big data and cognitive computing

MDPI

Article

Sentence-Level Rhetorical Role Labeling in Judicial Decisions

Gergely Márk Csányi [1,2,*], István Üveges [1,3], Dorina Lakatos [1], Dóra Ripszám [4], Kornélia Kozák [5], Dániel Nagy [1] and János Pál Vadász [1,4]

[1] MONTANA Knowledge Management Ltd., H-1029 Budapest, Hungary; uveges.istvan@montana.hu (I.Ü.); lakatos.dorina@montana.hu (D.L.); nagy.daniel@montana.hu (D.N.); vadasz.pal@uni-nke.hu (J.P.V.)
[2] Department of Electric Power Engineering, Budapest University of Technology and Economics, H-1111 Budapest, Hungary
[3] Political and Legal Text Mining & Artificial Intelligence Laboratory (poltextLAB), ELTE Centre for Social Sciences, H-1097 Budapest, Hungary
[4] UNESCO Chair on Digital Platforms for Learning Societies, Institute of the Information Society, Ludovika University of Public Service, H-1083 Budapest, Hungary; ripszam.dora@uni-nke.hu
[5] Department of European Public and Private Law, Faculty of Public Governance and International Studies, Ludovika University of Public Service, H-1083 Budapest, Hungary; kozak.kornelia@uni-nke.hu
* Correspondence: csanyi.gergely@montana.hu or csanyi.gergely@vet.bme.hu

Abstract

This paper presents an in-production Rhetorical Role Labeling (RRL) classifier developed for Hungarian judicial decisions. RRL is a sequential classification problem in Natural Language Processing, aiming to assign functional roles (such as facts, arguments, decision, etc.) to every segment or sentence in a legal document. The study was conducted on a human-annotated sentence-level RRL corpus and compares multiple neural architectures, including BiLSTM, attention-based networks, and a support vector machine as baseline. It further investigates the impact of late chunking during vectorization, in contrast to classical approaches. Results from tests on the labeled dataset and annotator agreement statistics are reported, and performance is analyzed across architecture types and embedding strategies. Contrary to recent findings in retrieval tasks, late chunking does not show consistent improvements for sentence-level RRL, suggesting that contextualization through chunk embeddings may introduce noise rather than useful context in Hungarian legal judgments. The work also discusses the unique structure and labeling challenges of Hungarian cases compared to international datasets and provides empirical insights for future legal NLP research in non-English court decisions.

Keywords: rhetorical role labeling; judicial decisions; sentence classification; late chunking

Academic Editor: Min Chen

Received: 1 October 2025
Revised: 25 November 2025
Accepted: 1 December 2025
Published: 5 December 2025

Citation: Csányi, G.M.; Üveges, I.; Lakatos, D.; Ripszám, D.; Kozák, K.; Nagy, D.; Vadász, J.P. Sentence-Level Rhetorical Role Labeling in Judicial Decisions. *Big Data Cogn. Comput.* **2025**, *9*, 315. https://doi.org/10.3390/bdcc9120315

1. Introduction

The digitization of legal systems has led to an exponential growth in unstructured text data, creating significant retrieval challenges for legal professionals. In the Hungarian context alone, the public repository of anonymized court decisions currently contains 235,310 documents. Navigating this massive corpus requires tools that go beyond simple keyword matching to understand the semantic structure of case law. By processing and structuring these documents, legal systems can enable advanced downstream tasks such as semantic search [1], legal case classification [2], and automatic summarization [3]. A fundamental step in structuring legal text is Rhetorical Role Labeling (RRL). RRL is a natural language processing task, which is a sequential classification problem, classifying each textual segment into its role in the given text. RRL has been in focus in the legal

domain in recent years [4–8]. This segmentation provides immediate practical value; for instance, by isolating the "Facts" of a case, a system can allow lawyers to search specifically for cases with similar factual backgrounds rather than just similar keywords, significantly facilitating their daily workflow [1]. While RRL has been a focus of the legal domain in recent years, current solutions are predominantly designed for English [4–8], leaving a gap for non-English jurisdictions with distinct legal traditions [9,10].

While these legal-system-specific factors require tailored approaches, technological advances in textual embeddings have also significantly improved the modeling side of RRL. Recent advances have introduced "late chunking," a technique where token-level representations are computed over a long context before chunking, preserving contextual information across boundaries. While late chunking has demonstrated consistent improvements in information retrieval tasks [11], its utility for sentence-level classification remains an open question. This study empirically tests whether the gains seen in retrieval transfer to the RRL task as well.

In this paper, we introduce an in-production RRL classifier for Hungarian judicial decisions. Our contributions are as follows:

- We introduce a human-annotated, sentence-level RRL corpus validated by legal experts, sharing Inter-Annotator Agreement (IAA) scores for tracking consistency.
- We compare multiple neural architectures, including BiLSTM and attention-based networks, against support vector machine baselines using two embedding strategies (huBERT CLS vs. Jina v3 with/without late chunking and with/without a positional feature).
- We provide a critical evaluation of late chunking, offering empirical evidence that, contrary to retrieval benchmarks, it does not yield consistent improvements for sentence-level RRL in legal texts.
- Finally, we present a system that moves beyond theory; the best-performing model is currently deployed at the National Office for the Judiciary, powering a Rhetorical Augmented Generation (RAG) pipeline that improves the searchability of court decisions.

This paper is structured as follows: Section 2 gives an overview of RRL in the legal domain. Section 3 describes the dataset, the labels and the annotator agreement, the used embeddings, and the classifier models. The results are presented in Section 4 and discussed in Section 5. The paper concludes in Section 6.

2. Relevant Works

Beyond machine learning-oriented approaches, RRL draws on discourse and rhetorical theory. Rhetorical Structure Theory (RST) formalized coherence via functional relations (nucleus/satellite) [12]. Follow-up surveys and applications [13,14] documented annotation challenges and showed the value of rhetorical relations for summarization, knowledge extraction, and text generation. These perspectives frame sentence-level role labeling within document-level coherence, motivating alignment of role categories with established discourse-analytic frameworks.

Related literature highlights linguistic indicators (especially cue phrases) as signals of rhetorical function. Early work linked coherence relations to lexical and syntactic correlates, showing connectives and discourse markers as reliable cues [15,16]. Teufel and Moens operationalized this in argumentative zoning for scientific articles, with annotation schemes and evidence that rhetorical status aids summarization [17]. In law, Hachey and Grover found rhetorical/topical cues improve extractive case-law summaries [18]. Together, these results support integrating cue phrases and rhetorical markers as complementary evidence

for role labeling in legal texts, bridging discourse theory with practical annotation and modeling.

RRL on long legal cases has steadily shifted toward methods that better capture local sequential dependencies and document structure. Prior to deep neural architectures, early work relied on handcrafted and rule-based systems. Saravanan et al. [19] proposed SLIPPER, a thematic segmentation method in legal tax texts based on expert-coded linguistic patterns, lacking scalable context modeling. Walker et al. [20] created manually labeled classifiers for U.S. judgments, focusing on small, script-driven samples without robust neural encoding. Sanchez et al. [21] used Conditional Random Field (CRF) and rule-based sentence boundary detection, segmenting text rather than modeling rhetorical functions. Kalamkar et al. [22] introduced a rhetorical role corpus and SciBERT-HSLN baselines, emphasizing document transfer but without chunk-aware or retrieval-based adaptations. These approaches typically operated on narrow features, small datasets, and lacked retrieval or chunking-aware design that drives current large-scale neural research.

Early work such as Bhattacharya et al. [4] pioneered neural sequence learning for sentence-level rhetorical role annotation in Indian Supreme Court judgments, demonstrating that BiLSTM models and deep architectures outperform CRF baselines and feature engineering. Their approach provided an initial human-annotated dataset and systematic inter-annotator agreement analysis, which laid the methodological foundation for subsequent neural sentence classification pipelines. Building on these first neural methods, Malik et al. [5] introduced a 13-role annotated corpus and multi-task deep learning models, systematically exploring transfer protocols, labeling granularity, and domain adaptation. This work enabled studies of domain robustness and transferability that shaped later directions in rhetorical segmentation and classification.

