TURKISH LEGAL NLP: A COMPREHENSIVE AI FRAMEWORK FOR LEGAL DOCUMENT UNDERSTANDING, SUMMARIZATION, AND RETRIEVAL-AUGMENTED GENERATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MEHMET ALİ ERKAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
STATISTICS

AUGUST 2025

Approval of the thesis:

TURKISH LEGAL NLP: A COMPREHENSIVE AI FRAMEWORK FOR LEGAL DOCUMENT UNDERSTANDING, SUMMARIZATION, AND RETRIEVAL-AUGMENTED GENERATION

submitted by **MEHMET ALİ ERKAN** in partial fulfillment of the requirements for the degree of **Master of Science in Statistics Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Vilda Purutçuoğlu
Head of Department, **Statistics** _____

Prof. Dr. Ceylan Yozgatlıgil
Supervisor, **Statistics, METU** _____

Examining Committee Members:

Prof. Dr. Özlem İlk
Department of Statistics, METU _____

Prof. Dr. Ceylan Yozgatlıgil
Department of Statistics, METU _____

Prof. Dr. İhsan Tolga Medeni
Management Information System, AYBU _____

Date: 26.08.2025

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Mehmet Ali Erkan

Signature :

ABSTRACT

TURKISH LEGAL NLP: A COMPREHENSIVE AI FRAMEWORK FOR LEGAL DOCUMENT UNDERSTANDING, SUMMARIZATION, AND RETRIEVAL-AUGMENTED GENERATION

Erkan, Mehmet Ali
M.S., Department of Statistics
Supervisor: Prof. Dr. Ceylan Yozgatlıgil

August 2025, 182 pages

Natural Language Processing (NLP) has been widely used across various domains; however, legal texts especially particularly in Turkish remain relatively underexplored. The complexity and richness of the Turkish legal language make it a valuable yet challenging source for artificial intelligence (AI) systems. This thesis presents an end-to-end AI-based framework designed to understand, summarize, and evaluate Turkish legal texts.

A large-scale dataset was compiled, consisting of 23,035 judicial decisions and 9,277 legislative and regulatory documents collected from Yargıtay, Danıştay, the Constitutional Court, UYAP, legislative bulletins, and ministry decisions. These texts are cleaned, normalized, and structured to create benchmark datasets for legal NLP in Turkish.

A detailed tokenization analysis was conducted, highlighting the trade-offs between efficiency and semantic representation across character-level, word-based, and legal-aware approaches. Embedding methods were systematically compared, and findings indicated that no single technique universally dominates; instead, their effectiveness

varies depending on the legal text category and downstream task.

Additionally, a new summarization algorithm was created that was specifically designed for the structural features of Turkish legal language. It beat traditional baselines by being more relevant, coherent, and legally comprehensive. Finally, a Retrieval-Augmented Generation (RAG) system was built by combining dense retrieval and generation models that work best in the legal field. This system showed that it could help lawyers and citizens with research, making decisions, and accessing legal information by being very accurate and reliable when it came to facts and citations.

Keywords: Turkish Law, Legal NLP, Text Summarization, Tokenization, Embedding Methods, Retrieval-Augmented Generation, Legal AI Tools, Turkish Language Processing

ÖZ

TÜRKÇE HUKUK NLP: HUKUKİ METİN ANLAMA, ÖZETLEME VE ERİŞİM DESTEKLİ ÜRETİM İÇİN KAPSAMLI YAPAY ZEKÂ YAKLAŞIMI

Erkan, Mehmet Ali
Yüksek Lisans, İstatistik Bölümü
Tez Yöneticisi: Prof. Dr. Ceylan Yozgatlıgil

Ağustos 2025 , 182 sayfa

Doğal Dil İşleme yöntemleri birçok alanda yaygın olarak kullanılmasına rağmen, özellikle Türkçe hukuk metinleri bu kapsamda görece az araştırılmış durumdadır. Türkçe hukuk dilinin karmaşık yapısı ve zengin içeriği, yapay zekâ (YZ) sistemleri için değerli fakat zorlu bir kaynak haline getirmektedir. Bu tez, Türkçe hukuk metinlerini anlama, özetleme ve değerlendirme amacıyla tasarlanmış uçtan uca bir YZ tabanlı yaklaşım sunmaktadır.

