8.09	51.9 ± 9.01
	-Multi2	Law	85.4 ± 7.00	75.8 ± 4.71	80.0 ± 3.72
		Article	70.1 ± 9.54	51.1 ± 5.01	58.6 ± 5.10
		Paragraph	62.4 ± 8.76	47.6 ± 4.65	53.6 ± 4.76
		Item	62.4 ± 8.76	47.6 ± 4.65	53.6 ± 4.76
LLM _{qwen-s}	-Single	Law	65.1 ± 0.83	90.0 ± 0.86	75.5 ± 0.48
		Article	31.6 ± 0.62	57.9 ± 1.55	40.9 ± 0.41
		Paragraph	19.3 ± 0.74	46.6 ± 1.31	27.3 ± 0.80
		Item	15.6 ± 0.67	45.8 ± 1.41	23.2 ± 0.80
	-Multi1	Law	60.5 ± 0.62	85.7 ± 1.12	70.9 ± 0.15
		Article	30.3 ± 0.44	50.8 ± 2.08	38.0 ± 0.56
		Paragraph	18.5 ± 0.74	41.7 ± 2.06	25.6 ± 1.02
		Item	15.7 ± 0.73	41.1 ± 2.08	22.7 ± 1.00
	-Multi2	Law	62.8 ± 1.02	85.9 ± 0.91	72.5 ± 0.77
		Article	30.7 ± 0.21	51.9 ± 1.57	38.6 ± 0.31
		Paragraph	17.9 ± 0.39	41.7 ± 1.40	25.0 ± 0.51
		Item	14.9 ± 0.38	41.0 ± 1.47	21.9 ± 0.50

Table 3: Performance of the pipeline systems and Naive-Top1. In the tables of this paper, “Pipe-” is omitted from the notation of the model names, and bold scores indicate the best score for each metric and granularity.

lar, this may encourage the source representation to broadly align with a cluster of related provisions rather than with a specific gold target. As a result, semantically related non-gold provisions may also receive higher similarity scores, weakening the separation of gold and non-gold provisions.

4.2 Effectiveness of Binary Classifier

Table 3 shows the main experimental results of the pipeline systems. In addition to Naive-Top1, we show the performance of the pipeline systems using the best candidate classifiers in terms of the

item-level F1 scores of their base models for each type of candidate classifiers; Pipe-Emb_{luke}-* for the label-embedding classifier, Pipe-Bin-* for the binary classifier, and Pipe-LLM_{qwen-s}-* for the LLM classifier⁴. Pipe-Bin-* outperform Naive-Top1 at the paragraph and item levels in F1 score, while Pipe-Emb_{luke}-* and Pipe-LLM_{qwen-s}-* underperform Naive-Top1 and Pipe-Bin-*. This indicates that the **binary classifier enhances the ability of the system** to identify multiple delegation targets at finer granularities, while the introduction of the **label-embedding classifier and the LLM classifier degrades performance** compared to using the candidate retriever alone.

The observed performance differences can be explained by how each classifier models the relationship between a delegation source and its candidate provisions. Compared to the LLM classifier, the binary classifier enables more reliable fine-grained discrimination among the candidates. By jointly encoding source-candidate pairs with a cross-encoder, it captures subtle lexical and structural cues that determine delegation. In contrast, the LLM classifier relies more on global semantic understanding, which may be less effective, especially when candidates share overlapping content with only slight differences. A similar pattern holds when comparing the binary classifier with the label-embedding classifier. The latter scores candidates based on independently computed representations of the delegation source and candidate provisions, limiting its ability to model context-dependent interactions.

4.3 Pipeline Behavior

Focusing on the comparison of Pipe-Bin-Single, Pipe-Bin-Multi1, and Pipe-Bin-Multi2, Table 3 shows that Pipe-Bin-Single achieves higher recall than Pipe-Bin-Multi1 and Pipe-Bin-Multi2, primarily due to the higher coverage of Ret-Single, which is reflected in its superior R@30 (Table 2). In contrast, the precision of Pipe-Bin-Single is lower than that of Pipe-Bin-Multi1 and Pipe-Bin-Multi2, resulting in comparable F1 scores across all models. This is primarily due to the fact that Ret-Single is trained using only one provision among multiple gold targets as positive and retrieves candidates with high similarity to that single provision, which

⁴Appendix C.2 shows the complete results of all the pipeline systems. The choice of base models for the label-embedding classifier has only a limited impact on overall performance, while using Japanese LLMs for the LLM classifier contributes to a performance improvement.

Model	Granularity	Precision	Recall	F1
Naive-Top1	Law	81.5 $\pm$ 5.51	22.4 $\pm$ 2.40	35.1 $\pm$ 3.34
	Article	62.8 $\pm$ 3.67	10.3 $\pm$ 1.00	17.8 $\pm$ 1.55
	Paragraph	38.6 $\pm$ 5.06	6.41 $\pm$ 0.94	11.0 $\pm$ 1.55
	Item	37.4 $\pm$ 5.35	6.22 $\pm$ 0.99	10.7 $\pm$ 1.64
Bin	-Single	Law	78.4 $\pm$ 1.99	46.4 $\pm$ 4.59
		Article	64.7 $\pm$ 8.66	25.2 $\pm$ 4.59
		Paragraph	45.7 $\pm$ 9.86	20.0 $\pm$ 5.69
		Item	45.6 $\pm$ 9.89	20.0 $\pm$ 5.68
	-Multi1	Law	77.4 $\pm$ 2.40	35.0 $\pm$ 9.22
		Article	68.6 $\pm$ 2.98	19.5 $\pm$ 5.36
		Paragraph	47.0 $\pm$ 5.65	16.1 $\pm$ 5.04
		Item	47.0 $\pm$ 5.70	16.1 $\pm$ 5.05
	-Multi2	Law	77.1 $\pm$ 7.58	39.2 $\pm$ 5.42
		Article	67.6 $\pm$ 8.80	22.0 $\pm$ 5.57
		Paragraph	49.1 $\pm$ 8.61	17.8 $\pm$ 5.15
		Item	49.1 $\pm$ 8.61	17.8 $\pm$ 5.16

Table 4: Performance of the pipeline systems evaluated on data with two or more gold delegation targets.

leads to the retrieval of difficult negative candidates that are semantically similar to that specific gold provision. Consequently, Bin-Single may fail to handle these harder-to-distinguish candidates, resulting in increased false positives and a lower precision observed in Pipe-Bin-Single. Overall, these results indicate that **improving the discriminative capacity of the binary classifier, especially for the harder-to-distinguish negative candidates, is crucial** for converting higher retriever coverage into precision gains and better overall performance.

5 Analysis

We conduct additional analyses to better understand the behaviors of the pipeline systems using the binary classifier, i.e., Pipe-Bin-*, which achieves the best performance at finer granularities. We provide further analyses and case studies in Appendices C and D.

5.1 Evaluation Restricted to Multi-Label Instances

To further examine the effectiveness of the binary classifier, we evaluate Pipe-Bin-* exclusively on the instances with two or more delegation target provisions⁵, which comprise around 14% of the dataset (see Table 1). Although evaluation on the full dataset, including both single-label and multi-label instances, more closely reflects practical delegation annotation scenarios, detailed analysis restricted to multi-label instances is also important,

⁵See Appendix C.2 for the evaluation results of the other pipeline systems under this setting.

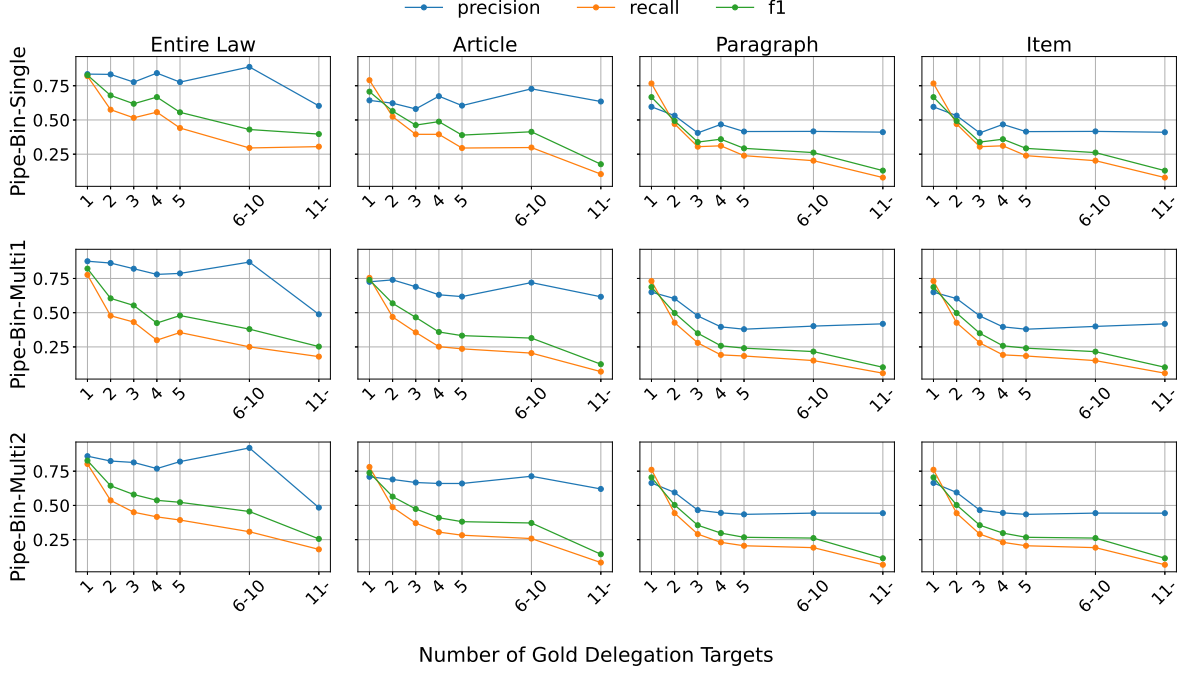


Figure 3: Precision, recall, and F1 scores grouped by the number of gold delegation targets. The rows represent the model variants, while the columns represent the evaluation granularities.

as these cases are inherently more difficult and impose a higher annotation burden.

Table 4 reports the results under this setting. Naive-Top1 exhibits extremely low recall across all granularities, reflecting the fundamental mismatch between single-target prediction and multi-label requirements. In contrast, **all the pipeline systems that use the binary classifier substantially outperform Naive-Top1**, confirming the effectiveness of the binary classifier in handling multiple delegation targets. Crucially, these gains on long-tail cases with two or more delegation targets are particularly important: such cases impose a disproportionately heavy burden on annotators, as they require extensive cross-referencing across multiple provisions, and improving performance on them can therefore substantially reduce the overall annotation workload.

5.2 Number of Gold Delegation Targets and Model Performance

To further analyze how the number of gold delegation targets affects the performance of the proposed pipeline systems under the multi-label setting, we group delegation keywords by the number of their gold delegation targets and evaluate precision, recall, and F1-score separately for each group. Specifically, we report results for keywords associated with 1 to 5 gold targets, as well as aggregated

groups of 6 to 10 targets and 11 or more targets.

Figure 3 shows the performance of Pipe-Bin-Single, Pipe-Bin-Multi1, and Pipe-Bin-Multi2 at different granularities. Across all models and granularities, the performance consistently declines as the number of gold delegation targets increases, indicating the challenging nature of the multi-label delegation target identification task. The decline is particularly pronounced for recall, indicating that identifying all valid delegation targets becomes increasingly difficult as their number grows. In contrast, precision is relatively more stable, suggesting that false positives are less sensitive to the number of gold targets, whereas false negatives grow rapidly as the target set expands. The other pipeline systems also exhibit similar degradation of the overall performance, as shown in Appendix C.3.

5.3 Error Taxonomy

To better understand the remaining challenges, we conduct a qualitative error analysis on cases where Pipe-Bin-Single, Pipe-Bin-Multi1, and Pipe-Bin-Multi2 all made incorrect predictions. Rather than treating all errors uniformly, we categorize errors by their underlying causes, with the goal of identifying systematic failure patterns to guide future improvements. Our analysis reveals two primary error types: (1) *granularity mismatch*, and (2) *confusion due to semantic similarity* between candidates.

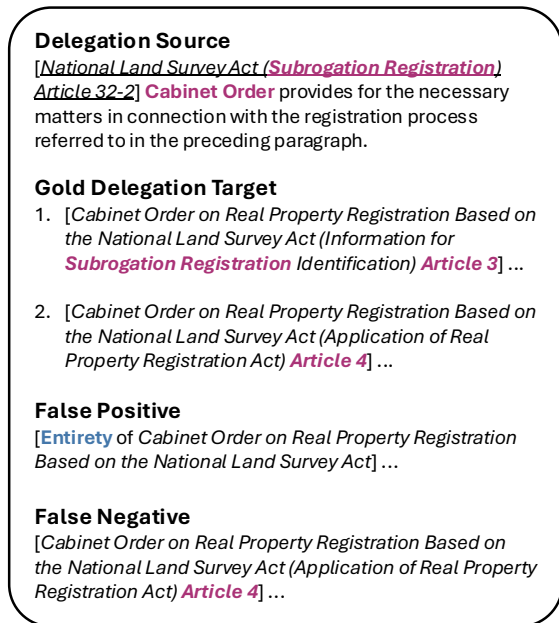


Figure 4: Example of granularity mismatch error. “False Positive” and “False Negative” in this Figure, as well as Figures 5, are the false positives and false negatives that were common for all of Pipe-Bin-Single, Pipe-Bin-Multi1, and Pipe-Bin-Multi2.

Granularity Mismatch Granularity mismatch arises when the model fails to distinguish between provisions at different hierarchical levels. Figure 4 illustrates this issue. Here, the delegation keyword is “Cabinet Order,” which delegates matters regarding the processes of subrogation registration to Articles 3 and 4 of the *Cabinet Order on Real Property Registration Based on the National Land Survey Act*. Importantly, these two articles are the only provisions within the Cabinet Order that regulate the processes of subrogation registration. Despite this narrow scope, the models incorrectly predicted the entire *Cabinet Order* as the delegation target, resulting in a false positive that encompasses many provisions unrelated to subrogation registration. This error highlights the difficulty in precisely identifying the appropriate granularity of delegation targets.