Hierarchical or curriculum-based training has recently shown measurable gains: Hierarchical Curriculum Learning for RRL (HiCuLR) integrates document- and role-level curricula to expose models from easy to hard instances while leveraging neighborhood context, improving performance on long judgments [6]. Large, newly released benchmarks (e.g., LegalSeg) similarly report that encoding broader local context and inter-sentence relations outperforms sentence-isolated approaches [8]. Earlier work on structural segmentation of legal documents motivates exploiting headings, paragraph boundaries, and rhetorical blocks during modeling [23].

Parallel to advances in sequence modeling, long-context embedding models and chunking strategies have been explored to improve retrieval and representation of lengthy texts. The late chunking technique proposes to first compute token-level representations with a long-context encoder and only then perform chunking prior to pooling, thereby yielding chunk vectors that retain document-level context. Results of late chunking show consistent gains in retrieval tasks across datasets and architectures [11]. Our study does not assume these gains transfer to sentence-level RRL, where the prediction target is a single sentence's rhetorical role rather than passage relevance. Instead, we treat late chunking as a testable hypothesis for RRL, asking under what conditions, if any, contextualized chunk embeddings can help model sentence-level roles.

On the multilingual front, models such as LEGAL-BERT have demonstrated that context-aware RRL is feasible across major European languages, including Italian and English legal cases [9]. Alongside these, alternative segmentation pipelines, such as named entity recognition-driven segmentation, offer complementary perspectives for discovering rhetorical structure in legal texts [24].

More broadly, new multilingual, multi-function, multi-granularity models (e.g., M3-Embedding) now explicitly support inputs ranging from sentences to documents of 8k tokens while targeting dense, multi-vector, and sparse retrieval in a unified framework [25].

Recent multilingual evaluations have confirmed that long-context embeddings can generalize successfully across diverse court decisions and legal traditions [26]. Similarly, `jinaai/jina-embeddings-v3` introduces long-context multilingual embeddings with task-specific adapters for retrieval, clustering, classification, and matching [27]. These models make it technically feasible to test context-aware chunking in domains beyond retrieval, including sentence-level classification on long legal documents. Recent RRL-focused architectures (e.g., MARRO) further indicate that attention mechanisms and auxiliary tasks can contribute to stronger role-aware sentence representations [7].

Concurrently, the Retrieval-Augmented Generation (RAG) literature has begun to argue that "one-size-fits-all" chunking is suboptimal and that granularity should be query- or task-adaptive. Mix-of-Granularity (MoG) and its graph extension (MoGG) route queries to different chunk scales and show improvements by dynamically selecting chunk size [28]. Broader retrieval studies likewise find that enhancing retrieval through better chunking, expansion, and re-ranking can have larger impact than tuning the generator alone [29]. These findings suggest that chunking benefits are highly task-dependent, reinforcing the need to examine late chunking specifically for RRL, rather than assuming transfer from retrieval benchmarks.

3. Materials and Methods

3.1. Dataset

Our goal was to provide rhetorical role labels for all of the currently publicly available 235,310 anonymized Hungarian court decisions [30]. There are five major areas of law in the Hungarian court decisions, namely: administrative, civil, economic, labor, and criminal, counting military criminal cases as part of the criminal area of law.

The structures of legal cases are usually quite similar to each other: they open with information about the two parties, the court, prior courts and case numbers, and the subject matter, then they provide a short description of the decision. After that, usually, the facts and the ruling(s) of previous court(s) are described, next the arguments of each side, then the decision again, and finally the ratio of the decision. Hungarian courts structured these documents differently before the year 2016, not providing clear section titles in the documents, and after 2016, only the Supreme Court was obliged to provide such descriptive section titles, slowly followed by lower courts as well. From now on, we refer to the documents not containing these descriptive titles as **old-type** and the ones that contain this as **new-type**.

Since an RRL dataset for Hungarian did not exist, involving human annotators was necessary. However, human annotation is costly and time-consuming. Therefore, we asked annotators to focus on "old-type" data, where easily identifiable section titles were absent. New-type documents with descriptive titles can be labeled automatically using simple heuristics.

We performed the automatic labeling as follows:

- The first step was filtering for documents newer than 2016 and collecting the section titles by keyword/phrase matching in sentences.
- A list of candidate section titles was created for each label.
- While iterating through the document, if a sentence matched one of the titles from the list of feasible section titles, all subsequent sentences up to the next section title were assigned the corresponding label.
- The Cost of the Case and Decision of the Court labels were identified using regular expressions.
- Finally, we only retained those documents that contained all the required section titles for our labeling scheme. This filtering step reduced the chance of mislabeling by

ensuring that each label could be assigned based on the presence of its corresponding section title in the document.

The automatic labeling procedure generally produced reliable results, but it was applicable primarily to Supreme Court documents. In total, around 10,000 documents were labeled automatically. For our experiments, however, we used only about 30 documents per area of law, in order to avoid biasing the models toward relying too heavily on the presence of descriptive titles. Each of the documents was validated by a legal expert. As guidelines for the annotation process, the annotators were given the list of the available labels (described in Section 3.3) that are easily understandable for a legal expert, and the annotators were instructed to give only one label for each sentence, selecting the most appropriate if more labels could apply. Finally, we managed to create a training (+dev) set containing 299 documents and a test set containing 120 documents.

The distribution of areas of law in the training and testing datasets was different, as Table 1 shows. During training, the areas of law were almost equally distributed, counting the Military criminal cases as criminal cases, while the test dataset followed the distribution of the whole (ca. 235k) dataset to be able to estimate real performance in a production environment. The training set contained almost equal numbers of old and new-type documents for each area of law, except for the administrative and military criminal areas. As discussed earlier, the old-type documents were particularly important for evaluation because, unlike the new-type documents, they did not contain easily identifiable section titles that could artificially simplify the labeling task.

Table 1. Number of documents in the training, validation, and test datasets by areas of law.

Area of Law	Train + Validation Data			Test Data			% of the Dataset	
	Old Type	New Type	Sum	Old Type	New Type	Sum	Test	Whole
Administrative	49	24	73	21	3	24	20.00	20.19
Civil	30	30	60	42	6	48	40.00	41.22
Criminal	27	26	53	13	3	16	13.33	15.83
Economic	28	28	56	12	3	15	12.50	12.41
Labor	26	26	52	10	4	14	11.67	8.34
Military criminal	5	0	5	3	0	3	2.50	2.01
All	165	134	299	101	19	120	100	100

3.2. Sentence Splitting

Sentence splitting in legal documents is challenging since legal documents contain significantly more dot characters compared to corpora from other domains. These include different legal or statutory references (e.g. II. Pfv.35.125/2010/4, BH 2010.21, KGD 2013.2345, 32/2008. (VII. 19.) IM rendelet 8. § etc.) and as a result of anonymization, monograms and three dots are also present. The sentences were split using a heuristic splitter specifically modified for judicial documents described in [31]. This sentence splitter could achieve state-of-the-art results on legal documents and could perform significantly better compared to the transformer-based solution provided by the Hungarian version of HuSpaCy [32], while also being significantly faster.

3.3. Labels

As in many previous studies [4,9,10], we have also decided to classify the documents at the sentence level. We classified the sentences into one of the following eight labels:

- **Facts (FAC):** description of the events leading to the case.
- **Case History (CH):** sentences referring to decisions of the lower courts.
- **Arguments of the parties (ARG):** arguments from both the plaintiff and defendant sides.
- **Decision of the Court (DEC):** the declaration for which side the court ruled. Usually one or two sentences.
- **Discussion of the Decision (DISC):** discussing the reasons and rationale of the decision in detail.
- **Cost of the Case (COST):** The costs of the case decided by the Court.
- **Operative Part of the Judgment (OP):** this is a bigger section in the cases appearing after the header and before the justification. In this section, the decision of the court and the costs are usually also mentioned, labeled accordingly, and any remaining sentences were labeled as OP.
- **OUT:** sentences that do not belong to any of the categories mentioned above, e.g., header and footer of the case, section titles, date, signatures, etc.