Çalışma kapsamında, Yargıtay, Danıştay, Anayasa Mahkemesi, UYAP, mevzuat bültenleri ve bakanlık kararlarından toplanan 23.035 yargı kararı ve 9.277 mevzuat ve düzenleyici belge içeren geniş ölçekli bir veri kümesi oluşturulmuştur. Bu metinler temizlenmiş, normalleştirilmiş ve Türkçe hukuk alanına yönelik kıyaslama veri kümeleri oluşturmak üzere yapılandırılmıştır.

Ayrıntılı bir tokenizasyon analizi gerçekleştirilmiş ve karakter tabanlı, kelime tabanlı ve hukuk-farkındalıklı yaklaşımlar arasında verimlilik ile anlamsal temsiliyet arasındaki denge ortaya konmuştur. Gömme yöntemleri sistematik olarak karşılaştırılmış ve

bulgular, genel olarak üstün bir tekniğin bulunmadığını; aksine yöntemlerin etkinliğinin, hukuki metin kategorisine ve ilgili alt göreve bağlı olarak değiştiğini göstermiştir.

Bunlara ek olarak, Türkçe hukuk dilinin yapısal özelliklerine özel olarak tasarlanmış yeni bir özetleme algoritması geliştirilmiştir. Bu algoritma, geleneksel taban çizgisi yöntemlerini aşarak daha yüksek düzeyde ilgililik, tutarlılık ve hukuki kapsayıcılık sağlamıştır. Son olarak, hukuk alanında en iyi performansı veren yoğun erişim ve üretim modellerini birleştirerek bir Erişim Destekli Üretim (RAG) sistemi inşa edilmiştir. Bu sistem, olgular ve atıflar konusunda yüksek doğruluk ve güvenilirlik sağlayarak yalnızca avukatlara değil, aynı zamanda vatandaşlara da araştırma yapma, karar verme ve hukuki bilgiye erişim süreçlerinde yardımcı olabileceğini göstermiştir.

Anahtar Kelimeler: Türk Hukuku, Hukuk NLP, Metin Özetleme, Karşılaştırmalı Veri Seti, Tokenizasyon, Gömme Yöntemleri, Erişim Destekli Üretim, Hukuk Yapay Zeka Araçları, Türkçe Dil İşleme

To my lovely family

ACKNOWLEDGMENTS

This intense journey would have been much harder without my esteemed professors, family, friends, and loved ones. I am deeply grateful to all of them for making this process easier.

First and greatest, I would like to express my sincere gratitude to my dear advisor, Professor Ceylan Yozgatlıgil, who opened the doors of this field for me, constantly supported me, and never stopped learning alongside me. I am also thankful to Professor İhsan Tolga Medeni, with whom I had the privilege to teach and work on numerous projects. Additionally, I am grateful to Professor Özlem İlk, from whom I took strong foundational courses during my undergraduate and graduate studies and for her valuable suggestions throughout my thesis work.

I am deeply thankful to TÜBİTAK TRUBA and the Barcelona Supercomputing Center for providing access to MN5.

I want to express my gratitude to Tarık and Halide, with whom I collaborated throughout the foundations of this project. I extend my thanks to my colleague Pelin, who accompanied me at the beginning of this academic journey, and to my dear friends Oğuzhan, Okan, Kübra, and Ekinsu, whose friendship has continued from my undergraduate years to graduate school, as well as to Dinçer, Onur, Mehmet, Sezai, and Kıvanç for their invaluable companionship.

I am especially thankful to my dear Elif Beyza, who gave me energy and helped me keep going through this challenging period. Also to my beloved family, who have always stood by me, supported me unconditionally and raised me to be who I am.