Confusion due to Semantic Similarity Confusion due to semantic similarity commonly occurs between adjacent provisions, as neighboring provisions often regulate closely related topics and therefore appear highly similar in meaning. An example of this confusion is shown in Figure 5, where the correct delegation targets are Articles 9 and 23 of the *Regulations for Enforcement of the Land Transport Industries Regulation Act*. The models incorrectly identified Article 8 as a delegation tar-

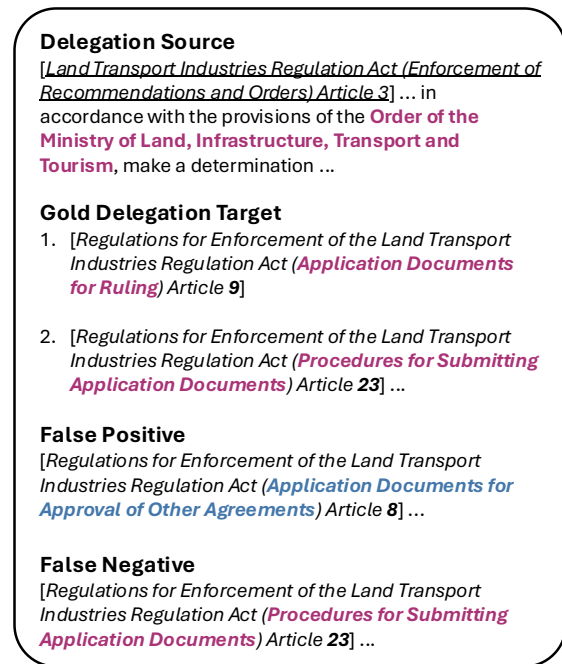


Figure 5: Example of confusion due to semantic similarity.

get instead of Article 23 because Article 8 is more semantically similar to Article 9 than Article 23 is: both Articles 8 and 9 define the contents of application documents and share highly overlapping language, whereas Article 23 addresses a different aspect, namely the procedures of document submission. As a result, the models are biased toward selecting Article 8 due to its strong similarity to Article 9, leading to a false positive while overlooking the correct but less similar Article 23.

6 Related Work

6.1 Delegation Extraction and Statute Retrieval

Fujita et al. (2026) propose a single-label delegation extraction task that identifies delegation keywords in legal texts and predicts one target provision for each keyword. We extend this task to a multi-label setting, where each keyword may correspond to multiple targets. This better reflects real-world statutes and introduces a more challenging precision–recall trade-off, as models must recover all correct targets while limiting false positives.

Additionally, statute retrieval is also closely related to multi-label delegation target identification. In legal NLP, statute retrieval aims to identify provisions relevant to a legal question or problem (Martinez-Gil, 2023), often under a multi-label

formulation. For example, BSARD (Louis and Spanakis, 2022), STARD (Su et al., 2024), and the COLIEE shared task (Goebel et al., 2024) require retrieving multiple relevant provisions for a single query, reflecting that legal reasoning often depends on multiple statutes.

Existing statute-retrieval tasks identify relevant provisions for the input as a whole and do not capture fine-grained correspondences between specific expressions in the input and retrieved provisions. In contrast, multi-label delegation target identification explicitly pinpoints multiple delegation targets grounded in localized expressions within a provision, enabling a more precise and interpretable representation of inter-provision relationships.

6.2 Multi-Label Retrieval

While dense retrievers are commonly trained with single-positive contrastive objectives, which assume one correct target per query, several studies extend contrastive learning to settings with multiple valid targets. In entity disambiguation and entity linking, Zhang et al. (2022) and Wang et al. (2024a) adopt multi-label Noise Contrastive Estimation (NCE) for candidate retrieval to treat multiple entities as correct targets for a single passage. In Section 2.2, we constructed our $\mathcal{L}_{multi2}$ following these works. Additionally, similar ideas have been explored in multi-label text classification. Lin et al. (2023) compare several contrastive losses, while Gupta et al. (2024) introduce a decoupled softmax loss for dual-encoder training in extreme multi-label classification (XMC), enabling retrieval of multiple labels over large label spaces.

6.3 Multi-Label Classification

Multi-Label Label-Embedding Classifier In multi-label classification tasks, prior works often learn dense label embeddings and score them against input representations to produce label probabilities (Wang et al., 2018; Du et al., 2019). Recent XMC methods extend this approach using techniques such as semantic label clustering (Chang et al., 2020) and regularization (Guo et al., 2019). While our label-embedding classifier follows a classic entity linking approach of Févry et al. (2020), it is also a straightforward adaptation of these label-embedding-based multi-label classifiers.

Cross-Encoder in Multi-Label Setting Cross-encoders have been adapted to multi-label prediction settings to refine candidate sets generated by

multi-label retrievers or classifiers. In XMC, Li et al. (2023) employ a cross-encoder to independently score labels retrieved by a bi-encoder, improving precision. Our retriever-classifier pipelines and the cross-encoder-based binary classifier align with this work, as well as the widely used entity linking system proposed by Wu et al. (2020). Additionally, in ultra-fine entity typing (Choi et al., 2018), which aims to predict free-form entity types given a sentence with an entity mention, Jiang et al. (2023) propose a multi-candidate cross-encoder to jointly score the candidate types generated by a multi-label classification model.

Multi-Label LLM Classifier Recent work applies LLMs to multi-label prediction through prompting and instruction-based classification. Peskine et al. (2023) employ an LLM for zero-shot multi-label classification, exploiting natural-language label definitions. In XMC, Zhu and Zamani (2024) propose a two-stage in-context learning framework that first reduces the label space with candidate generation before applying LLM-based prediction. Similarly, Tabatabaei et al. (2025) combine dense retrieval with LLM-based classification for hierarchical multi-label classification (HMC). While our LLM classifier is inspired by ChatEL (Ding et al., 2024), it also follows a similar retriever-classifier formulation, where an LLM predicts labels from a candidate set generated by a preceding multi-label retriever.

7 Conclusion

We proposed a novel multi-label delegation target identification task, which aims to predict all delegation target provisions. The task addresses the practical challenges of delegation annotation, in which annotators must identify multiple delegation targets dispersed across lower-level laws. For this task, we constructed retriever-classifier pipelines using diverse model architectures. Our experiments show that introducing a binary classifier improves pipeline performance at finer granularities. This suggests the practical utility of the pipeline in delegation annotation support scenarios and its effectiveness as a baseline for the task. Moreover, our analyses suggest that future work should focus on improving the model’s ability to distinguish semantically similar provisions and select appropriate granularity to achieve higher performance on instances with many gold delegation targets.

Limitations

Task Extension We focus on delegation as a well-scoped task linking a single source expression to multiple target provisions. Other legal phenomena, such as distributed definitions, where a concept is defined across multiple clauses, and exception clauses, where applicability depends on multiple conditions stated elsewhere, also involve multiple targets. Our task proposal and baselines provide a foundation for extending this work to such relations and, ultimately, toward a unified “multi-grounding” formulation of these multi-label legal phenomena.

Jurisdictional Scope This study focuses on Japanese legislation. Although the multi-label delegation target identification task is not unique to Japan, legislative drafting practices vary across jurisdictions (Whittington and Iuliano, 2017; Sim et al., 2024; Del Monte and Mañko, 2021). However, by formalizing the multi-label delegation target identification task and empirically analyzing its core technical challenges, this work provides a conceptual and methodological foundation that can be adapted to other legal systems.

Candidate Retriever Improvement While the candidate retrievers achieve over 90 points of R@30 at the law-level evaluation, performance at finer granularities leaves room for improvement. Future directions to enhance the candidate retrievers include exploiting structural information through graph-based representations of legal references (Yang et al., 2021). However, the primary focus of this work is to propose a novel and practical multi-label delegation target identification task, to establish a solid baseline, and to conduct comprehensive analyses to guide future research. Therefore, experimenting with straightforward methods is sufficient to achieve these goals, and exploring various model architectures is left for future work.

Binary Classifier Improvement Despite the improvements of the proposed approach, F1 scores of 50–60 at finer granularities indicate that multi-label delegation target identification remains challenging. Nevertheless, our experiments establish the candidate retriever and binary classifier pipeline as a solid baseline and show that, once retriever recall is sufficient, the classifier becomes the primary determinant of overall performance. Therefore, future work could improve the classifier through structure-aware representations of hierarchical relationships between provisions (Louis et al., 2023) or

joint modeling of candidate sets to capture dependencies among valid targets (Wang et al., 2024a; Barba et al., 2022). However, as this work focuses on formulating the task, establishing a strong baseline, and analyzing model behavior, exploring these extensions is beyond the scope of this paper.

LLM classifier The LLM classifier follows ChatEL (Ding et al., 2024), using recent LLMs with zero-shot multiple-choice prompting to provide a straightforward baseline, which is the primary goal of this study. Exploring alternative prompting paradigms, supervised training strategies, or different model types is beyond the scope of this work. Due to computational resource constraints, we adopt an efficient multiple-choice prompting framework rather than more complex prompting methods or fine-tuning approaches, which require substantially greater resources. In addition, our agreement with the data providers prohibits the use of API-based closed models.

Ethical Considerations

Licenses We used the dataset created by Fujita et al. (2026) based on the proprietary annotation data of DAI-ICHI HOKI CO., LTD, with the agreement that we will not transfer the data to third parties. Our experimental results could be reproduced by obtaining permission to use this dataset from DAI-ICHI HOKI CO., LTD. The translations of legal texts used in this paper are prepared by the authors with reference to the Japanese Law Translation Database System⁶. DAI-ICHI HOKI CO., LTD. permitted the authors to use the dataset, and the Japanese Law Translation Database System provides data under terms of use compatible with the Creative Commons Attribution 4.0 International License (CC BY), which permits the use of the data in this study.

Harmful Content The data used in this study are texts of Japanese laws and manual annotations of delegation on them, both of which are free of harmful content.

AI-assistant Usage Portions of the manuscript, such as wording adjustments, were drafted with the assistance of GPT-5.2 and GPT-5.5 to ensure linguistic clarity. All technical content, analysis scripts, and final decisions were made by the research team.

⁶<https://www.japaneselawtranslation.go.jp/>

Acknowledgements

In this study, we used the proprietary annotation data of delegation in Japanese law, provided by DAI-ICHI HOKI CO., LTD. We thank the anonymous reviewers for their valuable comments and suggestions.

References

- Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2022. [ExtEnD: Extractive entity disambiguation](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2478–2488, Dublin, Ireland. Association for Computational Linguistics.
- Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S. Dhillon. 2020. [Taming pre-trained transformers for extreme multi-label text classification](#). In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '20*, page 3163–3171, New York, NY, USA. Association for Computing Machinery.
- Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. 2018. [Ultra-fine entity typing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia. Association for Computational Linguistics.
- Micaela Del Monte and Rafał Mańko. 2021. Understanding delegated and implementing acts. [https://www.europarl.europa.eu/RegData/etudes/BRIE/2021/690709/EPRS_BRI\(2021\)690709_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/BRIE/2021/690709/EPRS_BRI(2021)690709_EN.pdf). Accessed: 2025-11-15.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yifan Ding, Qingkai Zeng, and Tim Weninger. 2024. [ChatEL: Entity linking with chatbots](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3086–3097, Torino, Italia. ELRA and ICCL.
- Cunxiao Du, Zhaozheng Chen, Fuli Feng, Lei Zhu, Tian Gan, and Liqiang Nie. 2019. [Explicit interaction model towards text classification](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6359–6366.
- Thibault Févry, Nicholas FitzGerald, Livio Baldini Soares, and Tom Kwiatkowski. 2020. [Empirical evaluation of pretraining strategies for supervised entity linking](#). In *Automated Knowledge Base Construction*.
- Tsuyoshi Fujita, Yuya Sawada, Yusuke Sakai, and Taro Watanabe. 2026. [Toward automatic delegation extraction in Japanese law](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 688–710, Rabat, Morocco. Association for Computational Linguistics.
- Daniel Gillick, Alessandro Presta, and Gaurav Singh Tomar. 2018. [End-to-end retrieval in continuous space](#). *Preprint*, arXiv:1811.08008.
- Randy Goebel, Yoshinobu Kano, Mi-Young Kim, Julianiano Rabelo, Ken Satoh, and Masaharu Yoshioka. 2024. [Overview of benchmark datasets and methods for the legal information extraction/entailment competition \(coliee\) 2024](#). In *New Frontiers in Artificial Intelligence*, pages 109–124, Singapore. Springer Nature Singapore.
- Chuan Guo, Ali Mousavi, Xiang Wu, Daniel N Holtmann-Rice, Satyen Kale, Sashank Reddi, and Sanjiv Kumar. 2019. [Breaking the glass ceiling for embedding-based classifiers for large output spaces](#). In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Nilesh Gupta, Fnu Devvrit, Ankit Singh Rawat, Srinadh Bhojanapalli, Prateek Jain, and Inderjit S Dhillon. 2024. [Dual-encoders for extreme multi-label classification](#). In *The Twelfth International Conference on Learning Representations*.
- Chengyue Jiang, Wenyang Hui, Yong Jiang, Xiaobin Wang, Pengjun Xie, and Kewei Tu. 2023. [Recall, expand, and multi-candidate cross-encode: Fast and accurate ultra-fine entity typing](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11597–11609, Toronto, Canada. Association for Computational Linguistics.
- Dan Li, Zi Long Zhu, Janneke van de Loo, Agnes Masip Gomez, Vikrant Yadav, Georgios Tsatsaronis, and Zubair Afzal. 2023. [Enhancing extreme multi-label text classification: Addressing challenges in model, data, and evaluation](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 313–321, Singapore. Association for Computational Linguistics.
- Nankai Lin, Guanqiu Qin, Gang Wang, Dong Zhou, and Aimin Yang. 2023. [An effective deployment of contrastive learning in multi-label text classification](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8730–8744, Toronto, Canada. Association for Computational Linguistics.
- Antoine Louis and Gerasimos Spanakis. 2022. [A statutory article retrieval dataset in French](#). In *Proceedings of the 60th Annual Meeting of the Association*