Due to differences in the structures of the Hungarian cases, the labels did not match the labels mentioned in other studies. For instance, Malik et al. [5] also used the Arguments of the Parties label, but further split it into Argument-Petitioner and Argument-Respondent labels. In contrast, we did not include Statutes, Dissent and Precedent (also split into three sub-labels: Relied and Not relied on and Overruled) because recent reforms have moved Hungarian law toward precedent and dissenting opinions are issued only by the Constitutional Court of Hungary; they do not appear in the anonymized ordinary court decisions we analyze in this study. Our labels are closer to the label set presented by Bhattacharya et al. [4], but we also used Cost of the Case and OUT labels while not using any Statutory label.

The distribution of labels, the corresponding number of sentences, their ratio in the dataset, and the number of tokens per sentence are shown in Table 2. Token counts were calculated using the `jina-embeddings-v3` huggingface model.

Table 2. Distribution of labels, average token per sentence.

Label	Train + Validation Data			Test Data		
	Nr. of Sentences	Ratio	Token/Sentence	Nr. of Sentences	Ratio	Token/Sentence
Arguments of the Parties	6639	0.1973	52.56	3198	0.1811	49.71
Case History	3735	0.1110	57.44	1653	0.0936	56.13
Cost of the Case	967	0.0287	57.08	450	0.0255	59.14
Decision of the Court	1657	0.0492	40.15	710	0.0402	47.70
Discussion of the Decision	9864	0.2931	57.16	5899	0.3340	54.11
Facts	4832	0.1436	49.32	4020	0.2276	49.52
Operative Part	441	0.0131	62.93	352	0.0199	55.45
OUT	5518	0.1640	11.38	1380	0.0781	17.14

3.4. Annotator Agreement

To validate our label system, 10 documents were selected to measure Inter-Annotator Agreement (IAA), each labeled independently by two legal expert annotators. The 10 documents contained 2734 sentences altogether. We selected the average agreement (accuracy) and Krippendorff's alpha score [33] measures for comparison and calculated them at the sentence level. Krippendorff's alpha is a widely used metric for measuring IAA since it works with multiple annotators, missing data points, and corrects for chance, while it is robust under imbalance [33,34]. The scores are presented in Table 3.

Table 3. Agreement between annotators.

Metric	Score
Avg. Agreement	0.8706
Krippendorff's Alpha	0.7777

All metrics show good inter-annotator agreement. We have also investigated these metrics label-wise. For this, the labels were binarized before the calculation of agreement metrics. The results are shown in Table 4.

Table 4. Label-wise inter-annotator agreement.

Label	Krippendorff's Alpha	Avg. Agreement
Arguments of the Parties	0.9479	0.9832
Case History	0.6768	0.9203
Cost of the Case	0.9283	0.9971
Decision of the Court	0.4243	0.9693
Discussion of the Decision	0.7568	0.8815
Facts	0.3899	0.8175
Operative Part	0.8066	0.9942
OUT	0.6998	0.9682

Six out of eight labels show good agreement (above 0.67 Krippendorff's alpha score) (Arguments of the Parties, Case History, Cost of the Case, Discussion of the Decision, Operative Part, OUT). Two labels reached only moderate agreement (around 0.4). Upon manual inspection of the annotated documents, we identified one case that largely explained the low agreement: it contained a 70-sentence-long list of evidence, which the two legal experts had labeled differently. The other disagreement was in a section that belonged to the Discussion of the Decision unit but contained many sentences discussing the Facts. This unit consisted of 273 sentences. The difference here was again the result of two different decisions by the annotators, resulting in a major disagreement in scores. However, the annotators agreed afterwards that both labels would have been acceptable.

3.5. Vectorization

3.5.1. Jina Embeddings with and Without Late Chunking

Generally, embedding models map a given text into a fixed-size vector representation. For longer texts, it is often necessary to split the input into smaller chunks so that each fits within the model's context window. However, embedding these chunks independently reduces the available document-level context, since relationships across chunk boundaries are no longer captured. A common mitigation strategy is to overlap some tokens between adjacent chunks [1,35], but this only partially preserves contextual information and does not fully resolve the problem.

Günther et al. [11] from JinaAI came up with an excellent yet simple idea of late chunking. The concept can be seen in Figure 1.

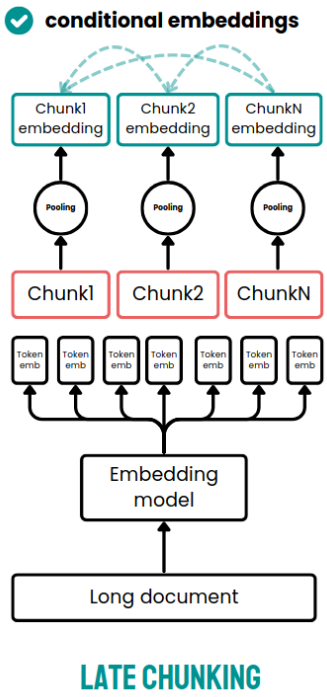


Figure 1. Late chunking [11].

During late chunking, the whole text is given to the embedder model and the token-level embeddings are calculated. After the calculation of token-level embeddings, the chunks are formed and the chunk-level embeddings are calculated by pooling the token-level embeddings. This way, it is ensured that the context between chunks is captured in the embeddings. To calculate this, the embedder model should return token-level embeddings as well and cover a relatively bigger number of tokens.

Currently, there are embedding models that can cover up to 8192 tokens. Some examples would be OpenAI's `text-embedding-3-large` model, Gemini's `gemini-embedding-001`, Beijing Academy of Artificial Intelligence's BGE-M3 [25], and JinaAI's `jina-embeddings-v3` [27] and `jina-embeddings-v4` models [36]. Nevertheless, from the above-mentioned models, only the Jina and BGE embeddings can return token-level embeddings, while Jina also provides late chunking behavior in its API by setting the late chunking flag.

Since we wanted to embed sentences that are contextually highly related, we chose the `jina-embeddings-v3` model with a `classification` task setting to embed our sentences, since use of the API late chunking is also provided. The model supports Hungarian language, although it was not fine-tuned specifically for tasks in Hungarian [37]. This model can cover 8192 tokens and returns 1024 dimensional vectors. We vectorized both with and without the late chunking setting.

3.5.2. BERT CLS

As a second embedding model, we used the classification (CLS) token representation from the Hungarian BERT-base variant `SZTAKI-HLT/hubert-base-cc` model [38]. Since it was pre-trained exclusively on Hungarian data, including legal documents, it provided some advantage for our task. The model has a maximum context window of 512 tokens and produces CLS embeddings of 768 dimensions. We did not fine-tune the model.

Sentences longer than the context window were split into maximum of 512 token-wide chunks, splitting the text only on word borders without any overlapping between the chunks. For each of these chunks, the BERT CLS embedding was calculated. The final embedding for the sentence was the average of these embeddings of the split data except

for the last chunk, where the chunk's vector was scaled by the ratio of the number of tokens and the context window, similarly to [1]. The process is shown in Figure 2.