Finally, I feel immense fulfillment in being a part of METU.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xviii

LIST OF FIGURES xx

LIST OF ABBREVIATIONS xxi

CHAPTERS

1 INTRODUCTION 1

2 LITERATURE REVIEW 7

2.1 Tokenization 8

2.1.1 Morphological complexity in Turkish tokenization 8

2.1.1.1 Subword and neural tokenization approaches 8

2.1.1.2 Legal domain tokenization challenges 9

2.1.1.3 Evaluation and benchmarking 9

2.2 Text summarization in general, legal domain, and Turkish contexts . . 9

2.2.1 General text summarization methodologies 10

2.2.2 Turkish legal document summarization 10

2.2.3 Evaluation frameworks for legal summarization 11
2.2.4 RAG architecture innovations for legal domain 11
2.2.5 Large language model applications in legal contexts 12
2.2.6 Legal chatbots and conversational AI systems 12
2.2.7 Evaluation benchmarks and methodologies 13
2.3 Large Language Models and legal domain adaptation 14
2.3.1 Domain-specific pre-training approaches 14
2.3.2 Turkish legal language model development 14
2.3.3 Cross-lingual legal language models 15
2.4 Turkish NLP morphological processing and challenges 15
2.4.1 Morphological disambiguation challenges 16
2.4.2 Stemming and lemmatization approaches 17
2.4.3 Parsing and syntactic analysis 17
2.4.4 Neural and transformer approaches 18
2.5 Legal NLP applications and cross-lingual approaches 18
2.5.1 Comprehensive legal AI applications 19
2.5.2 Cross-lingual legal information retrieval 19
2.5.3 Multilingual legal document processing 20
2.5.4 Legal information extraction across languages 20
2.5.5 Turkish integration in multilingual legal systems 21
2.5.6 Evaluation frameworks for cross-lingual legal NLP 21
2.6 Research gaps and future directions 22
2.6.1 Dataset and corpus development priorities 22

2.6.2 Evaluation methodology advancement 23
2.6.3 Technical architecture innovations 23
2.6.4 Cross-lingual integration opportunities 24
2.6.5 Practical implementation challenges 24
3 METHODOLOGY 27
3.1 Introduction 27
3.2 Data Collection and Preprocessing 27
3.2.1 Web Crawling Framework 27
3.2.2 Document Preprocessing Pipeline 29
3.2.3 Tokenization Methods 30
3.2.3.1 Character-Level Tokenization 30
3.2.3.2 Word-Level Tokenization 31
3.2.3.3 Morphological Tokenization 31
3.2.3.4 Legal-Aware Tokenization 32
3.2.3.5 Byte-Pair Encoding (BPE) 33
3.2.3.6 SentencePiece Tokenization 34
3.2.3.7 Compression Ratio Calculation 34
3.3 Document Embedding Methods 34
3.3.1 Count Vectorizer 35
3.3.2 TF-IDF (Term Frequency–Inverse Document Frequency) . . . 35
3.3.3 Latent Semantic Analysis (LSA) 35
3.3.4 Principal Component Analysis (PCA) Embedding 36
3.3.5 Non-negative Matrix Factorization (NMF) 37

3.3.6 Composite Score Calculation 37
3.4 Legal Document Summarization 38
3.4.1 Extractive Summarization Framework 38
3.4.1.1 Keyword Importance Score 39
3.4.1.2 Coherence Score 39
3.4.1.3 Positional Weight 39
3.4.1.4 Length Normalization 40
3.4.1.5 Legal Terminology Factor 40
3.4.2 Baseline Summarization Methods 41
3.4.2.1 TextRank Algorithm 41
3.4.2.2 Frequency-Based Summarization 42
3.4.2.3 Position-Based Summarization 42
3.4.2.4 Luhn's Algorithm 43
3.4.2.5 Maximal Marginal Relevance (MMR) 43
3.4.2.6 BERT-based Extractive Summarization 44
3.4.2.7 LegalBERT-based Summarization 44
3.4.2.8 Proposed Legal Summarization Algorithm 45
3.4.3 Evaluation Metrics for Summarization 46
3.4.3.1 ROUGE Scores 46
3.4.3.2 BLEU Score 48
3.4.3.3 Legal Coverage 49
3.4.3.4 Structure Preservation 50
3.4.3.5 Composite Text Generation Score 51