- for *Computational Linguistics (Volume 1: Long Papers)*, pages 6789–6803, Dublin, Ireland. Association for Computational Linguistics.
- Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2023. [Finding the law: Enhancing statutory article retrieval via graph neural networks](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2761–2776, Dubrovnik, Croatia. Association for Computational Linguistics.
- Jorge Martinez-Gil. 2023. [A survey on legal question-answering systems](#). *Computer Science Review*, 48:100552.
- Hibiki Nakatani, Hiroki Teranishi, Shohei Higashiyama, Yuya Sawada, Hiroki Ouchi, and Taro Watanabe. 2025. [A text embedding model with contrastive example mining for point-of-interest geocoding](#). In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7279–7291, Abu Dhabi, UAE. Association for Computational Linguistics.
- OpenAI, :, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, and 108 others. 2025. [gpt-oss-120b & gpt-oss-20b model card](#). *Preprint*, arXiv:2508.10925.
- Riccardo Orlando, Pere-Lluís Huguet Cabot, Edoardo Barba, and Roberto Navigli. 2024. [ReLiK: Retrieve and LinK, fast and accurate entity linking and relation extraction on an academic budget](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14114–14132, Bangkok, Thailand. Association for Computational Linguistics.
- Youri Peskine, Damir Korenčić, Ivan Grubisic, Paolo Pappotti, Raphael Troncy, and Paolo Rosso. 2023. [Definitions matter: Guiding GPT for multi-label classification](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4054–4063, Singapore. Association for Computational Linguistics.
- Duncan Sim, Richard Whitaker, and Graeme Cowie. 2024. Delegated powers and framework legislation. <https://researchbriefings.files.parliament.uk/documents/CBP-10046/CBP-10046.pdf>. Accessed: 2025-11-15.
- Weihang Su, Yiran Hu, Anzhe Xie, Qingyao Ai, Quezi Bing, Ning Zheng, Yun Liu, Weixing Shen, and Yiqun Liu. 2024. [STARD: A Chinese statute retrieval dataset derived from real-life queries by non-professionals](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10658–10671, Miami, Florida, USA. Association for Computational Linguistics.
- Seyed Amin Tabatabaei, Sarah Fancher, Michael Parsons, and Arian Askari. 2025. [Can large language models serve as effective classifiers for hierarchical multi-label classification of scientific documents at industrial scale?](#) In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 163–174, Abu Dhabi, UAE. Association for Computational Linguistics.
- Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. [Joint embedding of words and labels for text classification](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2321–2331, Melbourne, Australia. Association for Computational Linguistics.
- Junxiong Wang, Ali Mousavi, Omar Attia, Ronak Pradeep, Saloni Potdar, Alexander Rush, Umar Farooq Minhas, and Yunyao Li. 2024a. [Entity disambiguation via fusion entity decoding](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6524–6536, Mexico City, Mexico. Association for Computational Linguistics.
- Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024b. [Text embeddings by weakly-supervised contrastive pre-training](#). *Preprint*, arXiv:2212.03533.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024c. [Multilingual e5 text embeddings: A technical report](#). *Preprint*, arXiv:2402.05672.
- Keith E. Whittington and Jason Iuliano. 2017. [The myth of the nondelegation doctrine](#). *University of Pennsylvania Law Review*, 165(2):379–431.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Trans-formers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. [Scalable zero-shot entity linking with dense entity retrieval](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.
- Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. [LUKE: Deep contextualized entity representations with entity-aware self-attention](#). In *Proceedings of the 2020*

Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6442–6454, Online. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.

Jun Yang, Weizhi Ma, Min Zhang, Xin Zhou, Yiqun Liu, and Shaoping Ma. 2021. [Legalgnn: Legal information enhanced graph neural network for recommendation](#). *ACM Trans. Inf. Syst.*, 40(2).

Toshihiro Yoshida. 2012. [Introduction to legal research for R&D and business : Part 2: National legal system \[連載 研究・実務に役立つ! リーガル・リサーチ入門 第2回 法体系\]](#). *Journal of Information Processing and Management*, 55(8):591–595. (In Japanese).

Wenzheng Zhang, Wenyue Hua, and Karl Stratos. 2022. [EntQA: Entity linking as question answering](#). In *International Conference on Learning Representations*.

Yaxin Zhu and Hamed Zamani. 2024. [ICXML: An in-context learning framework for zero-shot extreme multi-label classification](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2086–2098, Mexico City, Mexico. Association for Computational Linguistics.

A Prompt for LLM Classifier

At the inference of the LLM classifier, we use an English prompt for LLM_{qwen}-* and LLM_{gpt}-*, while we use a Japanese prompt for LLM_{qwen}-s-* and LLM_{gpt}-s-*. Figures 6 and 7 show the English and Japanese prompt templates, respectively. `<input provision>` denotes the description of an input provision, and `<delegation keyword>` denotes a delegation keyword contained in the input provision. `<candidate 1>`, `<candidate 2>`, and so on, denote the descriptions of the candidate provisions corresponding to the input provision, fetched by the candidate retriever. `<input provision>` and candidate descriptions are constructed following [Fujita et al. \(2026\)](#).

B Detailed Experimental Setup

B.1 Dataset

Japanese laws exhibit a hierarchical structure comprising elements at different levels of granularity, such as “Article,” “Paragraph,” and “Item,” as illustrated in Figure 8 (reproduced from [Fujita et al. \(2026\)](#)). In this hierarchy, higher-level elements

```
<input provision>

Select all the provisions that are delegated targets indicated by “<delegation keyword>” in the above legal text from the list below, and provide their numbers as your answer. If none of the delegated target provisions are included in the list below, do not output anything. In your answer, output only the numbers separated by commas, and do not output any provision text, explanations, or anything else.

1. <candidate 1>
2. <candidate 2>
⋮
```

Figure 6: English prompt template used for LLM classifier.

```
<input provision>

上記の法令文中の「<delegation keyword>」が示す委任先条文を以下の中から全て選び、その番号を回答してください。委任先条文が以下の中に存在しない場合は、何も出力しないでください。回答では番号のみをコンマ区切りで出力し、委任先条文のテキストや説明等は一切出力しないでください。

1. <candidate 1>
2. <candidate 2>
⋮
```

Figure 7: Japanese prompt template used for LLM classifier.

may contain multiple lower-level elements. For example, “Article 1” of a law may include “Paragraph 1,” “Paragraph 2,” and so on, and “Paragraph 2” may further include “Item 1,” “Item 2,” and so forth. The provision database constructed by [Fujita et al. \(2026\)](#) covers approximately 2.28 million provisions from all laws enacted by the National Diet of Japan and orders issued by the Japanese national administrative bodies, at the granularities of entire law, article, paragraph, item, and entire supplementary provisions section. Each provision is assigned a unique provision ID. The delegation extraction dataset created by [Fujita et al. \(2026\)](#) comprises 20,386 sentences selected from those obtained by segmenting all currently effective statutory provisions at sentence boundaries, restricted to sentences that contain delegation keywords. Each sentence is manually annotated with the positions of 20,723 delegation keywords and 35,035 gold delegation target provision IDs corresponding to the keywords. The gold delegation target labels included in the delegation extraction dataset, as well as the breakdown of legal provisions by granularity in the provision database, are shown in Table 5 (reproduced from [Fujita et al. \(2026\)](#)).

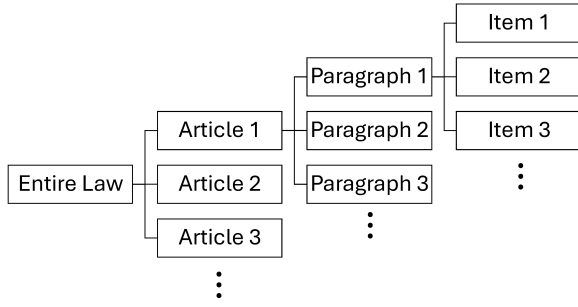


Figure 8: Hierarchical structure within Japanese law.

Granularity	Delegation target labels	Provision database
Entire law	5,169	29,788
Article	16,923	418,855
Paragraph	12,906	1,020,253
Item	25	599,426
Supp	12	208,252
Total	35,035	2,276,574

Table 5: Breakdown of delegation target labels and the provision database by granularity. The number of delegation target labels is reported as total occurrences. “Supp” denotes the granularity corresponding to the entire supplementary provisions section.

As described in Section 3, following Fujita et al. (2026), this study evaluates model performance at four levels of granularity: the entire law, article, paragraph, and item. For evaluation at a given granularity, when the delegation target is specified at a finer granularity than the evaluation level, a prediction is regarded as correct if it matches the gold target up to the evaluation granularity. For example, in article-level evaluation, even if the delegation target is specified at the paragraph or item level, a prediction is considered correct as long as it belongs to the same article as the gold target. In contrast, when the delegation target is specified at a coarser granularity than the evaluation level, a prediction is regarded as correct only if it exactly matches the gold target. For example, in article-level evaluation, if the delegation target is the entire law, only predicting the entire law is considered correct, whereas predicting a specific article within that law is considered incorrect.

B.2 Model Implementation

We use the Hugging Face Transformers (Wolf et al., 2020) implementation for the base models of the candidate retrievers and the candidate classifiers. The base models and their Hugging Face IDs are shown in Table 6. Qwen3 Swallow and gpt-

Models	Hugging Face ID
Candidate Retriever	
E5	intfloat/multilingual-e5-base
Label-Embedding Classifier	
BERT	tohoku-nlp/bert-large-japanese-v2
LUKE	studio-ousia/luke-japanese-large-lite
Binary Classifier	
BERT	tohoku-nlp/bert-large-japanese-v2
LLM Classifier	
Qwen3	Qwen/Qwen3-32B-AWQ
Qwen3 Swallow	tokyotech-llm/Qwen3-Swallow-32B-RL-v0.2-AWQ-INT4
gpt-oss	openai/gpt-oss-120b
gpt-oss Swallow	tokyotech-llm/GPT-OSS-Swallow-120B-RL-v0.1

Table 6: List of the models we used in this study and their corresponding Hugging Face IDs.

	Ret-Multi	Ret-Single
batch size (train)	4	16
batch size (eval)	16	16
learning rate	1e−5	1e−5
epochs	1	4
optimizer	AdamW	AdamW
eps	1e−6	1e−6
scheduler	linear	linear
warmup ratio	0.06	0.06
weight decay	0.01	0.01
max grad norm	0.00	0.00
β	[0.9, 0.98]	[0.9, 0.98]

Table 7: Hyperparameters used in the training of the candidate retrievers. Both Ret-Multi1 and Ret-Multi2 use the hyperparameters listed in the “Ret-Multi” column.

	Label-Embedding Classifier	Binary Classifier
batch size (train)	16	2
batch size (eval)	32	8
learning rate	5e−5	5e−5
epochs	20	4
optimizer	AdamW	AdamW
eps	1e−6	1e−6
scheduler	linear	linear
warmup ratio	0.1	0.1
weight decay	0.01	0.01
max grad norm	0.00	0.00
β	[0.9, 0.98]	[0.9, 0.98]

Table 8: Hyperparameters used in the training of the label-embedding classifiers and the binary classifiers. The same hyperparameters are used in the training of *-Single, *-Multi1, and *-Multi2 variants.

oss Swallow are developed through continual pre-training, supervised finetuning, and reinforcement learning to improve Japanese language abilities, based on Qwen3 and gpt-

B.3 Candidate Retriever

We train the candidate retrievers with the hyperparameters listed in Table 7, on a single NVIDIA

RTX A6000 GPU with 48GB of memory. Following Fujita et al. (2026), we adopt a granularity-aware training strategy that gradually changes the granularity of the training data in a coarse-to-fine manner, i.e., from the law-level to the article-level, and then to the paragraph-level. For each model, training took approximately 1 day, and inference took approximately 10 minutes.

Although Ret-Single and Ret-Multi use different batch sizes and epoch counts, they receive approximately the same number of optimization steps: Ret-Single uses four epochs with a batch size of 16, while Ret-Multi uses one epoch with a batch size of 4, resulting in approximately $N/4$ updates for both, where N is the number of training inputs. Their candidate budgets are also matched. Ret-Single uses one positive, 15 random negatives, and 10 hard negatives, where the random negatives are the positive candidates of other inputs in the minibatch following Fujita et al. (2026). Ret-Multi instead samples random negatives directly from the provision database following Zhang et al. (2022). For an input with one gold target, it uses 26 candidates consisting of one positive, 15 random negatives, and 10 hard negatives. Although the sampling procedures differ, both provide non-hard negatives selected without input-specific similarity and serve essentially the same role.

For inputs with multiple gold targets, Ret-Multi limits positives to at most half of the 26 candidates, because the number of gold targets can exceed the training batch size in the multi-label setting. Thus, it retains up to 13 positives; when an input has more than 13 gold targets, 13 are randomly sampled. Given $p \leq 13$ retained positives, the remaining $26 - p$ slots are filled with $\lfloor 0.6(26 - p) \rfloor$ random negatives and the rest with hard negatives. This keeps the total candidate budget fixed at 26 while allowing additional positives to replace negative candidates as required by the multi-label setting.

As an evaluation metric for the candidate retrievers, we use R@30, which is defined as follows:

$$\text{R@30} = \frac{\sum_{i=1}^N |G_i \cap C_i^{30}|}{\sum_{i=1}^N |G_i|} \quad (10)$$

Here, i indexes delegation keywords, N denotes the number of such keywords in the evaluation set, G_i represents the set of gold target provisions associated with keyword i , and C_i^{30} denotes the set of the top-30 delegation target candidates retrieved by the model for keyword i .

B.4 Candidate Classifier

We train the label-embedding classifiers and the binary classifiers with the hyperparameters listed in Table 8. Both training and inference are conducted on a single NVIDIA RTX A6000 GPU with 48GB of memory. Training of the label-embedding classifiers and binary classifiers took approximately 5 hours and 14 hours, respectively, while inference took approximately 1 minute for label-embedding classifiers and 20 minutes for binary classifiers. During the inference of the LLM classifiers, we use a single NVIDIA RTX A6000 GPU for LLM_{qwen} and $\text{LLM}_{\text{qwen-s}}$, a single NVIDIA RTX PRO 6000 GPU with 96GB of memory for LLM_{gpt} , and four NVIDIA A100 GPUs with 80 GB of memory for $\text{LLM}_{\text{gpt-s}}$. The inference took approximately 2 days for LLM_{qwen} and $\text{LLM}_{\text{qwen-s}}$, 1.5 hours for LLM_{gpt} , and 5.5 hours for $\text{LLM}_{\text{gpt-s}}$.

At the inference of the label-embedding classifiers and the binary classifiers, the classification threshold is set to 0.5, and candidate provisions with scores of 0.5 or higher are predicted as delegation targets. Decoding temperature of the LLM classifiers is set to 0 for deterministic outputs.

B.5 Pipeline System

We evaluate the pipeline systems by computing precision and recall as defined in Equations 11 and 12, respectively. F1 score is defined as the harmonic mean of precision and recall.

$$\text{Precision} = \frac{\sum_{i=1}^N |P_i \cap G_i|}{\sum_{i=1}^N |P_i|} \quad (11)$$

$$\text{Recall} = \frac{\sum_{i=1}^N |P_i \cap G_i|}{\sum_{i=1}^N |G_i|} \quad (12)$$

Here, P_i denotes the set of delegation target candidates predicted by the pipeline systems for delegation keyword i .

C Detailed Analysis

C.1 Effect of Candidate Set Size

The main experiments use a fixed candidate set size of 30. To examine how the candidate budget affects pipeline performance, we additionally evaluate Pipe-Bin-* with candidate set size $K = \{10, 100, 150\}$, together with the original setting of $K = 30$. This range allows us to evaluate both smaller and substantially larger candidate sets, including a setting in which complete recovery of all gold delegation targets is theoretically possible.