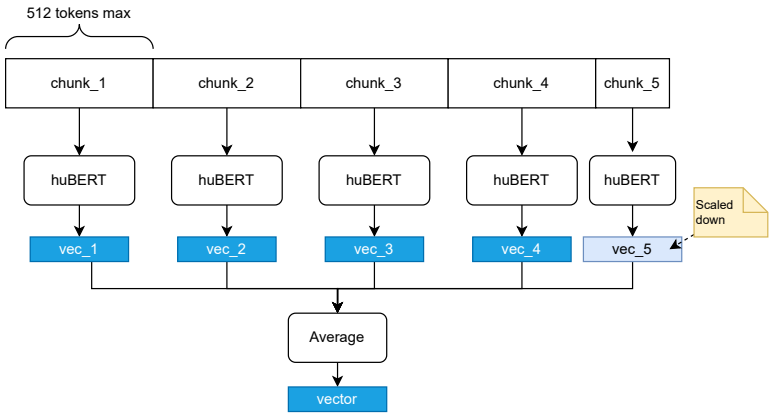


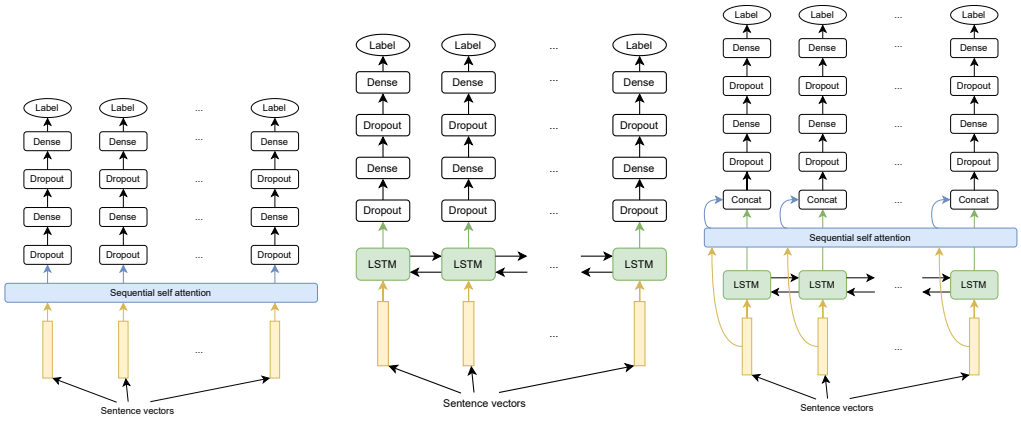
Figure 2. Handling long sentences with BERT CLS vectorization.

3.5.3. Position Feature

When the sentences are embedded and no late chunking is applied, the position of the sentence in the document is completely lost alongside the context. Hence, we calculated the Position feature, which is the relative position of the given sentence in the document, which was a value between 0 and 1. This value was appended to the already available embeddings.

3.6. Models Used for Classification

The RRL task is a sequential classification task, and the sentences are not independent of one another in meaning. As a consequence, the same applies to the labels. To validate this, we have tried SVMs, which cannot harness this dependence, as baseline and neural models that are good for sequential labeling: Bidirectional Long-Short Term Memory (BiLSTM) [39], attention [40] and BiLSTM+attention networks. The structure of each architecture can be seen in Figure 3.



(**a**) Attention model (**b**) BiLSTM model (**c**) BiLSTM+attention model

Figure 3. Neural architectures for rhetorical role labeling.

Earlier studies have shown that using a CRF layer in a sequence classification task would be beneficial [41]. However, this can be counterproductive when the labels are not following each other as label blocks but can fluctuate between two labels depending on the meaning of the sentences, and this can happen in old-type documents. That is why we decided to omit the addition of a CRF layer. In all architectures, each sentence

is first mapped to a fixed vector (e.g., BERT/Jina embedding). These embeddings are fed into the sequential neural architectures. Each sequential output vector was fed to stacked Dense and Dropout layers, with a per-sentence softmax with eight neurons as the classification layer.

In the attention architecture (shown in Figure 3a), the self-attention layer attends over the entire sentence sequence to produce context-aware representations. Self-attention allows every sentence to condition on any other, which is crucial for roles whose identity depends on distant cues. Moreover, the model can learn position-dependent patterns such as the tendency of Facts to appear early and Discussion late.

The BiLSTM architecture (shown in Figure 3b) produces context-enriched states. BiLSTM summarizes past and future sentence context that can encode typical narrative flow of judgments. The network can struggle on very long sequences.

The BiLSTM+attention architecture has complementary strengths: the LSTM provides order-aware representations and smooths local transitions, while the subsequent attention bridges long distances (e.g., linking a concluding dispositive formula to earlier legal grounds).

For implementing the neural architectures, we used the `keras` framework (version 2.15.0). For sequential self attention, the `keras-self-attention` (version 0.51.0) package was used. `Scikit-learn` (version 1.6.1) [42] was used for the cross-validation and the linear SVM classifier.

4. Results

4.1. Baselines

As a baseline, we classified the sentence vectors using a SVM with linear kernel. We fine-tuned the C parameter that was best at C = 1 setting and also tried the `class_weight="balanced"` setting since we had an imbalanced dataset. This classifier is not able to harness the information from the sequential characteristics, except when using the Jina late chunking embeddings. We performed 5-fold stratified cross-validation in the training+validation set with 299 documents and tested the Jina and BERT CLS embeddings with and without position features and the `class_weight="balanced"` setting. As a metric, macro F1 average was chosen. The results are shown in Figure 4.

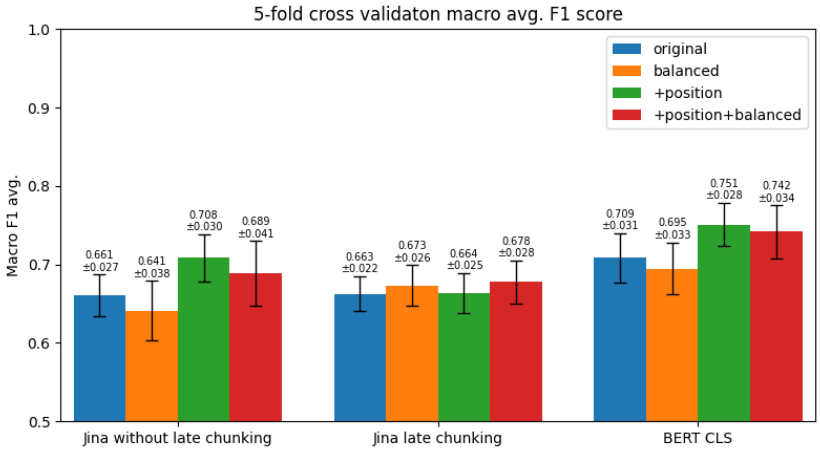


Figure 4. Comparing different vector forms and training parameters: `position`: relative position added as feature, `balanced`: training with balanced class weights.

The embeddings without late chunking (Jina without late chunking and BERT CLS) followed a similar pattern: using the `class_weight="balanced"` setting decreased the performance with and without adding the position feature. In these cases, the addition

of the position feature was also beneficial, raising the macro F1 average by 4.76% (from 66.07% to 70.83%) and 4.25% (from 70.85% to 75.1%), respectively.

In contrast, in case of late chunking, the `class_weight="balanced"` setting increased the results to some extent, and adding the position feature also had positive but marginal effects; meanwhile, the best late chunking result remained below the best result without late chunking (70.83% vs. 67.77%).

The results show that there is a significant number of sentences that can be classified correctly without fully harnessing any additional sequential information. However, the results proved that the sequential information is important since by adding the positional feature, the results improved significantly. Hence, the main room for improvement points towards neural structures that are capable of capturing the sequential information.

4.2. Neural Models

Since the results with the linear SVM suggested that the Jina embeddings may not perform well in our RRL task, we compared the embeddings using only BiLSTM models. Note that during training, the train data was split into 85% training set, 15% validation set using a stratified split by area of law and by old vs. new data type as well. During training, in each epoch, the documents (but not the sentences inside the documents) were shuffled, and categorical cross entropy loss was used. The models were fine-tuned using the validation set, trying to minimize both bias and variance errors. Table 5 shows the parameters used during training.

Table 5. Parameters used during BiLSTM training.

Parameter	Value
Epochs	max. 50
Learning rate	0.001
Dropout	0.2
Recurrent dropout	0.2
Batch size	32
LSTM cells	128
Distributed dense	32
Early stopping on	validation loss
Early stopping patience	5
Optimizer	AdamW
Halving learning rate	after every 10 epochs
Attention window	10

We compared the models using the test set since we wanted to obtain performance estimations for the whole dataset. The evaluation was done using the following metrics:

- Document Average Accuracy (DAA), which is the average of accuracies calculated at the document level,
- Accuracy at sentence level,
- Macro F1 average among labels averaged at document level,
- and the Weighted F1 average averaged at document level.