3.5 Retrieval-Augmented Generation (RAG) 53
3.5.1 Dense Retrieval 53
3.5.2 BM25 Scoring 54
3.5.3 Hybrid Retrieval 54
3.5.4 Retrieval Metrics 55
3.6 Document Classification 55
3.6.1 Hierarchical Classification 55
3.6.2 Classification Metrics 56
3.7 Clustering and Topic Modeling 57
3.7.1 K-means Clustering 57
3.7.2 Silhouette Score 58
3.7.3 Latent Dirichlet Allocation (LDA) 58
3.7.4 Topic Coherence (UMass) 59
3.8 Statistical Analysis Methods 59
3.8.1 Power Law Analysis 60
3.8.2 Zipf's Law Analysis 60
3.8.3 Type–Token Ratio (TTR) 61
3.8.4 Kolmogorov–Smirnov Test 61
3.9 Computational Efficiency Metrics 62
3.9.1 Scaling Law Analysis 63
3.9.2 Processing Efficiency 63
4 ANALYSIS OF TURKISH LEGAL LANGUAGE PROCESSING SYSTEMS 67
4.1 Dataset Characteristics and Statistical Analysis 68

4.1.1 Data Set Creation 68
4.1.2 Corpus Overview and Distribution Analysis 70
4.1.3 Morphological Complexity Analysis 72
4.1.4 Legal Domain Characteristics 73
4.2 Tokenization Method Evaluation 73
4.2.1 Comparative Analysis of Tokenization Approaches 74
4.2.2 Subword Tokenization Performance and Vocabulary Optimization 76
4.2.3 Cross-Category Analysis and Deployment Recommendations . 76
4.3 Embedding Method Analysis 77
4.3.1 Document Embedding Evaluation Framework 77
4.3.2 Semantic Representation Quality Analysis 78
4.3.3 Clustering Performance and Geometric Structure 79
4.3.4 Dimensionality and Efficiency Analysis 80
4.3.5 Feature Engineering Impact on Turkish Legal Text 80
4.3.6 Method Selection Guidelines 81
4.4 Legal Document Summarization Results 82
4.4.1 Extractive Summarization Analysis 82
4.4.2 Abstractive Summarization Evaluation 85
4.5 Retrieval-Augmented Generation Analysis 86
4.5.1 Dense Retrieval Performance 87
4.5.2 RAG Generation Quality Analysis 88
4.6 Legal Document Classification Results 90

4.6.1 Multi-label Classification Performance 90
4.7 Clustering and Topic Analysis 93
4.7.1 Legal Document Clustering Performance 93
4.7.2 Topic Modeling Results 94
4.7.3 Semantic Network Analysis 95
4.8 Computational Efficiency Analysis 98
4.8.1 Model Size and Inference Speed 98
4.8.2 Task-Specific Performance Analysis 100
4.8.3 Scaling Analysis and Resource Optimization 101
4.9 Error Analysis and Limitations 103
4.9.1 Systematic Error Patterns 103
4.9.2 Performance Limitations by Document Category 104
4.9.3 Model-Specific Limitations and Failure Analysis 105
5 CONCLUSION AND FUTURE STUDIES 107
REFERENCES 177
APPENDICES 182

LIST OF TABLES

TABLES

Table 4.1 Turkish Legal Corpus Distribution by Institution and Document Type 70

Table 4.2 Comprehensive Tokenization Method Comparison on Turkish Legal Corpus 74