Model		Granularity	Precision	Recall	F1
Bin ($K = 10$)	-Single	Law	89.7 $\pm$ 2.05	74.6 $\pm$ 4.53	81.4 $\pm$ 3.42
		Article	75.4 $\pm$ 6.42	49.0 $\pm$ 3.96	59.3 $\pm$ 3.98
		Paragraph	67.3 $\pm$ 7.86	45.6 $\pm$ 4.33	54.1 $\pm$ 4.41
		Item	67.3 $\pm$ 7.88	45.6 $\pm$ 4.32	54.1 $\pm$ 4.41
	-Multi1	Law	91.1 $\pm$ 2.66	64.9 $\pm$ 7.03	75.7 $\pm$ 5.78
		Article	78.9 $\pm$ 5.02	41.8 $\pm$ 6.63	54.5 $\pm$ 6.90
		Paragraph	68.5 $\pm$ 10.9	39.1 $\pm$ 6.65	49.7 $\pm$ 8.18
		Item	68.5 $\pm$ 10.9	39.1 $\pm$ 6.66	49.7 $\pm$ 8.19
	-Multi2	Law	89.9 $\pm$ 4.47	67.2 $\pm$ 3.81	76.8 $\pm$ 2.41
		Article	77.5 $\pm$ 7.46	43.2 $\pm$ 3.05	55.3 $\pm$ 3.26
		Paragraph	70.4 $\pm$ 6.72	40.6 $\pm$ 2.73	51.3 $\pm$ 2.91
		Item	70.4 $\pm$ 6.72	40.6 $\pm$ 2.74	51.3 $\pm$ 2.91
Bin ($K = 30$)	-Single	Law	83.1 $\pm$ 3.58	78.1 $\pm$ 4.57	80.5 $\pm$ 3.97
		Article	64.3 $\pm$ 9.68	53.2 $\pm$ 4.70	57.9 $\pm$ 5.66
		Paragraph	56.2 $\pm$ 10.5	48.9 $\pm$ 5.61	51.8 $\pm$ 6.61
		Item	56.2 $\pm$ 10.6	48.9 $\pm$ 5.61	51.7 $\pm$ 6.62
	-Multi1	Law	87.0 $\pm$ 4.06	72.9 $\pm$ 8.32	79.2 $\pm$ 6.58
		Article	71.6 $\pm$ 6.27	48.6 $\pm$ 8.05	57.8 $\pm$ 7.77
		Paragraph	60.8 $\pm$ 10.2	45.3 $\pm$ 8.09	51.9 $\pm$ 9.00
		Item	60.7 $\pm$ 10.2	45.3 $\pm$ 8.09	51.9 $\pm$ 9.01
	-Multi2	Law	85.4 $\pm$ 7.00	75.8 $\pm$ 4.71	80.0 $\pm$ 3.72
		Article	70.1 $\pm$ 9.54	51.1 $\pm$ 5.01	58.6 $\pm$ 5.10
		Paragraph	62.4 $\pm$ 8.76	47.6 $\pm$ 4.65	53.6 $\pm$ 4.76
		Item	62.4 $\pm$ 8.76	47.6 $\pm$ 4.65	53.6 $\pm$ 4.76
Bin ($K = 100$)	-Single	Law	75.0 $\pm$ 5.99	80.3 $\pm$ 4.56	77.5 $\pm$ 5.17
		Article	53.2 $\pm$ 11.8	57.3 $\pm$ 6.13	54.4 $\pm$ 8.04
		Paragraph	45.8 $\pm$ 12.0	52.2 $\pm$ 7.37	47.8 $\pm$ 9.01
		Item	45.8 $\pm$ 12.0	52.2 $\pm$ 7.37	47.8 $\pm$ 9.03
	-Multi1	Law	80.8 $\pm$ 6.42	78.6 $\pm$ 9.02	79.5 $\pm$ 7.32
		Article	61.8 $\pm$ 7.01	54.7 $\pm$ 10.0	57.9 $\pm$ 8.71
		Paragraph	51.1 $\pm$ 8.85	50.8 $\pm$ 9.96	50.9 $\pm$ 9.39
		Item	51.0 $\pm$ 8.88	50.8 $\pm$ 9.96	50.9 $\pm$ 9.41
	-Multi2	Law	79.1 $\pm$ 10.1	81.8 $\pm$ 5.31	79.9 $\pm$ 5.88
		Article	60.9 $\pm$ 10.9	57.9 $\pm$ 7.05	58.5 $\pm$ 6.69
		Paragraph	53.4 $\pm$ 10.2	53.4 $\pm$ 6.61	52.6 $\pm$ 6.27
		Item	53.4 $\pm$ 10.2	53.4 $\pm$ 6.61	52.6 $\pm$ 6.27
Bin ($K = 150$)	-Single	Law	71.9 $\pm$ 6.96	81.0 $\pm$ 4.62	76.1 $\pm$ 5.78
		Article	49.5 $\pm$ 12.2	58.4 $\pm$ 6.68	52.6 $\pm$ 8.84
		Paragraph	42.5 $\pm$ 12.2	53.3 $\pm$ 8.02	46.1 $\pm$ 9.71
		Item	42.5 $\pm$ 12.2	53.3 $\pm$ 8.02	46.1 $\pm$ 9.73
	-Multi1	Law	78.4 $\pm$ 7.36	79.9 $\pm$ 9.19	78.9 $\pm$ 7.70
		Article	58.0 $\pm$ 6.99	56.5 $\pm$ 10.8	57.1 $\pm$ 8.91
		Paragraph	47.5 $\pm$ 8.18	52.5 $\pm$ 10.7	49.8 $\pm$ 9.31
		Item	47.4 $\pm$ 8.22	52.5 $\pm$ 10.7	49.8 $\pm$ 9.33
	-Multi2	Law	76.5 $\pm$ 11.7	83.1 $\pm$ 5.37	79.0 $\pm$ 7.12
		Article	57.5 $\pm$ 11.4	59.6 $\pm$ 7.76	57.5 $\pm$ 7.28
		Paragraph	50.2 $\pm$ 10.7	54.9 $\pm$ 7.30	51.5 $\pm$ 6.80
		Item	50.2 $\pm$ 10.6	54.9 $\pm$ 7.30	51.5 $\pm$ 6.80

Table 9: Performance of the pipeline systems using binary classifiers with various candidate set sizes K .

Specifically, the maximum number of gold delegation targets associated with a single delegation keyword in our dataset is 145; therefore, a candidate set size of 150 is sufficient to contain all gold targets for any input.

Table 9 shows the results. There is a clear precision–recall trade-off as the candidate set size

Model		Granularity	Precision	Recall	F1
Emb _{bert}	-Single	Law	44.7 _{±1.98}	74.5 _{±1.88}	55.8 _{±1.54}
		Article	15.0 _{±0.87}	30.5 _{±1.00}	20.1 _{±0.82}
		Paragraph	7.33 _{±0.65}	17.2 _{±0.86}	10.3 _{±0.75}
		Item	6.28 _{±0.59}	16.5 _{±0.85}	9.09 _{±0.71}
	-Multi1	Law	40.9 _{±0.54}	60.4 _{±2.03}	48.8 _{±0.99}
		Article	13.2 _{±0.37}	22.2 _{±0.78}	16.5 _{±0.31}
		Paragraph	5.76 _{±0.17}	11.3 _{±0.43}	7.62 _{±0.14}
		Item	4.91 _{±0.20}	10.6 _{±0.37}	6.70 _{±0.16}
-Multi2	Law	44.7 _{±3.16}	61.4 _{±4.73}	51.5 _{±1.91}	
	Article	13.1 _{±0.93}	22.4 _{±2.72}	16.4 _{±0.75}	
	Paragraph	5.73 _{±0.54}	11.5 _{±1.38}	7.60 _{±0.47}	
	Item	4.80 _{±0.43}	10.7 _{±1.30}	6.58 _{±0.40}	
Emb _{luke}	-Single	Law	44.4 _{±2.64}	75.4 _{±1.09}	55.8 _{±1.83}
		Article	15.0 _{±0.92}	31.2 _{±1.51}	20.2 _{±0.57}
		Paragraph	7.35 _{±0.69}	17.7 _{±0.72}	10.4 _{±0.67}
		Item	6.29 _{±0.65}	16.9 _{±0.68}	9.14 _{±0.68}
	-Multi1	Law	30.4 _{±1.11}	78.7 _{±0.47}	43.8 _{±1.16}
		Article	8.91 _{±0.36}	30.3 _{±1.57}	13.8 _{±0.53}
		Paragraph	4.02 _{±0.14}	15.8 _{±0.75}	6.40 _{±0.22}
		Item	3.25 _{±0.16}	14.5 _{±0.75}	5.32 _{±0.25}
	-Multi2	Law	29.6 _{±1.44}	76.7 _{±0.71}	42.7 _{±1.50}
		Article	8.36 _{±0.37}	27.5 _{±1.03}	12.8 _{±0.51}
		Paragraph	3.86 _{±0.17}	14.5 _{±1.00}	6.10 _{±0.29}
		Item	3.16 _{±0.13}	13.5 _{±0.96}	5.12 _{±0.23}

Table 10: Performance of the pipeline systems using label-embedding classifiers.

increases. In general, larger candidate sets improve recall by giving the candidate classifier access to more potentially relevant targets, but they also reduce precision because the classifier must distinguish the gold targets from a larger number of negative candidates. Consequently, increasing the candidate set size does not necessarily improve F1. This trend is particularly evident at the finest item-level evaluation. Pipe-Bin-Single achieves its highest F1 with 10 candidates, whereas Pipe-Bin-Multi1 and Pipe-Bin-Multi2 achieve their highest F1 with the original candidate set size of 30. Increasing the candidate set size to 100 or 150 further improves recall in many cases, but the accompanying reduction in precision offsets these gains and results in lower F1.

C.2 Comparison across Base Models of Candidate Classifier

Tables 10 and 11 show the performance of the pipeline systems using all the label-embedding classifiers and LLM classifiers, respectively. Precision of the *-Single variants of these pipeline systems, as well as their recall, is generally higher than that of the *-Multi1 and *-Multi2 variants, leading to higher F1 scores for the *-Single variants. These

Model		Granularity	Precision	Recall	F1
LLM _{qwen}	-Single	Law	53.8 $\pm$ 0.62	92.4$\pm$1.11	67.9 $\pm$ 0.29
		Article	22.3 $\pm$ 0.55	61.5$\pm$1.79	32.7 $\pm$ 0.65
		Paragraph	13.3 $\pm$ 0.34	50.9$\pm$1.75	21.0 $\pm$ 0.41
		Item	9.85 $\pm$ 0.32	49.8$\pm$1.87	16.4 $\pm$ 0.44
	-Multi1	Law	47.6 $\pm$ 1.13	88.2 $\pm$ 1.00	61.8 $\pm$ 0.77
		Article	20.7 $\pm$ 0.52	54.7 $\pm$ 1.99	30.0 $\pm$ 0.55
		Paragraph	12.5 $\pm$ 0.26	46.1 $\pm$ 1.73	19.7 $\pm$ 0.40
		Item	10.0 $\pm$ 0.22	45.4 $\pm$ 1.78	16.4 $\pm$ 0.36
	-Multi2	Law	50.5 $\pm$ 0.74	88.4 $\pm$ 1.05	64.3 $\pm$ 0.72
		Article	20.9 $\pm$ 0.48	56.0 $\pm$ 1.29	30.4 $\pm$ 0.33
		Paragraph	12.3 $\pm$ 0.30	47.2 $\pm$ 0.88	19.5 $\pm$ 0.33
		Item	9.66 $\pm$ 0.32	46.3 $\pm$ 0.97	16.0 $\pm$ 0.42
LLM _{qwen-s}	-Single	Law	65.1 $\pm$ 0.83	90.0 $\pm$ 0.86	75.5 $\pm$ 0.48
		Article	31.6 $\pm$ 0.62	57.9 $\pm$ 1.55	40.9 $\pm$ 0.41
		Paragraph	19.3 $\pm$ 0.74	46.6 $\pm$ 1.31	27.3$\pm$0.80
		Item	15.6 $\pm$ 0.67	45.8 $\pm$ 1.41	23.2$\pm$0.80
	-Multi1	Law	60.5 $\pm$ 0.62	85.7 $\pm$ 1.12	70.9 $\pm$ 0.15
		Article	30.3 $\pm$ 0.44	50.8 $\pm$ 2.08	38.0 $\pm$ 0.56
		Paragraph	18.5 $\pm$ 0.74	41.7 $\pm$ 2.06	25.6 $\pm$ 1.02
		Item	15.7 $\pm$ 0.73	41.1 $\pm$ 2.08	22.7 $\pm$ 1.00
	-Multi2	Law	62.8 $\pm$ 1.02	85.9 $\pm$ 0.91	72.5 $\pm$ 0.77
		Article	30.7 $\pm$ 0.21	51.9 $\pm$ 1.57	38.6 $\pm$ 0.31
		Paragraph	17.9 $\pm$ 0.39	41.7 $\pm$ 1.40	25.0 $\pm$ 0.51
		Item	14.9 $\pm$ 0.38	41.0 $\pm$ 1.47	21.9 $\pm$ 0.50
LLM _{gpt}	-Single	Law	46.8 $\pm$ 0.96	87.8 $\pm$ 1.15	61.1 $\pm$ 0.65
		Article	17.3 $\pm$ 0.59	55.7 $\pm$ 1.29	26.4 $\pm$ 0.68
		Paragraph	11.2 $\pm$ 0.37	50.4 $\pm$ 1.29	18.3 $\pm$ 0.44
		Item	8.34 $\pm$ 0.31	49.5 $\pm$ 1.41	14.3 $\pm$ 0.43
	-Multi1	Law	43.2 $\pm$ 1.49	83.2 $\pm$ 1.21	56.8 $\pm$ 1.11
		Article	15.9 $\pm$ 0.40	46.3 $\pm$ 2.46	23.6 $\pm$ 0.63
		Paragraph	9.92 $\pm$ 0.35	41.3 $\pm$ 2.19	16.0 $\pm$ 0.59
		Item	8.07 $\pm$ 0.30	40.6 $\pm$ 2.23	13.5 $\pm$ 0.50
	-Multi2	Law	46.9 $\pm$ 1.93	83.8 $\pm$ 1.10	60.1 $\pm$ 1.82
		Article	16.3 $\pm$ 0.27	48.2 $\pm$ 1.73	24.4 $\pm$ 0.39
		Paragraph	9.85 $\pm$ 0.19	42.9 $\pm$ 1.36	16.0 $\pm$ 0.27
		Item	7.88 $\pm$ 0.22	42.2 $\pm$ 1.37	13.3 $\pm$ 0.32
LLM _{gpt-s}	-Single	Law	75.0$\pm$0.41	90.7 $\pm$ 0.97	82.1$\pm$0.43
		Article	38.2 $\pm$ 0.74	59.4 $\pm$ 1.88	46.5$\pm$0.91
		Paragraph	19.3 $\pm$ 0.58	37.4 $\pm$ 1.30	25.4 $\pm$ 0.70
		Item	14.8 $\pm$ 0.58	36.4 $\pm$ 1.38	21.0 $\pm$ 0.73
	-Multi1	Law	70.7 $\pm$ 0.75	86.3 $\pm$ 0.94	77.7 $\pm$ 0.65
		Article	38.1 $\pm$ 0.42	53.0 $\pm$ 2.05	44.3 $\pm$ 0.86
		Paragraph	20.4$\pm$1.02	36.1 $\pm$ 2.54	26.1 $\pm$ 1.47
		Item	16.7$\pm$0.98	35.4 $\pm$ 2.62	22.7 $\pm$ 1.41
	-Multi2	Law	72.0 $\pm$ 0.89	86.3 $\pm$ 0.96	78.5 $\pm$ 0.84
		Article	38.3$\pm$0.63	54.0 $\pm$ 1.63	44.8 $\pm$ 0.73
		Paragraph	19.5 $\pm$ 0.64	35.7 $\pm$ 1.81	25.3 $\pm$ 0.94
		Item	15.7 $\pm$ 0.58	34.9 $\pm$ 1.85	21.6 $\pm$ 0.83