The results for the best SVM baselines (BERT CLS and BERT CLS+pos) are also shown. The results can be seen in Figure 5.

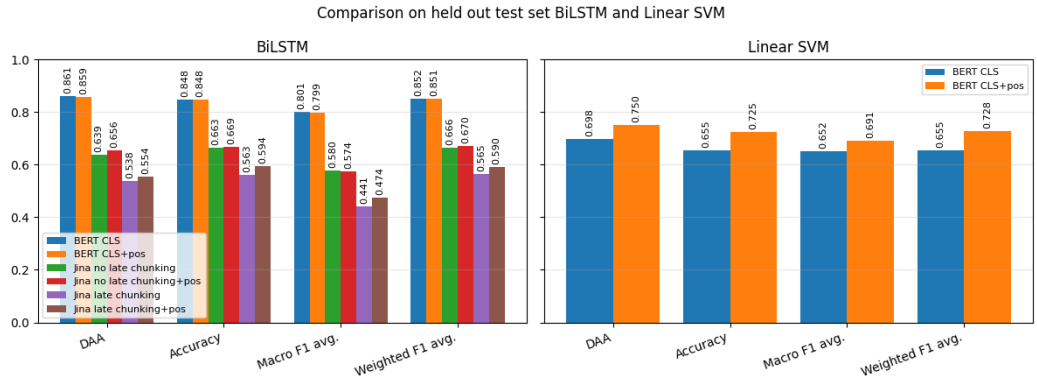


Figure 5. Comparing different embedding using BiLSTM and Linear SVM as baseline.

It can be seen that late chunking had the same effect on performance as in the SVMs: the best results could be achieved by the BERT CLS embeddings, then the Jina without late chunking and Jina with late chunking embeddings. Adding the positional features resulted in negligible effect for BERT CLS and Jina when no late chunking embeddings were used, while minor gains were measured with the Jina late chunking setup.

When compared to the baseline results calculated using linear SVMs, it was possible to beat the baselines by 11.1% in DAA, 12.3% in accuracy, 11% in macro F1 avg., and 12.4% in weighted F1 avg. This was expected since the information from the surrounding sentences is gathered by the BiLSTM network, and this is required in this task for better performance. Surprisingly, none of the Jina embeddings could perform better than the best baseline model. This was particularly surprising in the case of the Jina no late chunking setting since the BiLSTMs are capable of capturing the relevant information from the surrounding sentences, and the Jina embeddings are capable of providing meaningful interpretations for Hungarian texts for retrieval settings, as shown in [1].

Since we measured significant inferiority of the Jina embeddings in our RRL task, and our main goal was to train a model for deployment, we decided to compare the neural architectures using only the BERT CLS and BERT CLS+pos embeddings.

4.3. Comparing Neural Models

We trained three neural models using the BERT CLS and BERT CLS+pos setups, with the settings shown in Table 6, while keeping the 85%–15% train–validation split during training. Each training was performed three times, setting three different random states for the train–validation splits, using a stratified split by area of law and by old vs. new data type as well.

Table 6. Parameters used during training.

Parameter	Value
Epochs	max 200
Learning rate	0.001
Dropout	0.4
Recurrent dropout	0.4
Batch size	32
LSTM cells	128
Distributed dense	32

Table 6. *Cont.*

Parameter	Value
Early stopping on	validation loss
Early stopping patience	10
Optimizer	AdamW
Halving learning rate	no
Attention window	10

For the main comparison metric, we selected the macro F1 average and weighted F1 average metrics, averaged on a document level, and the DAA and accuracy scores as a secondary metric. The results on the test set are shown in Figure 6.

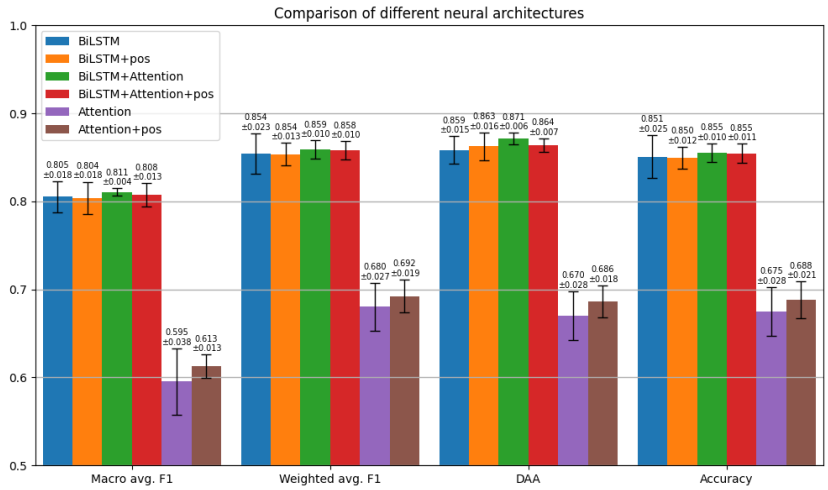


Figure 6. Neural architecture results using BERT CLS and BERT CLS + positional features.

The results suggest that the attention network is capable of learning the regularities of a legal case, but only to a limited extent. The addition of the position feature slightly improved the performance across all metrics. Meanwhile, the BiLSTM and BiLSTM+attention networks captured these regularities significantly better. This is illustrated by the fact that the effect of adding the positional feature was negligible in both architectures.

Hence, we compared only the BiLSTM and BiLSTM+attention networks (as the best candidates for high performance) on the test set.

4.4. Results on the Test Set

Our training and validation dataset has a different distribution of areas of law than the test set. The test set was created in a way to give an estimation for the performance on the entire corpus; hence, the different areas of law do not appear homogeneously. We retrained the BiLSTM and BiLSTM+attention networks on the joined training and validation sets using the same setups shown in the previous section. The resulting models were then evaluated on the test set. The corresponding evaluation metrics can be seen in Table 7.

The results showed that the BERT CLS+BiLSTM setup worked the best, although all setups performed relatively similarly, all showing very good performance on the RRL task.

The results of the models on different areas of law are shown in Table 8.

Table 7. Results on the test set.

Embedding	Neural Model	DAA	Accuracy	Macro F1	Weighted F1
BERT CLS	BiLSTM	0.9226	0.9247	0.8849	0.9252
BERT CLS+ pos	BiLSTM	0.8926	0.8828	0.8356	0.8853
BERT CLS	BiLSTM+attention	0.8806	0.8668	0.8209	0.8690
BERT CLS+pos	BiLSTM+attention	0.8964	0.8731	0.8317	0.8751

Table 8. Results across areas of law (**bold**: the best result in the given metric in the given area of law).

Domain	Embedding	Neural Model	Macro avg. F1	Weighted avg. F1	DAA	Accuracy
Administrative	BERT CLS	BiLSTM	**0.8300**	**0.8688**	**0.8972**	**0.8664**
	BERT CLS+ pos	BiLSTM	0.8075	0.8474	0.8757	0.8443
	BERT CLS	BiLSTM+attention	0.7763	0.8300	0.8523	0.8250
	BERT CLS+pos	BiLSTM+attention	0.7833	0.8376	0.8654	0.8330
Civil	BERT CLS	BiLSTM	**0.9034**	**0.9422**	**0.9408**	**0.9416**
	BERT CLS+pos	BiLSTM	0.8694	0.9245	0.9194	0.9220
	BERT CLS	BiLSTM+attention	0.8450	0.9032	0.9060	0.9002
	BERT CLS+pos	BiLSTM+attention	0.8867	0.9382	0.9322	0.9366
Criminal	BERT CLS	BiLSTM	**0.8326**	**0.9001**	**0.8913**	**0.9018**
	BERT CLS+pos	BiLSTM	0.7182	0.8094	0.8102	0.8082
	BERT CLS	BiLSTM+attention	0.7353	0.7928	0.8037	0.7921
	BERT CLS+pos	BiLSTM+attention	0.6697	0.7540	0.8147	0.7505
Economic	BERT CLS	BiLSTM	**0.9017**	**0.9509**	**0.9176**	**0.9503**
	BERT CLS+pos	BiLSTM	0.8346	0.9004	0.8941	0.8997
	BERT CLS	BiLSTM+attention	0.8315	0.8885	0.8882	0.8910
	BERT CLS+pos	BiLSTM+attention	0.8460	0.9002	0.8941	0.9005
Labor	BERT CLS	BiLSTM	**0.9251**	**0.9736**	**0.9563**	**0.9737**
	BERT CLS+pos	BiLSTM	0.9026	0.9604	0.9501	0.9594
	BERT CLS	BiLSTM+attention	0.9018	0.9565	0.9489	0.9555
	BERT CLS+pos	BiLSTM+attention	0.9129	0.9685	0.9494	0.9682