Table 4.3 Category-Specific Tokenization Performance Analysis 75

Table 4.4 Cross-Category Performance Summary 76

Table 4.5 Document Embedding Evaluation Results on Turkish Legal Corpus . 78

Table 4.6 Legal Document Summarization Results with Detailed Metrics . . . 84

Table 4.7 Abstractive Legal Summarization Performance 85

Table 4.8 Dense Retrieval Performance on Turkish Legal Corpus 87

Table 4.9 RAG System Performance on Turkish Legal Queries 89

Table 4.10 Legal Document Classification Results 91

Table 4.11 Comprehensive Comparison of Methods 93

Table 4.12 Document Clustering Performance by Embedding and Method . . . 94

Table 4.13 Dominant Topics in Turkish Legal Documents 95

Table 4.14 High-Centrality Legal Concepts in Turkish Legal Discourse 97

Table 4.15 Legal Domain Distribution in Topic Analysis 98

Table 4.16 Computational Efficiency Comparison Across Turkish Legal NLP . 99

Table 4.17 Task-Specific Computational Performance 100

Table 4.18 Error Pattern Analysis Across Legal NLP Tasks 104

Table 4.19 Performance Limitations by Legal Document Type 105

Table 5.1 Legal Questions and Responses from Different AI - RAG Models . . 111

LIST OF FIGURES

FIGURES

Figure 3.1 Methodology Framework for Turkish Legal 65

Figure 4.1 Web Crawling Framework 69

Figure 4.2 Detailed Distribution of Turkish Legal Corpus by Document Type 71

Figure 4.3 Morphological Complexity Distribution and Comparison 72

Figure 4.4 Document Embedding Methods and Performance Trade-offs . . 81

Figure 4.5 Structure of Our Summarization Method 83

Figure 4.6 Structure of Our RAG Method 87

Figure 4.7 Semantic Network 96

LIST OF ABBREVIATIONS

ABBREVIATIONS	
AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
BLEU	Bilingual Evaluation Understudy
BM25	Best Matching 25
BP	Brevity Penalty
BPE	Byte-Pair Encoding
BTF	Binary Term Frequency
CPU	Central Processing Unit
CR	Compression Ratio
CV	Count Vectorizer
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
ELMo	Embeddings from Language Models
FST	Finite State Transducer
GloVe	Global Vectors for Word Representation
GPU	Graphics Processing Unit
KS	Kolmogorov–Smirnov Test
LDA	Latent Dirichlet Allocation
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
LCS	Longest Common Subsequence
MLM	Masked Language Model
MMR	Maximal Marginal Relevance
MRR	Mean Reciprocal Rank

NER	Named Entity Recognition
NLP	Natural Language Processing
NSP	Next Sentence Prediction
NMF	Non-negative Matrix Factorization
PCA	Principal Component Analysis
P@k	Precision@k (Precision at top-k retrieved items)
R@k	Recall@k (Recall at top-k retrieved items)
QA	Question Answering
RAG	Retrieval-Augmented Generation
RNN	Recurrent Neural Network
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SENTENCEPIECE	SentencePiece Tokenization Algorithm
SVD	Singular Value Decomposition
TF-IDF	Term Frequency-Inverse Document Frequency
TTR	Type–Token Ratio
UYAP	Ulusal Yargı Ağı Bilişim Sistemi (National Judiciary Informatics System)
YZ	Yapay Zeka (Artificial Intelligence in Turkish)

CHAPTER 1

INTRODUCTION

The digital revolution has fundamentally changed how information is created, stored, and processed. Today, text data represents the largest portion of all digital information, with studies showing that unstructured text comprises approximately 80% of organizational data [1]. This explosion of textual information spans across all sectors, from social media and news to scientific publications and legal documents. The growth of digital text has been particularly dramatic in recent decades. IBM estimated that 90% of the world's data was created in just two years between 2016 and 2018 [2]. This trend has increased with the widespread adoption of digital communication platforms, online publishing, and the digitization of traditional paper-based records. Government institutions such as including courts and legal bodies have been major contributors to this digital transformation, moving from paper-based systems to comprehensive digital archives.

This huge collection of textual data has created both opportunities and challenges. Even though we have a lot of information now, processing it by hand is getting harder because of its sheer volume. The amount of available text data has outgrown the capabilities of traditional information retrieval and analysis techniques, which relied mainly on human judgment and manual review. This has created a critical need for automated systems that can understand, analyze, and summarize textual information efficiently and accurately. The legal domain has been particularly affected by this digital transformation. Courts around the world have digitized millions of case files, creating extensive repositories of legal knowledge. In the United States, systems such as Westlaw and LexisNexis have provided digital access to legal documents since the 1970s [3]. Similarly, European courts have increasingly made their decisions

available online, with the European Court of Justice launching its digital platform in the early 2000s [4].

The area of Natural Language Processing (NLP) has developed significantly since its inception in the 1950s. Early work by researchers like Noam Chomsky on formal grammar applied the theoretical foundation for computational linguistics. The development of statistical methods in the 1990s, followed by the deep learning revolution of the 2010s, has enabled machines to perform increasingly sophisticated language understanding tasks.