Table 11: Performance of the pipeline using LLM classifiers.

results suggest that the label-embedding classifiers and LLM classifiers are relatively robust against the harder-to-distinguish negative candidates, which the binary classifiers may struggle with (see Section 4.3), and that they can convert the better coverage of Ret-Single to the overall performance gain. At the same time, the absolute F1 scores of these

Model		Granularity	Precision	Recall	F1
Emb _{bert}	-Single	Law	57.6 _{±4.65}	37.6 _{±5.24}	45.4 _{±5.13}
		Article	31.3 _{±2.04}	17.9 _{±1.70}	22.7 _{±1.62}
		Paragraph	15.2 _{±1.77}	10.5 _{±1.10}	12.4 _{±1.22}
		Item	12.6 _{±1.57}	9.49 _{±0.98}	10.8 _{±1.10}
	-Multi1	Law	48.2 _{±2.55}	24.8 _{±3.63}	32.6 _{±3.22}
		Article	25.2 _{±1.42}	11.9 _{±1.03}	16.1 _{±1.06}
		Paragraph	11.4 _{±0.90}	6.49 _{±0.79}	8.24 _{±0.80}
		Item	9.21 _{±0.67}	5.64 _{±0.55}	6.98 _{±0.53}
	-Multi2	Law	51.9 _{±3.51}	26.4 _{±6.20}	34.6 _{±6.08}
		Article	27.5 _{±2.40}	13.4 _{±1.70}	17.9 _{±1.68}
		Paragraph	12.3 _{±1.27}	7.25 _{±1.04}	9.07 _{±0.97}
		Item	9.72 _{±0.99}	6.25 _{±1.05}	7.57 _{±0.95}
Emb _{luke}	-Single	Law	56.2 _{±4.14}	37.8 _{±3.75}	45.2 _{±3.98}
		Article	31.1 _{±2.14}	18.4 _{±1.49}	23.0 _{±1.08}
		Paragraph	15.4 _{±1.29}	11.0 _{±0.74}	12.8 _{±0.53}
		Item	12.8 _{±1.12}	9.93 _{±0.68}	11.1 _{±0.41}
	-Multi1	Law	42.5 _{±2.37}	42.1 _{±2.87}	42.2 _{±2.07}
		Article	21.2 _{±1.48}	21.5 _{±0.86}	21.4 _{±0.94}
		Paragraph	10.1 _{±0.65}	12.3 _{±0.36}	11.1 _{±0.42}
		Item	7.89 _{±0.61}	10.7 _{±0.37}	9.08 _{±0.48}
	-Multi2	Law	40.4 _{±2.32}	40.1 _{±3.49}	40.1 _{±2.05}
		Article	20.0 _{±1.94}	18.9 _{±0.90}	19.4 _{±1.30}
		Paragraph	9.57 _{±0.81}	10.9 _{±0.47}	10.2 _{±0.59}
		Item	7.63 _{±0.71}	9.58 _{±0.52}	8.48 _{±0.60}

Table 12: Performance of the pipeline systems using label-embedding classifiers evaluated on data with two or more gold delegation targets.

models are substantially lower than those of the pipeline systems using the binary classifier (see also Table 3 in Section 4.2).

Comparing Pipe-Emb_{bert}-* with Pipe-Emb_{luke}-, there is no substantial difference in their performance. This suggests that the choice of base models for the label-embedding classifier has only a limited impact on overall performance in our setting. On the other hand, comparing Pipe-LLM_{qwen}-* and Pipe-LLM_{gpt}-* with Pipe-LLM_{qwen-s}-* and Pipe-LLM_{gpt-s}-* respectively, the latter consistently outperforms the former. This indicates that the enhancement of the Japanese language abilities in Qwen3 Swallow and gpt-oss Swallow is beneficial in the multi-label delegation target identification task.

Tables 12 and 13 show the performance of the pipeline systems with the label-embedding classifiers and LLM classifiers, evaluated exclusively on instances with two or more gold delegation target provisions. Under this evaluation setting, all the pipeline systems using the LLM classifiers as candidate classifiers, as well as Pipe-Emb_{bert}-Single and Pipe-Emb_{luke}-Single, outperform Naive-Top1 (see also Table 4 in Section 5.1). This suggests that

Model		Granularity	Precision	Recall	F1
LLM _{qwen}	-Single	Law	61.4 $\pm$ 4.53	58.3$\pm$4.84	59.5$\pm$2.09
		Article	37.2 $\pm$ 1.85	38.7$\pm$1.92	37.9 $\pm$ 1.36
		Paragraph	19.7 $\pm$ 0.85	32.3$\pm$1.73	24.4$\pm$0.62
		Item	14.9 $\pm$ 0.69	30.5$\pm$1.93	20.0 $\pm$ 0.73
	-Multi1	Law	49.6 $\pm$ 1.64	45.1 $\pm$ 5.11	47.2 $\pm$ 3.15
		Article	29.3 $\pm$ 1.50	28.3 $\pm$ 1.92	28.7 $\pm$ 0.99
		Paragraph	15.9 $\pm$ 0.35	23.0 $\pm$ 1.91	18.8 $\pm$ 0.76
		Item	12.8 $\pm$ 0.17	21.9 $\pm$ 1.87	16.1 $\pm$ 0.61
	-Multi2	Law	51.1 $\pm$ 4.15	46.5 $\pm$ 3.55	48.6 $\pm$ 3.01
		Article	31.6 $\pm$ 1.47	32.0 $\pm$ 1.40	31.8 $\pm$ 0.59
		Paragraph	16.9 $\pm$ 0.56	26.7 $\pm$ 1.14	20.7 $\pm$ 0.30
		Item	13.1 $\pm$ 0.55	25.2 $\pm$ 1.28	17.2 $\pm$ 0.49
LLM _{qwen-s}	-Single	Law	66.0 $\pm$ 3.11	48.1 $\pm$ 3.21	55.6 $\pm$ 2.27
		Article	45.1 $\pm$ 1.92	30.8 $\pm$ 1.13	36.5 $\pm$ 0.20
		Paragraph	24.0 $\pm$ 1.06	24.1 $\pm$ 0.84	24.0 $\pm$ 0.31
		Item	18.8 $\pm$ 0.82	22.7 $\pm$ 1.04	20.5$\pm$0.36
	-Multi1	Law	56.2 $\pm$ 1.58	38.2 $\pm$ 5.01	45.3 $\pm$ 3.30
		Article	37.5 $\pm$ 1.27	22.4 $\pm$ 1.64	28.0 $\pm$ 1.20
		Paragraph	20.0 $\pm$ 0.49	17.1 $\pm$ 1.30	18.5 $\pm$ 0.89
		Item	16.7 $\pm$ 0.50	16.2 $\pm$ 1.31	16.4 $\pm$ 0.86
	-Multi2	Law	58.0 $\pm$ 4.76	38.7 $\pm$ 1.46	46.3 $\pm$ 1.24
		Article	40.3 $\pm$ 1.26	25.3 $\pm$ 1.30	31.0 $\pm$ 0.88
		Paragraph	20.8 $\pm$ 0.57	19.6 $\pm$ 0.90	20.1 $\pm$ 0.30
		Item	16.8 $\pm$ 0.53	18.5 $\pm$ 1.02	17.6 $\pm$ 0.43
LLM _{gpt}	-Single	Law	57.0 $\pm$ 2.49	53.7 $\pm$ 3.12	55.2 $\pm$ 2.24
		Article	31.0 $\pm$ 2.64	34.0 $\pm$ 1.06	32.4 $\pm$ 1.42
		Paragraph	17.2 $\pm$ 1.18	29.1 $\pm$ 1.14	21.5 $\pm$ 0.82
		Item	13.0 $\pm$ 0.87	27.7 $\pm$ 1.39	17.7 $\pm$ 0.76
	-Multi1	Law	45.8 $\pm$ 2.59	40.2 $\pm$ 4.47	42.6 $\pm$ 2.44
		Article	22.9 $\pm$ 1.88	22.6 $\pm$ 2.18	22.6 $\pm$ 1.36
		Paragraph	12.7 $\pm$ 0.69	18.5 $\pm$ 1.84	15.0 $\pm$ 0.86
		Item	10.2 $\pm$ 0.56	17.6 $\pm$ 1.86	12.9 $\pm$ 0.80
	-Multi2	Law	47.7 $\pm$ 4.67	41.7 $\pm$ 2.98	44.4 $\pm$ 3.15
		Article	24.7 $\pm$ 1.44	26.0 $\pm$ 1.35	25.3 $\pm$ 0.42
		Paragraph	13.4 $\pm$ 0.65	21.8 $\pm$ 1.19	16.6 $\pm$ 0.36
		Item	10.7 $\pm$ 0.54	20.8 $\pm$ 1.31	14.1 $\pm$ 0.41
LLM _{gpt-s}	-Single	Law	73.4$\pm$2.92	47.6 $\pm$ 3.10	57.7 $\pm$ 2.66
		Article	51.9$\pm$1.67	32.4 $\pm$ 1.52	39.9$\pm$1.28
		Paragraph	25.1$\pm$1.19	22.8 $\pm$ 1.05	23.9 $\pm$ 0.92
		Item	19.0$\pm$1.10	21.1 $\pm$ 1.09	20.0 $\pm$ 0.90
	-Multi1	Law	63.6 $\pm$ 2.74	35.9 $\pm$ 2.50	45.8 $\pm$ 2.57
		Article	46.8 $\pm$ 1.30	24.0 $\pm$ 1.80	31.7 $\pm$ 1.57
		Paragraph	23.4 $\pm$ 0.53	16.7 $\pm$ 1.30	19.5 $\pm$ 1.03
		Item	18.9 $\pm$ 0.49	15.6 $\pm$ 1.36	17.1 $\pm$ 1.00
	-Multi2	Law	64.2 $\pm$ 3.61	37.0 $\pm$ 2.64	46.9 $\pm$ 2.26
		Article	49.4 $\pm$ 1.50	27.3 $\pm$ 1.73	35.2 $\pm$ 1.35
		Paragraph	23.8 $\pm$ 0.66	19.2 $\pm$ 1.18	21.2 $\pm$ 0.63
		Item	18.7 $\pm$ 0.70	17.9 $\pm$ 1.22	18.3 $\pm$ 0.56

Table 13: Performance of the pipeline systems using LLM classifiers evaluated on data with two or more gold delegation targets.

the core designs of the label-embedding classifier and the LLM classifier are well-aligned with the multi-label delegation target identification task.

C.3 Number of Gold Delegation Targets and Model Performance

Figures 9–11 show the performance of the pipeline systems using the label-embedding classifiers and LLM classifiers as candidate classifiers, grouped by the number of gold delegation targets. Recall of these pipeline systems across all the granularities drops as the number of gold delegation targets increases, while precision at law, paragraph, and item levels remains stable or increases. The general tendency that precision is more robust against a larger number of gold delegation targets than recall is also observed in Pipe-Bin-* (see Figure 3 in Section 5.2). The evaluation results in Figures 9–11 indicate that the robustness in terms of precision is more pronounced in the pipeline systems using the label-embedding classifiers and the LLM classifiers than in those using the binary classifiers.

C.4 Threshold Values for Binary Classifier

We analyze the performance of the binary classifiers under different classification threshold values ranging from 0.1 to 0.9, with a step size of 0.1. This analysis focuses on the behavior of the binary classifiers themselves, independently of candidate retrievers. Accordingly, scores are computed with respect to the gold delegation targets contained in the retrieved candidate sets.

In this setting, recall is defined as:

$$\text{Recall}_{\text{bin}} = \frac{\sum_{i=1}^N |P_i \cap G_i|}{\sum_{i=1}^N |G_i \cap C_i^{30}|}. \quad (13)$$

By restricting the denominator to $G_i \cap C_i^{30}$, this metric isolates the performance of the binary classifier from errors introduced by the candidate retrieval stage. F1 score is calculated using $\text{Recall}_{\text{bin}}$ and precision defined in Equation 11.