According to the evaluation, the BiLSTM with BERT CLS embeddings outperformed all other models across every legal domain and metric. The BiLSTM+attention with BERT CLS+pos embeddings followed closely, achieving strong results in four of the five domains. Yet, in the criminal law domain, disregarding the DAA metric, the model performed less effectively. Criminal cases stand out from other legal areas because they are generally longer, frequently involve several defendants or plaintiffs, and their operative parts are considerably more extensive. In many instances, they also include exhaustive evidence lists, which further complicates the texts.

4.5. Label-Level Results

The label-level results can be seen in Table 9.

Table 9. Label-level results of the best-performing model.

Label	Precision	Recall	F1
Arguments of the Parties	0.9073	0.9407	0.9237
Case History	0.9458	0.9178	0.9316
Cost of the Case	0.9389	0.9862	0.9620

Table 9. *Cont.*

Label	Precision	Recall	F1
Decision of the Court	0.8341	0.8066	0.8201
Discussion of the Decision	0.9747	0.9425	0.9583
Facts	0.8823	0.9432	0.9117
Operative Part	0.6144	0.6676	0.6399
OUT	0.9939	0.8768	0.9317

The model performed well across the majority of the labels: only two labels performed worse than 0.9 F1 score, namely the Operative Part (OP) (0.6399) and the Decision of the Court (DEC) (0.8201) labels. The confusion matrix can be seen in Figure 7.

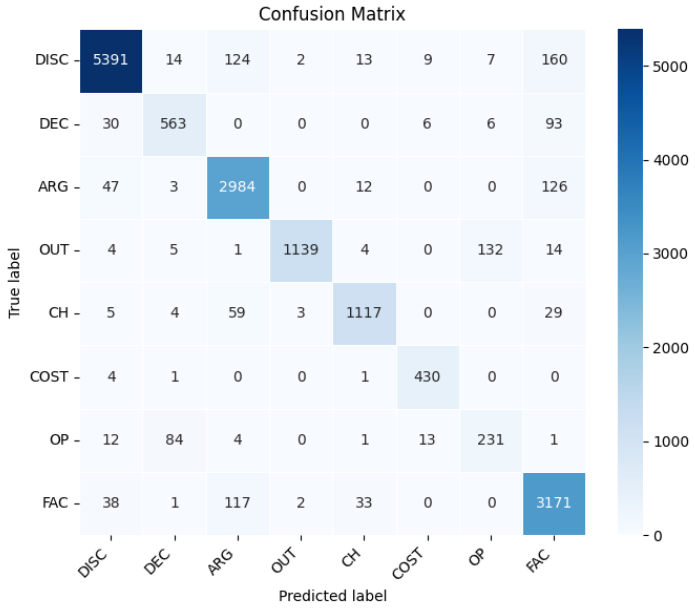


Figure 7. Confusion matrix of the best model.

The confusion matrix shows that the OP labels were predicted mainly as DEC labels (84 sentences), and many OUT labels were predicted as OP (132 sentences), explaining the lower F1 results on the OP label. The moderate F1 score of the DEC label was mainly because it was predicted as Facts (FAC) in 93 sentences, and in 84 sentences, OP sentences were predicted as DEC. It is important to point out that from a practical application perspective, correct predictions in the case of OP and OUT labels are less crucial since filtering for these labels is not supported in the deployed application, so the best model was selected for deployment. In our system, filtering for these labels is unnecessary because the OUT class consists of section titles, headers and footers, dates, and signatures. Although these segments may contain metadata (e.g., judges' names), such metadata are extracted separately and made available for filtering. The OP category is similar to OUT but is confined to the section between the header and the justification. This section contains COST and DEC sentences, and the remaining sentences (often none) within it were labeled as OP.

5. Discussion

5.1. Late Chunking Hurts Performance

The most surprising finding was that late chunking did not improve the results significantly compared to the non-late chunking setting and proved to be worse than a

non-finetuned BERT CLS embedding setting both using classifiers aware and unaware of sentence order using stratified data split by area of law and by old vs. new data type as well. The reason for this being surprising is that vectorizing with late chunking provides context-aware embeddings, as shown in [11]. This advantage can be exploited, for instance, in the retrieval phase of Retrieval Augmented Generation (RAG) processes, where relevant information may spread through chunks. Our case is quite similar: a sentence of a legal case usually cannot be categorized by itself, but the context of the surrounding sentences is also important. Günther et al. [11] showed that in many databases, late chunking resulted in greater improvement in shorter chunks than in longer ones. Sentences in our corpus were relatively short, averaging around 50 tokens per sentence, so a relevant gain in performance was awaited. In contrast, late chunking proved to be the least effective embedding setup. Even a simple, non-finetuned but "native" Hungarian BERT CLS embedding performed better, both with traditional machine learning classifiers and with neural models. Although categorization benefits from the context of the sentence, the results suggest that capturing the context into each embedding before classification (in other words, late chunking) significantly hurts the performance of the RRL task. In a recent study, Merola and Singh [43] showed that in a Q&A task, late chunking significantly decreased or at most marginally improved the performance of the retrieval. Despite their different task setting (Q&A vs. RRL), their results also suggest that the application of late chunking is not always beneficial. This emphasizes that in an RRL task, information from surrounding sentences is less important than the flow across the sentences themselves, which carries the key information for classification.

The possible reason for this can lie in the structure of the data itself. The court decisions consist of bigger blocks of sentences following each other, and usually, there is only limited information from other blocks that are useful for classifying a given sentence. Therefore, it is likely that the information from surrounding sentences introduces noise instead of useful context information. Instead, the information regarding the position in the legal document and the patterns from previous and forthcoming sentence vectors gathered by a BiLSTM network was proved to be significantly more valuable.

5.2. Possible Lack of Domain-Specific Knowledge

Another surprising result was that the Jina embeddings without late chunking also worked badly on our Hungarian legal cases. A reason for this could be the lack of Hungarian and domain-specific knowledge since the Jina embeddings were not fine-tuned specifically for Hungarian [37]. In contrast, these representations performed among the best in a Q&A-type retrieval task for Hungarian legal cases [1].

5.3. Effect of Vectorization Tasks

An additional reason for this could be a bad setting for the Jina vectorizer. Jina provides the following vectorization tasks: `classification`, `text-matching`, query and passage retrieval (`retrieval.passage`, `retrieval.query`), and `separation`, so the selection of `classification` seemed to be correct in this case. To validate this, we performed a linear separability comparison via 5-fold cross-validation on the training + validation sets with two repetitions, fixing the random state for the fold creation and the linear SVM classifier. We also applied stratified splitting on the labels since linear classification is independent of the order of the sentences. We compared the Jina vectors without late chunking using the `separation`, `text-matching`, and `classification` vector types and BERT CLS embeddings. The results are shown in Figure 8.