In the 1970s and 1980s, legal natural language processing (NLP) became a separate field of study. McCarty (1977) created TAXMAN that is one of the first systems to analyze tax law. Systems like HYPO [5], which could evaluate court cases and produce arguments also were developed in the 1980s. These early systems showed the promise of computational methods for legal reasoning.

The intersection of law and computational linguistics offers unique challenges that determine legal NLP from other domain applications. Legal texts follow formal designs and use specialized terminology that differs significantly from everyday language. Court decisions generally follow established patterns that they begin with the facts of the case, proceed through legal analysis, and conclude with a ruling. Legislative texts have hierarchical structures consisting of sections, subsections, and clauses. These structural regularities make legal documents amenable to automated processing. However, the legal language also presents significant complexity. Legal terms often have specific meanings that differ from ordinary usage. For example, the legal concept of "consideration" in contract law has a precise technical meaning that differs from its everyday usage; it refers to something of value exchanged between parties as part of a contractual agreement. Legal reasoning involves complex logical relationships, including conditional statements, exceptions, and precedent-based arguments. Studies have shown that legal texts contain more complex syntactic structures than general domain texts. [6].

Recent advances in NLP have extended new possibilities for legal applications. The development of transformer-based models, such as BERT [7], has significantly improved performance on legal text classification tasks. Legal AI systems such as ROSS

Intelligence and Lex Machina, have demonstrated practical applications in legal research and case outcome prediction. However, most of these systems focus on English language legal systems, especially those of the United States and the United Kingdom.

Turkey's legal system presents a unique and complex environment for NLP research. The modern Turkish legal system was established after the founding of the Republic of Turkey in 1923, drawing heavily from European legal traditions, particularly the Swiss Civil Code and German Commercial Code [8]. This adoption of continental European legal frameworks created a comprehensive legal system that combines civil law traditions with unique Turkish characteristics.

The Turkish court system operates through a multitiered structure that includes several specialized high courts. The Yargıtay (Court of Cassation) functions as the highest court for civil and criminal matters, while the Danıştay (Council of State) handles administrative law cases. The Constitutional Court, established in 1962, reviews the constitutionality of laws and government actions [9]. Each of these institutions produces different types of legal documents with varying structures and terminology.

The digitization of Turkish legal documents has accelerated significantly over the past few years. The National Judiciary Informatics System (UYAP), launched in 2000, has digitized court operations and made many legal documents publicly available [10]. The Turkish Parliament has also digitized legislative records and various ministries publish regulatory decisions online. This effort has created a considerable corpus of Turkish legal texts appropriate for computational analysis.

However, the Turkish language offers unique challenges for NLP applications. Turkish is an agglutinative language meaning that complex words are created by adding multiple suffixes to root words. A single Turkish word can express meaning equivalent to entire phrases in other languages. For example, "yapabileceklerimizden" means "from the things we will be able to do." This morphological complexity presents challenges for traditional NLP approaches which are typically designed for languages with simpler word structures. Moreover, most NLP research is conducted in English. For example, systematic reviews show English appears in over 2,200 papers, while other languages typically feature in only a few hundred, and fewer than 10% of clinical NLP studies address non-English languages [11]. Furthermore, most legal NLP research is

English-centric; reviewed corpora indicate English accounts for approximately 56% of published work, significantly outpacing languages like Chinese, German, French, and Japanese [12].

Turkish legal terminology adds another term of complexity. Legal terms in Turkish have evolved through several historical phases. Legal terminology was heavily influenced by Arabic and Persian during the Ottoman period. The language reforms of the 1920s and 1930s aimed to replace foreign terms with Turkish equivalents [13]. However, many legal concepts retained their Arabic or Persian origins which result in a mixed terminological system that persists to this day.

Studies of Turkish NLP have shown that morphological complexity significantly affects system performance. It is demonstrated that Turkish word segmentation requires specialized approaches distinct from those used in Indo-European languages [14]. More recent work has shown that morphological analysis is crucial for practical Turkish NLP [15]. However, most of this research has focused on general domain texts with limited attention to legal language.