Figure 12 shows the F1 score of Bin-Single, Bin-Multi1, and Bin-Multi2 at various threshold values. The optimal classification threshold in terms of F1 score is not fixed, but varies depending on both the model and the target granularity. Across settings, the highest F1 scores are generally achieved for threshold values between 0.4 and 0.8. In addition, a clear trend can be observed with respect to granularity: as the target granularity becomes finer, the threshold that maximizes the F1 score tends to shift toward larger values. This indicates that finer-grained prediction requires more conservative decision boundaries to suppress false positives,

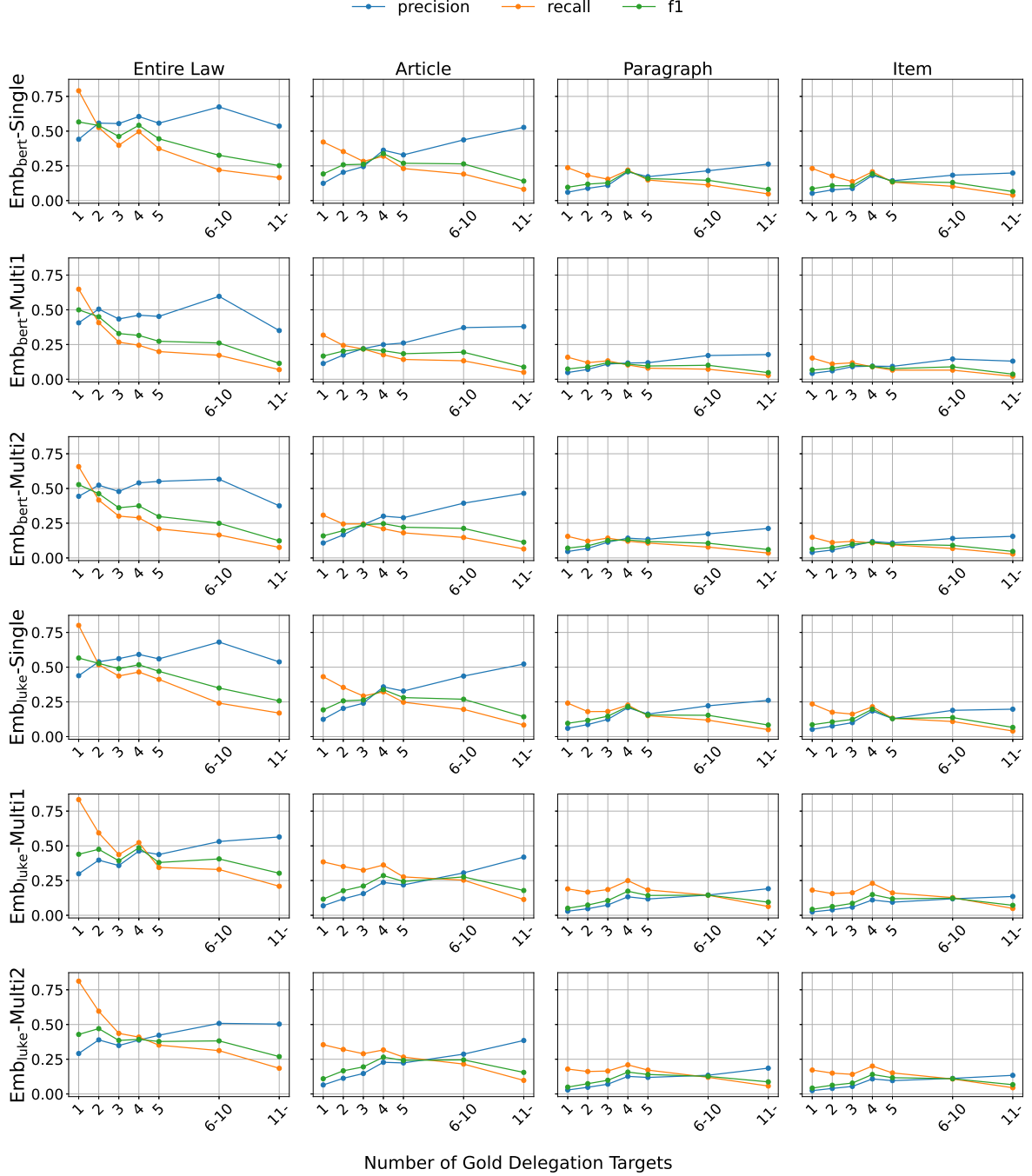


Figure 9: Precision, recall, and F1 scores of Pipe-Emb_{bert}-* and Pipe-Emb_{luke}-, grouped by the number of gold delegation targets. The rows represent the model variants, while the columns represent the evaluation granularities. “Pipe” is omitted from the notation of the model names.

whereas coarser-grained settings benefit from relatively lower thresholds that favor recall.

While the main experiments use a fixed threshold of 0.5 to provide a simple and uniform comparison across models and granularities, the above analysis suggests that selecting the threshold on development data can further improve performance. Therefore, we additionally evaluate Pipe-Bin-* us-

ing calibrated thresholds. For each cross-validation model and evaluation granularity, we select the threshold that achieves the highest F1 score on the corresponding development set and apply that threshold to the test set.

Table 14 reports the resulting pipeline performance. Compared with their fixed-threshold counterparts, the calibrated pipeline systems improve

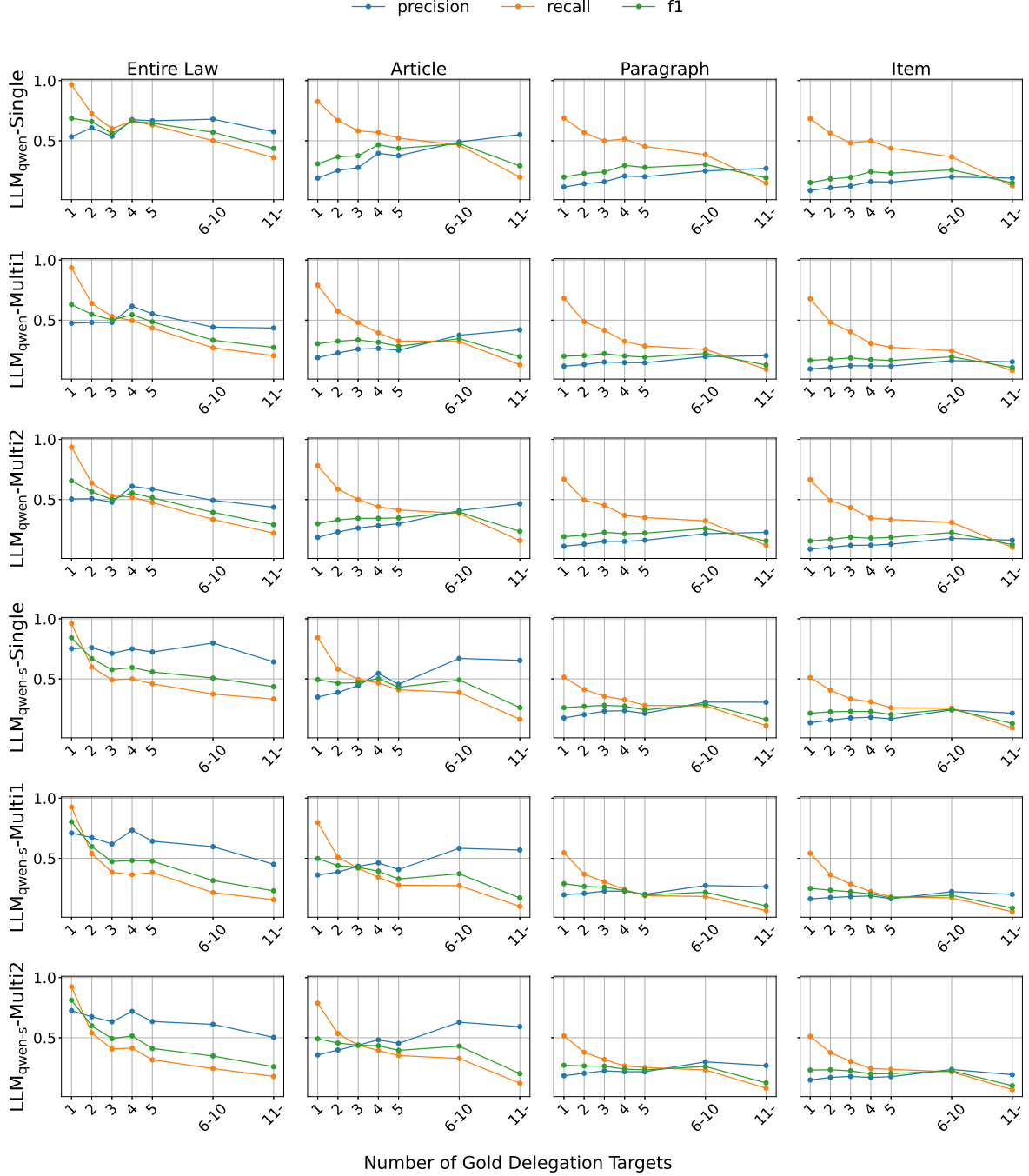


Figure 10: Precision, recall, and F1 scores of Pipe-LLM_{qwen}-* and Pipe-LLM_{qwen-s}-, grouped by the number of gold delegation targets. The rows represent the model variants, while the columns represent the evaluation granularities. “Pipe” is omitted from the notation of the model names.

F1 by approximately 2–3 points at the article, paragraph, and item levels. They also slightly outperform Naive-Top1 at the article level. These results demonstrate that the threshold selection using the development set provides a stronger configuration of Pipe-Bin-*, while the fixed threshold of 0.5 used in the main experiments offers a simpler uniform setting for comparing model behavior.

C.5 Prediction Set Level Evaluation Metrics

In the main experiments, we report micro-averaged precision, recall, and F1 over all delegation keywords. While these metrics summarize overall target-level performance, they do not directly measure whether the complete set of delegation targets is recovered for each query. Therefore, we evaluate the best-performing pipeline systems, Pipe-

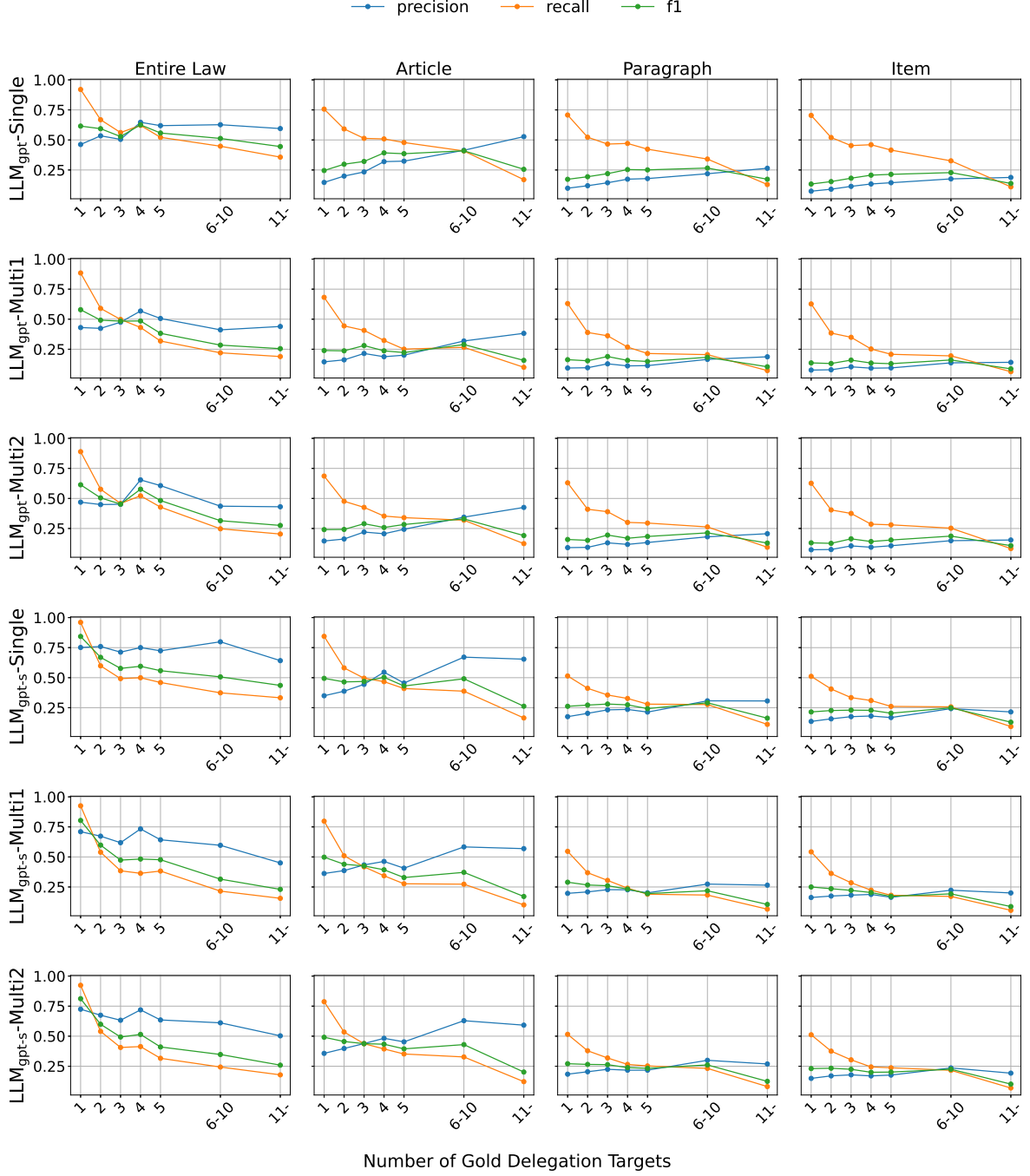


Figure 11: Precision, recall, and F1 scores of Pipe-LLM_{gpt}-* and Pipe-LLM_{gpt-s}*, grouped by the number of gold delegation targets. The rows represent the model variants, while the columns represent the evaluation granularities. “Pipe” is omitted from the notation of the model names.