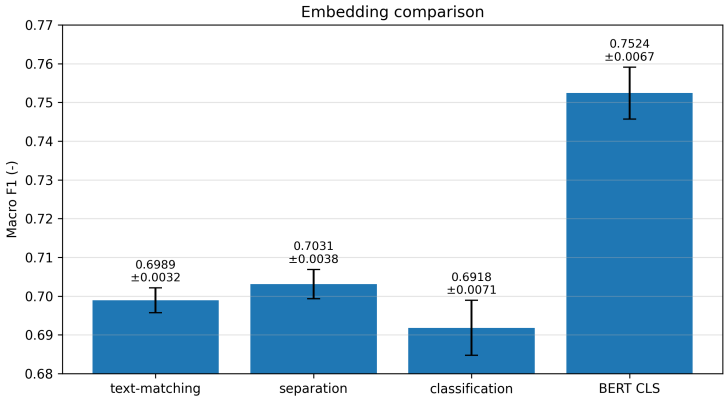


Figure 8. Comparison of the `separation` and `text-matching`, `classification` Jina embedding settings and BERT CLS, linear classification with SVMs.

We compared the BERT CLS and Jina embedding models with a two-sided paired t-test on macro F1 over n = 10 matched evaluations (same folds/seeds). The difference was statistically significant (`classification`: Δmacro-F1 = 0.061 (95% CI 0.055–0.066), $p = 2.51 \times 10^{-9}$, `separation`: Δmacro-F1 = 0.049 (95% CI 0.044–0.055), $p = 5.25 \times 10^{-9}$, `text-matching`: Δmacro-F1 = 0.053 (95% CI 0.048–0.059), $p = 3.99 \times 10^{-9}$). Interestingly, the `classification` setting proved to be the worst, while the `separation` was the best from Jina embeddings. This means that the results with Jina embeddings may be underestimated using the `classification` task. Nevertheless, none of the task settings could pair with the BERT CLS embeddings, while the `classification` result remained close to the `separation` result, so it is very unlikely that the separation setting would help in the RRL task's performance.

6. Conclusions

We presented, to our knowledge, the first sentence-level Rhetorical Role Labeling (RRL) system on Hungarian legal decisions, evaluated on a newly curated corpus and compared across classical and neural architectures. In production settings, the model supports role-aware retrieval for legal retrieval setup, enabling downstream tasks that benefit from filtering by rhetorical roles.

Empirically, sequential encoders that model local order and document-internal regularities proved to be the most effective. A BiLSTM fed with Hungarian BERT (huBERT) CLS embeddings achieved the strongest overall results on the held-out test set, clearly surpassing a linear SVM baseline, which validates the importance of sequence information for this task.

Contrary to recent findings in retrieval, late chunking hurt performance for sentence-level RRL, and multilingual Jina v3 embeddings did not outperform Hungarian BERT CLS, suggesting that injecting broad document-level context into fixed sentence vectors introduces noise for role prediction.

Label-wise analysis indicates consistently high scores on major categories (e.g., Discussion, Arguments, Case History), with Decision and Operative Part remaining comparatively difficult.

The work has immediate operational impact: the best model is deployed at the National Office for the Judiciary and already powers a rhetorical role-conditioned RAG pipeline, improving searchability and explainability of Hungarian court decisions.

Author Contributions: Conceptualization, G.M.C.; methodology, G.M.C., I.Ü., and D.L.; software, G.M.C.; validation, D.R. and K.K.; formal analysis, G.M.C.; investigation, G.M.C.; resources, D.N., G.M.C., D.R., and K.K.; data curation, D.N., G.M.C., D.R., and K.K.; writing—original draft prepara-

tion, G.M.C. and I.Ü.; writing—review and editing, I.Ü., D.N., and J.P.V.; visualization, G.M.C. and D.L.; supervision, G.M.C., D.N., and J.P.V.; project administration, D.N. and J.P.V.; funding acquisition, J.P.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: Gergely Márk Csányi, István Üveges, Dorina Lakatos, Dániel Nagy, and János Pál Vadász were employed by MONTANA Knowledge Management Ltd. The remaining authors declare that this research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Csányi, G.M.; Lakatos, D.; Üveges, I.; Megyeri, A.; Vadász, J.P.; Nagy, D.; Vági, R. From Fact Drafts to Operational Systems: Semantic Search in Legal Decisions Using Fact Drafts. *Big Data Cogn. Comput.* **2024**, *8*, 185. https://doi.org/10.3390/bdcc8120185.
2. Wang, H.; He, T.; Zou, Z.; Shen, S.; Li, Y. Using case facts to predict accusation based on deep learning. In Proceedings of the 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), Sofia, Bulgaria, 22–26 July 2019; IEEE: New York, NY, USA, 2019; pp. 133–137. https://doi.org/10.1109/QRS-C.2019.00038.
3. Muhammed, A.; Muslihuddeen, H.; Sankar, S.; Kumar, M.A. Impact of Rhetorical Roles in Abstractive Legal Document Summarization. In Proceedings of the 2024 5th International Conference on Innovative Trends in Information Technology (ICITIIT), Kottayam, India, 15–16 March 2024; IEEE: New York, NY, USA, 2024; pp. 1–6. https://doi.org/10.1109/ICITIIT61487.2024.10580671.
4. Bhattacharya, P.; Paul, S.; Ghosh, K.; Ghosh, S.; Wyner, A. Identification of rhetorical roles of sentences in Indian legal judgments. In *Legal Knowledge and Information Systems*; IOS Press: Amsterdam, The Netherlands, 2019; pp. 3–12. https://doi.org/10.3233/FAIA190301.
5. Malik, V.; Sanjay, R.; Guha, S.K.; Hazarika, A.; Nigam, S.K.; Bhattacharya, A.; Modi, A. Semantic segmentation of legal documents via rhetorical roles. In Proceedings of the Natural Legal Language Processing Workshop 2022, Abu Dhabi, United Arab Emirates, 8 December 2022; pp. 153–171. https://doi.org/10.18653/v1/2022.nllp-1.13.
6. Santosh, T.; Isaia, A.; Hong, S.; Grabmair, M. HiCuLR: Hierarchical Curriculum Learning for Rhetorical Role Labeling of Legal Documents. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, FL, USA, 12–16 November 2024; pp. 7357–7364. https://doi.org/10.18653/v1/2024.findings-emnlp.433.
7. Bambroo, P.; Adhikary, S.; Bhattacharya, P.; Chakraborty, A.; Ghosh, S.; Ghosh, K. MARRO: multi-headed attention for rhetorical role labeling in legal documents. In *Artificial Intelligence and Law*; Springer: Berlin/Heidelberg, Germany, 2025; pp. 1–30. https://doi.org/10.1007/s10506-025-09449-7.
8. Nigam, S.K.; Dubey, T.; Sharma, G.; Shallum, N.; Ghosh, K.; Bhattacharya, A. LegalSeg: Unlocking the Structure of Indian Legal Judgments Through Rhetorical Role Classification. In Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, NM, USA, 29 April–4 May 2025; pp. 1129–1144. https://doi.org/10.18653/v1/2025.findings-naacl.63.
9. Marino, G.; Licari, D.; Bushipaka, P.; Comandé, G.; Cucinotta, T. Automatic rhetorical roles classification for legal documents using legal-transformer over BERT. In Proceedings of the CEUR WORKSHOP PROCEEDINGS. CEUR-WS, Örebro, Sweden, 15–17 June 2023; Volume 3441, pp. 28–36.
10. Aragy, R.; Fernandes, E.R.; Caceres, E.N. Rhetorical role identification for Portuguese legal documents. In Proceedings of the Brazilian Conference on Intelligent Systems, Virtual, 29 November–3 December 2021; pp. 557–571. https://doi.org/10.1007/978-3-030-91699-2_38.
11. Günther, M.; Mohr, I.; Williams, D.J.; Wang, B.; Xiao, H. Late chunking: Contextual chunk embeddings using long-context embedding models. *arXiv* **2024**, arXiv:2409.04701. https://doi.org/10.48550/arXiv.2409.04701.
12. MANN, W.C.; THOMPSON, S.A. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text -Interdiscip. J. Study Discourse* **1988**, *8*, 243–281. https://doi.org/10.1515/text.1.1988.8.3.243.
13. Taboada, M.; Mann, W.C. Rhetorical Structure Theory: Looking back and moving ahead. *Discourse Stud.* **2006**, *8*, 423–459. https://doi.org/10.1177/1461445606061881.