Bin-*, and Naive-Top1 using three additional metrics: exact-set accuracy (ESA), sample-macro F1 (SMF1), and per-query complete recall (PQCR). They are defined as:

$$ESA = \frac{1}{N} \sum_{i=1}^N \mathbf{1}[P_i = G_i], \quad (14)$$

$$SMF1 = \frac{1}{N} \sum_{i=1}^N \frac{2|P_i \cap G_i|}{|P_i| + |G_i|}, \quad (15)$$

$$PQCR = \frac{1}{N} \sum_{i=1}^N \mathbf{1}[G_i \subseteq P_i]. \quad (16)$$

Here, G_i and P_i are the gold and predicted target sets for query i , respectively. N is the number of delegation keywords, and $\mathbf{1}[\cdot]$ is an indicator

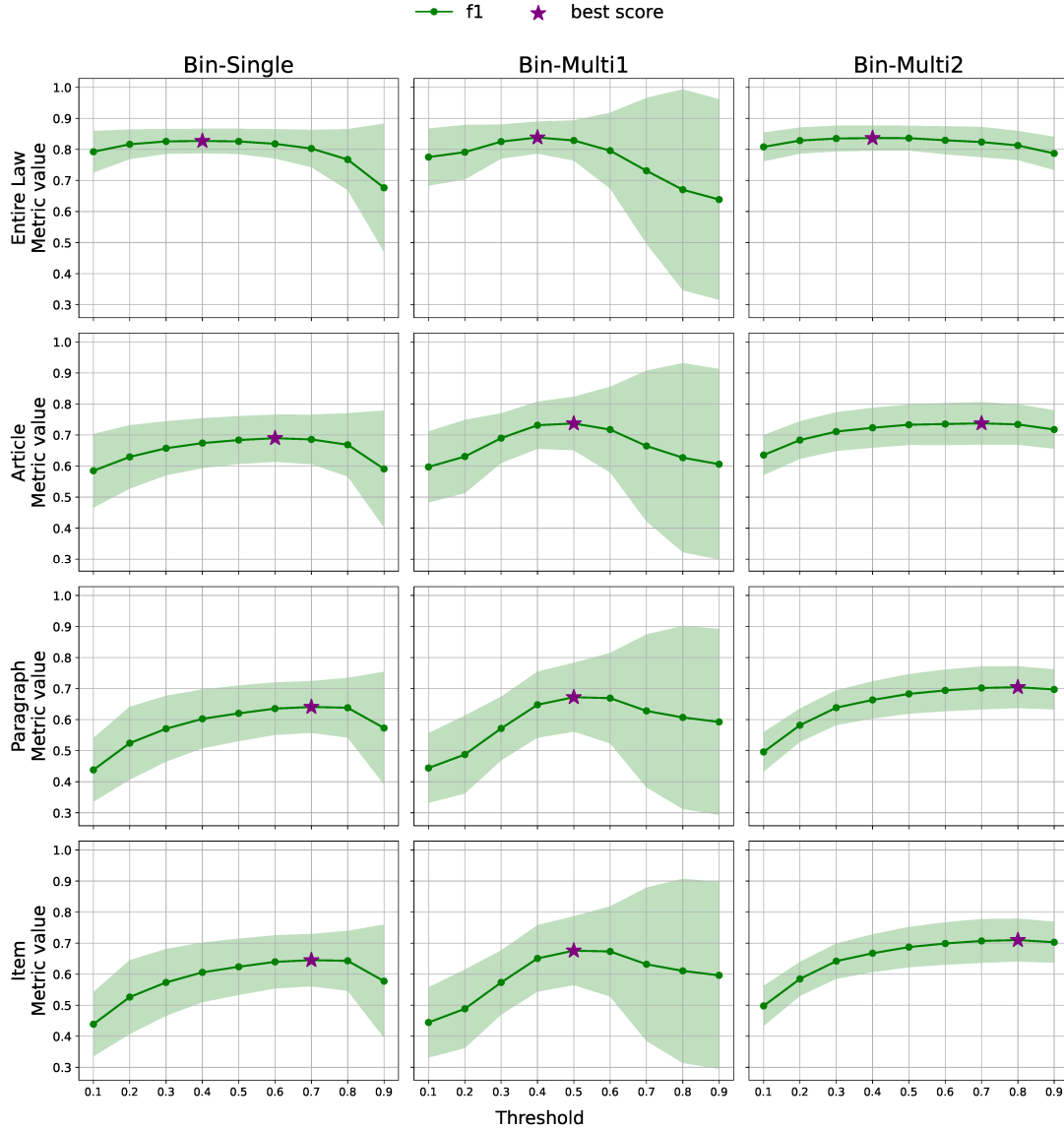


Figure 12: F1 scores of the three binary classifiers at various classification threshold values, evaluated at different granularities. The columns represent the model variants, while the rows represent the granularities. The shaded band shows the standard deviation.

function that equals 1 when the condition inside the brackets is true and 0 otherwise. ESA measures the proportion of delegation keywords for which the predicted and gold target sets match exactly. SMF1 computes F1 scores independently for each query and then averages them across delegation keywords, assigning equal weight to each query regardless of the number of gold targets. PQCR measures the proportion of delegation keywords for which all gold targets are included in the predicted set, regardless of whether additional targets are predicted. We compute these metrics on (1) the entire test set and (2) the subset of examples with 2–10 gold delegation targets.

Tables 15 and 16 shows the results. At the paragraph and item levels, Pipe-Bin-* outperform Naive-Top1 in SMF1 and PQCR while achieving comparable ESA when evaluated on the entire test set. These results further support the use of the proposed pipeline systems as effective baselines for the task. In contrast, performance on examples with 2–10 gold targets is substantially lower across all three set-level metrics than performance on the full test set, highlighting the intrinsic difficulty of recovering all delegation targets for these multi-label examples.

Model	Granularity	Precision	Recall	F1
-Single	Law	81.4 $\pm$ 4.75	80.2 $\pm$ 5.24	80.8 $\pm$ 4.63
	Article	68.4 $\pm$ 9.15	54.2 $\pm$ 4.94	60.4 $\pm$ 6.31
	Paragraph	63.0 $\pm$ 9.42	48.5 $\pm$ 5.97	54.7 $\pm$ 7.17
	Item	62.9 $\pm$ 9.44	48.5 $\pm$ 5.97	54.7 $\pm$ 7.18
Bin	Law	82.7 $\pm$ 7.15	77.6 $\pm$ 2.95	80.0 $\pm$ 4.92
	Article	70.5 $\pm$ 11.9	51.7 $\pm$ 2.29	59.4 $\pm$ 5.93
	Paragraph	64.6 $\pm$ 14.6	46.3 $\pm$ 4.16	53.4 $\pm$ 8.15
	Item	64.5 $\pm$ 14.7	46.3 $\pm$ 4.16	53.4 $\pm$ 8.19
-Multi2	Law	84.3 $\pm$ 3.95	78.3 $\pm$ 2.52	81.2 $\pm$ 2.78
	Article	72.8 $\pm$ 5.80	52.5 $\pm$ 4.12	60.9 $\pm$ 4.32
	Paragraph	68.7 $\pm$ 4.85	47.5 $\pm$ 3.63	56.1 $\pm$ 3.89
	Item	68.6 $\pm$ 4.85	47.5 $\pm$ 3.63	56.1 $\pm$ 3.90

Table 14: Performance of the pipeline systems using binary classifiers with threshold values selected on the development set.

Model	Granularity	ESA	SMF1	PQCR
Naive-Top1	Law	90.0 $\pm$ 0.73	91.4 $\pm$ 0.79	90.0 $\pm$ 0.73
	Article	70.6 $\pm$ 1.51	74.6 $\pm$ 1.68	70.6 $\pm$ 1.51
	Paragraph	51.0 $\pm$ 6.08	53.6 $\pm$ 6.33	51.0 $\pm$ 6.08
	Item	50.8 $\pm$ 6.11	53.4 $\pm$ 6.38	50.8 $\pm$ 6.11
-Single	Law	70.0 $\pm$ 5.82	77.4 $\pm$ 5.12	80.4 $\pm$ 4.65
	Article	51.1 $\pm$ 9.56	66.3 $\pm$ 7.46	71.0 $\pm$ 5.45
	Paragraph	47.6 $\pm$ 8.94	62.3 $\pm$ 7.48	68.2 $\pm$ 5.33
	Item	47.6 $\pm$ 8.94	62.3 $\pm$ 7.48	68.2 $\pm$ 5.33
Bin	Law	69.1 $\pm$ 9.47	74.2 $\pm$ 8.74	75.8 $\pm$ 8.20
	Article	54.1 $\pm$ 10.5	65.4 $\pm$ 9.60	67.3 $\pm$ 8.13
	Paragraph	47.8 $\pm$ 15.4	60.6 $\pm$ 12.1	64.7 $\pm$ 8.50
	Item	47.8 $\pm$ 15.4	60.5 $\pm$ 12.1	64.7 $\pm$ 8.49
-Multi2	Law	70.1 $\pm$ 5.20	76.3 $\pm$ 4.45	78.4 $\pm$ 4.77
	Article	54.5 $\pm$ 7.57	67.0 $\pm$ 4.97	69.7 $\pm$ 3.82
	Paragraph	50.8 $\pm$ 6.89	63.5 $\pm$ 4.73	67.3 $\pm$ 3.70
	Item	50.8 $\pm$ 6.89	63.5 $\pm$ 4.73	67.3 $\pm$ 3.70

Table 15: Prediction set level evaluation metrics of Naive-Top1 and the pipeline systems using binary classifiers, evaluated on the entire test set.

D Case Studies

To examine the effect of the introduction of a binary classifier, we analyze cases where a candidate retriever alone failed to assign the highest similarity scores to correct candidates, but the pipeline system with the binary classifier predicted delegation targets correctly.

Figure 13 shows a case where the top-1 candidate retrieved by Ret-Single was not a gold target, but Pipe-Bin-Single succeeded in producing a correct prediction. In this example, the delegation keyword is “Nuclear Regulation Authority Regulations,” and the correct delegation targets are Article 1-4 of the *Regulations on the Installation, Operation, etc. of Nuclear Reactors Used for Testing and Research* and Article 4-2 of the *Regulations*

Model	Granularity	ESA	SMF1	PQCR
Naive-Top1	Law	0.00 $\pm$ 0.00	46.6 $\pm$ 2.12	0.00 $\pm$ 0.00
	Article	0.00 $\pm$ 0.00	33.0 $\pm$ 2.37	0.00 $\pm$ 0.00
	Paragraph	0.00 $\pm$ 0.00	21.3 $\pm$ 2.90	0.00 $\pm$ 0.00
	Item	0.00 $\pm$ 0.00	20.9 $\pm$ 3.04	0.00 $\pm$ 0.00
-Single	Law	28.0 $\pm$ 3.93	58.3 $\pm$ 4.42	32.6 $\pm$ 4.57
	Article	12.0 $\pm$ 3.72	44.4 $\pm$ 5.59	20.5 $\pm$ 3.91
	Paragraph	8.64 $\pm$ 2.55	35.3 $\pm$ 5.33	16.3 $\pm$ 3.29
	Item	8.64 $\pm$ 2.54	35.3 $\pm$ 5.30	16.3 $\pm$ 3.27
Bin	Law	17.9 $\pm$ 5.91	48.7 $\pm$ 11.0	21.1 $\pm$ 7.04
	Article	11.4 $\pm$ 3.30	39.9 $\pm$ 8.31	15.3 $\pm$ 4.39
	Paragraph	7.29 $\pm$ 3.19	31.8 $\pm$ 8.11	12.8 $\pm$ 3.75
	Item	7.29 $\pm$ 3.19	31.8 $\pm$ 8.13	12.8 $\pm$ 3.75
-Multi2	Law	18.8 $\pm$ 3.17	53.8 $\pm$ 5.95	23.7 $\pm$ 3.68
	Article	10.9 $\pm$ 3.24	42.2 $\pm$ 5.61	16.2 $\pm$ 3.38
	Paragraph	7.85 $\pm$ 1.99	34.1 $\pm$ 4.83	13.5 $\pm$ 2.53
	Item	7.85 $\pm$ 1.99	34.1 $\pm$ 4.85	13.5 $\pm$ 2.54

Table 16: Prediction set level evaluation metrics of Naive-Top1 and the pipeline systems using binary classifiers, evaluated on the examples with 2–10 gold delegation targets.

on the *Installation, Operation, etc. of Nuclear Reactors Installed on Ships (Excluding Those at the Research and Development Stage)*. These *Regulations* themselves are “Nuclear Regulation Authority Regulations,” and “Article 25, item (iii) of the Act” appearing in them indicates the delegation source provision, which together allow the correct targets to be inferred. While Ret-Single retrieved the correct targets in the top-30 candidates by similarity scores, it failed to rank either of them as the top-1 candidate. The top-1 candidate retrieved by Ret-Single instead referred to “Article 15, item (iii) of the Act,” which belongs to the same Act as the true delegation source provision, but specifies a different article. On the other hand, employing the binary classifier, Bin-Single, to make an inference on the top-30 candidates, Pipe-Bin-Single successfully produced correct predictions, properly handling the references to the source provision. This case highlights a key advantage of the pipeline approach: even when candidate retrieval is imperfect, maintaining a sufficiently broad candidate set and applying a more expressive binary classifier can recover correct delegation targets.

Figure 14 illustrates a case where the top-2 candidates retrieved by Ret-Multi2 were not gold targets, but Pipe-Bin-Multi2 successfully made correct predictions. In this case, the delegation keyword is “Order of the competent ministry,” and the correct delegation targets are Article 12 of the *Order Concerning Foreign Exchange Transaction Analysis Service Providers* and Article 12 of the *Cabinet*

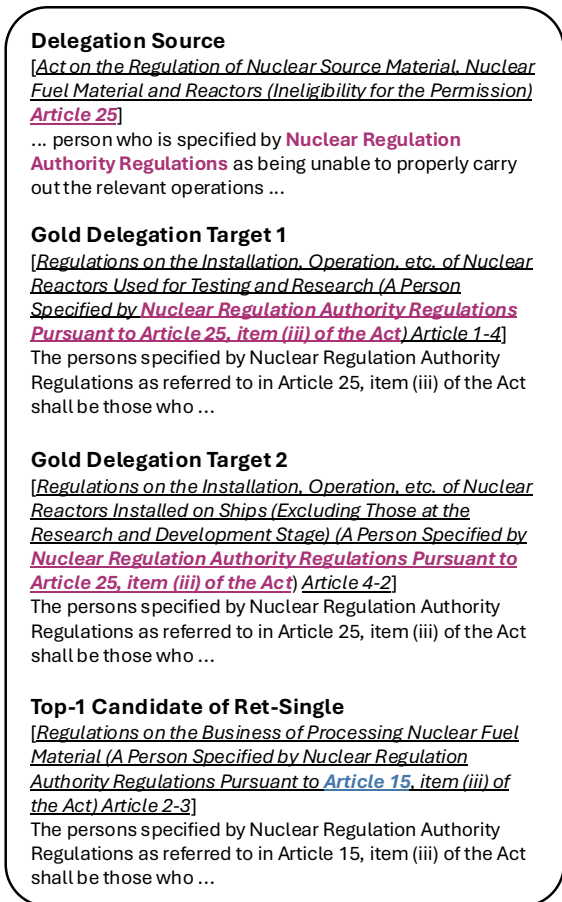


Figure 13: Example where the top-1 candidate retrieved by Ret-Single was incorrect, but Pipe-Bin-Single succeeded in producing a correct prediction

Office Order Concerning Foreign Exchange Transaction Analysis Service Providers. These *Order* and *Cabinet Office Order* are the Orders issued by the competent minister in the context of this example, and “Article 63-29 ... of the Act” appearing in them refer to the delegation source provision, which collectively allow the identification of the gold delegation targets. While Ret-Multi2 retrieved the correct targets in the top-30 candidates by similarity scores, it failed to rank either of them within the top-2 candidates. Instead, the top-2 candidates of Ret-Multi2 were the entirety of the correct Orders, which is an error in the identification of the granularity of the targets. On the other hand, combining Ret-Multi2 and Bin-Multi2, Pipe-Bin-Multi2 succeeded in identifying the correct targets, with Bin-Multi2 effectively dealing with the selection of granularity and recovering the gold targets from the list of candidates retrieved by Ret-Multi2. This example also represents the efficacy of the binary classifier in our approach.

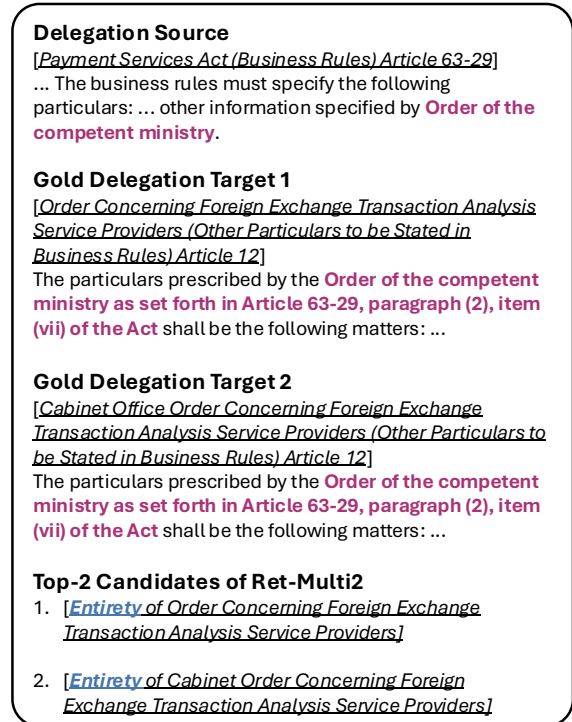


Figure 14: Example where the top-2 candidates retrieved by Ret-Multi2 were incorrect, but Pipe-Bin-Multi2 succeeded in producing correct predictions

Additionally, we also investigate a case where Bin-Single struggles with hard-to-distinguish candidates retrieved by Ret-Single, illustrated in Figure 15. Here, the gold delegation targets are Article 50 of the *Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions* and Article 39 of the *Order Concerning Special Measures for Strengthening Financial Functions of Labor Bank and Rokinren Bank*. Of the top-30 candidates retrieved by Ret-Single, 26 are provisions from the *Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions*. Many of these provisions are lexically and structurally similar to the correct target (Article 50), making them difficult to distinguish from the correct provision. As a result, Bin-Single incorrectly labeled two provisions from the same law as positive. This case illustrates one source of the lower precision of Pipe-Bin-Single compared to Pipe-Bin-Multi1 and Pipe-Bin-Multi2 observed in Section 4.

E Japanese Source Text and English Translation

Tables 17–19 provide the original Japanese legal texts and their corresponding English translations referenced in this study. The original Japanese texts

Delegation Source
[Order for Enforcement of the Act on Special Measures for Strengthening Financial Functions ... Article 18]
 ... Cooperative financial institutions that satisfy the criteria specified by the **Order of the competent ministry** ...

Gold Delegation Targets

- *[Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions ... Article 50]*
- *[Order Concerning Special Measures for Strengthening Financial Functions of Labor Bank and Rokinren Bank ... Article 39]*

Breakdown of Ret-Single top-30 Candidates by Laws

- *Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions → 26 provisions*
- *Order Concerning Special Measures for Strengthening Financial Functions of Labor Bank and Rokinren Bank → 3 provisions*
- *Standards for determining whether ... → 1 provision*

Prediction of Pipe-Bin-Single

- *[Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions ... Article 50]*
- *[Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions ... Article 49]*
- *[Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions ... Article 73]*
- *[Order Concerning Special Measures for Strengthening Financial Functions of Labor Bank and Rokinren Bank ... Article 39]*

Figure 15: Example where Pipe-Bin-Single produced multiple false positives

are retrieved from e-Gov Legislation Search (e-Gov 法令検索) maintained by the Digital Agency of Japan. The English translations were prepared by the authors with reference to the Japanese Law Translation Database System maintained by the Ministry of Justice of Japan. Where portions of the text were omitted for brevity, the symbol “...” is used to indicate such omissions. Indentation and other layout formatting are simplified.

Japanese	English
外国為替及び外国貿易法 (権限の委任) 第六十八条の二 主務大臣は、政令で定めるところにより、この法律に基づく権限の一部を地方支分部局の長に委任することができる。	Foreign Exchange and Foreign Trade Act (Delegation of Authority) Article 68-2 The competent minister may delegate part of their authority under this Act to the heads of local branch offices pursuant to the provisions of Cabinet Order.
外国為替令 (権限の委任) 第二十五条 次に掲げる財務大臣の権限は、税関長に委任する。...	Foreign Exchange Order (Delegation of Authority) Article 25 (1) The following parts of the authority of the Minister of Finance are delegated to the Directors-General of Custom-Houses: ...
輸出貿易管理令 (権限の委任) 第十二条 次に掲げる経済産業大臣の権限は、税関長に委任されるものとする。...	Export Trade Control Order (Delegation of Authority) Article 12 The following authority of the Minister of Economy, Trade and Industry is delegated to the Directors-General of Custom-Houses.
輸入貿易管理令 (権限の委任) 第十八条 次に掲げる経済産業大臣の権限は、税関長に委任されるものとする。...	Import Trade Control Order (Delegation of Authority) Article 18 The following authority of the Minister of Economy, Trade and Industry is delegated to the Directors-General of Custom-Houses.
国土調査法 (代位登記) 第三十二条の二 ...前項の登記の手続に関し必要な事項は、政令で定める。	National Land Survey Act (Subrogation Registration) Article 32-2 ... Cabinet Order provides for the necessary matters in connection with the registration process referred to in the preceding paragraph.
国土調査法による不動産登記に関する政令 (代位登記の登記識別情報) 第三条 ...	Cabinet Order on Real Property Registration Based on the National Land Survey Act (Information for Subrogation Registration Identification) Article 3 ...
国土調査法による不動産登記に関する政令 (不動産登記法等の適用) 第四条 ...	Cabinet Order on Real Property Registration Based on the National Land Survey Act (Application of Real Property Registration Act) Article 4 ...
陸上交通事業調整法 (勧告及び命令の実施) 第三条 ...協定成立セザルトキハ国土交通大臣ハ国土交通省令ノ定ムル所ニ依リ協議調ハザル事項ヲ裁定ス...	Land Transport Industries Regulation Act (Enforcement of Recommendations and Orders) Article 3 ... If an agreement is not concluded, the Minister of Land, Infrastructure, Transport and Tourism shall, in accordance with the provisions of the Order of the Ministry of Land, Infrastructure, Transport and Tourism, make a determination on the matters for which consultation has not been settled ...
陸上交通事業調整法施行規則 (その他の協定の認可申請書) 第八条 ...	Regulations for Enforcement of the Land Transport Industries Regulation Act (Application Documents for Approval of Other Agreements) Article 8 ...
陸上交通事業調整法施行規則 (裁定申請書) 第九条 ...	Regulations for Enforcement of the Land Transport Industries Regulation Act (Application Documents for Ruling) Article 9 ...
陸上交通事業調整法施行規則 (申請書類の提出手続) 第二十三条 ...	Regulations for Enforcement of the Land Transport Industries Regulation Act (Procedures for Submitting Application Documents) Article 23 ...

Table 17: Original Japanese legal texts and their English translations referenced in the main text

Japanese	English
核原料物質、核燃料物質及び原子炉の規制に関する法律 (許可の欠格条項) 第二十五条 次の各号のいずれかに該当する者には、第二十三条第一項又は第二十三条の二第一項の許可を与えない...その業務を適確に行うことができない者として原子力規制委員会規則で定める者...	Act on the Regulation of Nuclear Source Material, Nuclear Fuel Material and Reactors (Ineligibility for the Permission) Article 25 Permission under Article 23, paragraph (1) or Article 23-2, paragraph (1) shall not be granted to any person who falls under any of the following items ... person who is specified by Nuclear Regulation Authority Regulations as being unable to properly carry out the relevant operations ...
試験研究の用に供する原子炉等の設置、運転等に関する規則 (法第二十五条第三号の原子力規制委員会規則で定める者) 第一条の四 法第二十五条第三号の原子力規制委員会規則で定める者は、精神の機能の障害により、業務を適正に行うに当たつて必要な認知、判断及び意思疎通を適切に行うことができない者とする。	Regulations on the Installation, Operation, etc. of Nuclear Reactors Used for Testing and Research (A Person Specified by Nuclear Regulation Authority Regulations Pursuant to Article 25, item (iii) of the Act) Article 1-4 The persons specified by Nuclear Regulation Authority Regulations as referred to in Article 25, item (iii) of the Act shall be those who, due to a disorder of mental functions, are unable to appropriately carry out the cognition, judgment, and communication necessary for the proper performance of their duties.
船舶に設置する原子炉（研究開発段階にあるものを除く。）の設置、運転等に関する規則 (法第二十五条第三号の原子力規制委員会規則で定める者) 第四条の二 法第二十五条第三号の原子力規制委員会規則で定める者は、精神の機能の障害により、業務を適正に行うに当たつて必要な認知、判断及び意思疎通を適切に行うことができない者とする。	Regulations on the Installation, Operation, etc. of Nuclear Reactors Installed on Ships (Excluding Those at the Research and Development Stage) (A Person Specified by Nuclear Regulation Authority Regulations Pursuant to Article 25, item (iii) of the Act) Article 4-2 The persons specified by Nuclear Regulation Authority Regulations as referred to in Article 25, item (iii) of the Act shall be those who, due to a disorder of mental functions, are unable to appropriately carry out the cognition, judgment, and communication necessary for the proper performance of their duties.
核燃料物質の加工の事業に関する規則 (法第十五条第三号の原子力規制委員会規則で定める者) 第二条の三 法第十五条第三号の原子力規制委員会規則で定める者は、精神の機能の障害により、業務を適正に行うに当たつて必要な認知、判断及び意思疎通を適切に行うことができない者とする。	Regulations on the Business of Processing Nuclear Fuel Material (A Person Specified by Nuclear Regulation Authority Regulations Pursuant to Article 15, item (iii) of the Act) Article 2-3 The persons specified by Nuclear Regulation Authority Regulations as referred to in Article 15, item (iii) of the Act shall be those who, due to a disorder of mental functions, are unable to appropriately carry out the cognition, judgment, and communication necessary for the proper performance of their duties.
資金決済に関する法律 (業務方法書) 第六十三条の二十九 業務方法書には、次に掲げる事項を定めなければならない。...七その他主務省令で定める事項	Payment Services Act (Business Rules) Article 63-29 (2) The business rules must specify the following particulars: ... (vii) other information specified by Order of the competent ministry.
為替取引分析業者に関する命令 (業務方法書のその他の記載事項) 第十二条 法第六十三条の二十九第二項第七号に規定する主務省令で定める事項は、次に掲げる事項とする。...	Order Concerning Foreign Exchange Transaction Analysis Service Providers (Other Particulars to be Stated in Business Rules) Article 12 The particulars prescribed by the Order of the competent ministry as set forth in Article 63-29, paragraph (2), item (vii) of the Act shall be the following matters: ...
為替取引分析業者に関する内閣府令 (業務方法書のその他の記載事項) 第十二条 法第六十三条の二十九第二項第七号に規定する主務省令で定める事項は、次に掲げる事項とする。...	Cabinet Office Order Concerning Foreign Exchange Transaction Analysis Service Providers (Other Particulars to be Stated in Business Rules) Article 12 The particulars prescribed by the Order of the competent ministry as set forth in Article 63-29, paragraph (2), item (vii) of the Act shall be the following matters: ...

Table 18: Original Japanese legal texts and their English translations referenced in the Appendix D (Part 1)

Japanese	English
金融機能の強化のための特別措置に関する法律施行令 （法第十九条第一項の規定による承認に係る金融機関等の存続又は金融組織再編成が地域の経済にとって不可欠であると認められる場合） 第十八条 法第十九条第三項第四号ニに規定する政令で定める場合は、次の各号に掲げる変更後の経営強化計画を提出した計画提出金融機関等（同条第一項に規定する計画提出金融機関等をいう。以下この条において同じ。）の区分に応じ当該各号に定める場合とする。 …二協同組織金融機関 当該計画提出金融機関等が、その主として業務を行っている地域に密着した事業の展開を図っていると認められる協同組織金融機関として主務省令で定める基準に適合するものであり…	Order for Enforcement of the Act on Special Measures for Strengthening Financial Functions (Cases When the Continued Existence of, or the Financial Reorganization of, a Financial Institution, etc. Subject to Approval Under Article 19, Paragraph (1) of the Act is Deemed to be Indispensable to the Regional Economy) Article 18 The cases specified by Cabinet Order as prescribed in Article 19, paragraph (3), item (iv), (d) of the Act shall be the cases specified in the following items, according to the category of the plan-submitting financial institution, etc. (meaning a plan-submitting financial institution, etc. as prescribed in paragraph (1) of that Article; the same applies hereinafter in this Article) that has submitted a revised business strengthening plan set forth in the respective items. …(ii) Cooperative financial institutions: cooperative financial institutions that satisfy the criteria specified by the Order of the competent ministry as institutions recognized as seeking to develop businesses closely tied to the local community in the region where the relevant plan-submitting financial institution, etc. primarily conducts its business. …
金融機能の強化のための特別措置に関する内閣府令 （令第十八条第二号の主務省令で定める基準） 第五十条 令第十八条第二号に規定する主務省令で定める基準は、次に掲げる要件のいずれにも該当することとする。…	Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions (Criteria Specified by the Order of the Competent Ministry pursuant to Article 18, item (ii) of the Cabinet Order) Article 50 The criteria specified by the Order of the competent ministry, as prescribed in Article 18, item (ii) of the Cabinet Order, shall be that all of the following requirements are satisfied: …
労働金庫及び労働金庫連合会の金融機能の強化のための特別措置に関する命令 （令第十八条第二号の主務省令で定めるもの） 第三十九条 令第十八条第二号に規定する主務省令で定める基準は、次に掲げる要件のいずれにも該当することとする。…	Order Concerning Special Measures for Strengthening Financial Functions of Labor Bank and Rokinren Bank (Criteria Specified by the Order of the Competent Ministry pursuant to Article 18, item (ii) of the Cabinet Order) Article 39 The criteria specified by the Order of the competent ministry, as prescribed in Article 18, item (ii) of the Cabinet Order, shall be that all of the following requirements are satisfied: …
金融機能の強化のための特別措置に関する内閣府令 （法第十九条第三項第一号等の経営の改善の目標に関する基準） 第四十九条 …	Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions (Criteria Concerning Management Improvement Targets as pursuant to Article 19, paragraph (3), item (i), etc. of the Act) Article 49 …
金融機能の強化のための特別措置に関する内閣府令 （法第二十八条第一項第二号イの経営の改善の目標に関する基準） 第七十三条 …	Cabinet Office Order Concerning Special Measures for Strengthening Financial Functions (Criteria Concerning Management Improvement Targets as pursuant to Article 28, paragraph (1), item (ii), (a), etc. of the Act) Article 73 …

Table 19: Original Japanese legal texts and their English translations referenced in the Appendix D (Part 2)