14. Taboada, M.; Mann, W.C. Applications of Rhetorical Structure Theory. *Discourse Stud.* **2006**, *8*, 567–588. https://doi.org/10.1177/1461445606064836.
15. Knott, A.; Dale, R. Using linguistic phenomena to motivate a set of coherence relations. *Discourse Process* **1994**, *18*, 35–62. https://doi.org/10.1080/01638539409544883.
16. Knott, A.; Sanders, T. The classification of coherence relations and their linguistic markers: An exploration of two languages. *J. Pragmat.* **1998**, *30*, 135–175. https://doi.org/10.1016/s0378-2166(98)00023-x.
17. Teufel, S.; Moens, M. Summarizing Scientific Articles: Experiments with Relevance and Rhetorical Status. *Comput. Linguist.* **2002**, *28*, 409–445. https://doi.org/10.1162/089120102762671936.
18. Hachey, B.; Grover, C. Extractive summarisation of legal texts. *Artif. Intell. Law* **2007**, *14*, 305–345. https://doi.org/10.1007/s10506-007-9039-z.
19. Saravanan, M.; Ravindran, B. Identification of Rhetorical Roles for Segmentation and Summarization of a Legal Judgment. *Artif. Intell. Law* **2010**, *18*, 45–76. https://doi.org/10.1007/s10506-010-9087-7.
20. Walker, V.R.; Pillaipakkamnatt, K.; Davidson, A.M.; Linares, M.; Pesce, D.J. Automatic Classification of Rhetorical Roles for Sentences: Comparing Rule-Based Scripts with Machine Learning. *ASAIL@ ICAIL* **2019**, *2385*, 1–10
21. Sanchez, G. Sentence Boundary Detection in Legal Text. In Proceedings of the Natural Legal Language Processing Workshop 2019. Association for Computational Linguistics, Minneapolis, MN, USA, 6–7 June 2019; pp. 31–38. https://doi.org/10.18653/v1/w19-2204.
22. Kalamkar, P.; Tiwari, A.; Agarwal, A.; Karn, S.; Gupta, S.; Raghavan, V.; Modi, A. Corpus for Automatic Structuring of Legal Documents. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France, 20–25 June 2022; pp. 4420–4429.
23. Aumiller, D.; Almasian, S.; Lackner, S.; Gertz, M. Structural text segmentation of legal documents. In Proceedings of the 18th International Conference on Artificial Intelligence and Law, São Paulo, Brazil, 21–25 June 2021; ACM: New York, NY, USA, 2021; ICAIL '21, pp. 2–11. https://doi.org/10.1145/3462757.3466085.
24. Guimarães, G.M.C.; da Silva, F.X.B.; Macêdo, L.d.A.B.; Lisboa, V.H.F.; Marcacini, R.M.; Queiroz, A.L.; Borges, V.R.P.; Faleiros, T.d.P.; Garcia, L.P.F. Legal Document Segmentation and Labeling Through Named Entity Recognition Approaches. *J. Inf. Data Manag.* **2024**, *15*, 123–131.
25. Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; Liu, Z. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In Proceedings of the Findings of the Association for Computational Linguistics ACL 2024, Bangkok, Thailand, 11–16 August 2024; pp. 2318–2335. https://doi.org/10.18653/v1/2024.findings-acl.137.
26. Niklaus, J. Decoding Legalese Without Borders: Multilingual Evaluation of Language Models on Long Legal Texts. Ph.D. Thesis, University of Bern, Bern, Switzerland, 2024.
27. Sturua, S.; Mohr, I.; Kalim Akram, M.; Günther, M.; Wang, B.; Krimmel, M.; Wang, F.; Mastrapas, G.; Koukounas, A.; Wang, N.; et al. Jina Embeddings V3: Multilingual Text Encoder with Low-Rank Adaptations. In Proceedings of the European Conference on Information Retrieval, Lucca, Italy, 6–10 April 2025; Springer: Berlin/Heidelberg, Germany, 2025; pp. 123–129. https://doi.org/10.1007/978-3-031-88720-8_21.
28. Zhong, Z.; Liu, H.; Cui, X.; Zhang, X.; Qin, Z. Mix-of-Granularity: Optimize the Chunking Granularity for Retrieval-Augmented Generation. In Proceedings of the 31st International Conference on Computational Linguistics, Abu Dhabi, United Arab Emirates, 19–24 January 2025; pp. 5756–5774.
29. Setty, S.; Thakkar, H.; Lee, A.; Chung, E.; Vidra, N. Improving Retrieval for RAG based Question Answering Models on Financial Documents. *arXiv* **2024**, arXiv:2404.07221. https://doi.org/10.48550/ARXIV.2404.07221.
30. National Office for the Judiciary. Anonymized Hungarian Court Documents. Available online: https://eakta.birosag.hu/anonimizalt-hatarozatok (accessed on 24 November 2025).
31. Csányi, G.M.; Lakatos, D.; Üveges, I.; Vági, R.; Megyeri, A.; Fülöp, A.; Nagy, D.; Vadász, J.P. Evaluating the Effectiveness of Automatic Sentence Segmentation for Judicial Decisions, (Bírósági határozatok automatikus mondatszegmentálásának hatékonyságmérése). In Proceedings of the XX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2024), Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Hungary, 25–26 January 2024.
32. Orosz, G.; Szabó, G.; Berkecz, P.; Szántó, Z.; Farkas, R. Advancing Hungarian Text Processing with HuSpaCy: Efficient and Accurate NLP Pipelines. In Proceedings of the Text, Speech, and Dialogue, Pilsen, Czech Republic, 23 August 2023; Springer Nature: Cham, Switzerland, 2023; pp. 58–69. https://doi.org/10.1007/978-3-031-40498-6_6.
33. Krippendorff, K. Computing Krippendorff's Alpha-Reliability. Available online: https://repository.upenn.edu/entities/publication/034a6030-c584-4d14-9d3d-7b7e8d16df20 (accessed on 24 November 2025).
34. Orosz, T.; Vági, R.; Csányi, G.M.; Nagy, D.; Üveges, I.; Vadász, J.P.; Megyeri, A. Evaluating Human versus Machine Learning Performance in a LegalTech Problem. *Appl. Sci.* **2021**, *12*, 297. https://doi.org/10.3390/app12010297.

35. Vatsal, S.; Meyers, A.; Ortega, J.E. Classification of US Supreme Court Cases Using BERT-Based Techniques. In Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, Varna, Bulgaria, 6 May 2023; pp. 1207–1215. https://doi.org/10.26615/978-954-452-092-2_128.
36. Günther, M.; Sturua, S.; Akram, M.K.; Mohr, I.; Ungureanu, A.; Wang, B.; Eslami, S.; Martens, S.; Werk, M.; Wang, N.; et al. jina-embeddings-v4: Universal embeddings for multimodal multilingual retrieval. In Proceedings of the 5th Workshop on Multilingual Representation Learning (MRL 2025), Suzhuo, China, 8–9 November 2025; pp. 531–550. https://doi.org/10.18653/v1/2025.mrl-main.36.
37. Jina AI. Jina Embedding v3 HuggingFace. Available online: https://huggingface.co/jinaai/jina-embeddings-v3 (accessed on 24 November 2025).
38. Nemeskey, D.M. Introducing `huBERT`. In Proceedings of the XVII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2021), Szeged, Hungary, 28–29 January 2021; p. TBA.
39. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.
40. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30 6000 - 6010*.
41. Bhattacharya, P.; Paul, S.; Ghosh, K.; Ghosh, S.; Wyner, A. DeepRhole: deep learning for rhetorical role labeling of sentences in legal case documents. *Artif. Intell. Law* **2021**, *31*, 53–90. https://doi.org/10.1007/s10506-021-09304-5.
42. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
43. Merola, C.; Singh, J. Reconstructing context: Evaluating advanced chunking strategies for retrieval-augmented generation. In Proceedings of the International Workshop on Knowledge-Enhanced Information Retrieval, Lucca, Italy, 10 April 2025; Springer: Berlin/Heidelberg, Germany, 2025; https://doi.org/10.1007/978-3-032-02899-0_1.