Despite the growing availability of Turkish legal texts and advances in NLP technology, research on Turkish legal NLP remains limited. A survey of legal NLP literature reveals that most studies focus on English-language legal systems [16]. To the best of our knowledge, studies of non-English legal systems are rare, and research on Turkish legal texts is virtually nonexistent. This research gap is important given the substantial volume of Turkish legal documents available digitally. The UYAP system contains millions of court decisions, while legislative databases include thousands of laws and regulations. However, the lack of specialized NLP tools means that this wealth of legal information remains unavailable mainly for automated analysis. Legal professionals still rely primarily on manual search methods, limiting their ability to navigate the complex legal landscape efficiently.

The few existing studies of the Turkish legal language have been mainly descriptive in nature rather than computational in nature. Linguistic analyzes have examined the syntactic and semantic properties of legal Turkish [17] but these studies have not led to the development of practical NLP tools. Similarly, while general Turkish NLP has made significant progress, applications in the legal domain have been largely ignored.

International research has shown the potential benefits of legal NLP systems. In the United States, systems like Westlaw Edge utilize NLP to enhance legal research efficiency. European projects such as JUSTICIA, have developed multilingual legal information systems [18]. However, these systems do not support Turkish. Developing legal Turkish NLP tools could handle several critical needs. Automated analysis could significantly reduce research time and improve case preparation efficiency for legal professionals. Moreover, NLP tools could enable large-scale analysis of legal trends and patterns for researchers. Then, improved access to legal information could support legal literacy and civic participation for citizens.

This thesis addresses the research gap in Turkish legal natural language processing (NLP) through a comprehensive approach that combines dataset creation, algorithm development, and system implementation. The work makes several key contributions to both Turkish NLP community and broader field of legal informatics.

Firstly, we create the first large scale data sets for the Turkish legal NLP. These datasets compiled from publicly available sources including court decisions, legislative texts, and regulatory documents. This provide a crucial foundation for future research. The data sets include documents from major Turkish legal institutions, including Yargıtay decisions, Danıştay rulings, Constitutional Court judgments, UYAP case files, legislative bulletins and ministry regulations.

Secondly, we conduct a systematic evaluation of NLP techniques for Turkish legal texts. This includes a comparison of different tokenization approaches, word embedding methods, and transformer based neural network architectures. Our analysis provides evidence based recommendations for the design of the Turkish legal NLP system, and adressing the specific challenges created by the Turkish morphology and legal terminology.

Thirdly, we develop a specialized summarization algorithm that is tailored to Turkish legal documents. Our approach accounts for the distinct structural and semantic characteristics of legal documents unlike standard summarization methods. The algorithm includes features specific to the legal domain such as hierarchical document structure, precedent relationships, and specialized terminology.

Finally, we implement a Retrieval Augmented Generation (RAG) system that integrates text summarization and document retrieval. This end to end system which provides automated tools for searching, evaluating, and summarizing legal texts, demonstrates the usefulness of Turkish legal natural language processing.

In light of these contributions, this thesis is organized as follows. Chapter 2 provides a comprehensive literature review on legal NLP and related methodologies. Chapter 3 outlines the proposed methodology, including the design choices tailored to the Turkish legal context. Chapter 4 presents the data set construction process along with the experimental analyzes and evaluations. The main findings, practical implications, and suggestions for future research are all in Chapter 5.

CHAPTER 2

LITERATURE REVIEW

Turkish legal Natural Language Processing represents a convergence of two complex domains which called the morphologically rich Turkish language and the specialized legal domain. The intersection of these areas presents unique computational challenges that require sophisticated approaches beyond traditional NLP methodologies. This literature review discusses the current state of Turkish legal NLP research, spanning foundational morphological processing through cutting-edge transformer architectures and their applications to legal document analysis.

The Turkish language's complex nature creates fundamental challenges for computational processing. Words can contain multiple morphemes that encode grammatical information. This leads to potentially infinite vocabulary sizes and complex morphophonological processes [19]. These challenges become even more pronounced when combined with legal texts' precision requirements and domain-specific terminology. Legal documents demand exact interpretation, making errors in morphological analysis or semantic understanding potentially consequential for legal practice [20].

Recent advances in transformer architectures and domain-specific language models have opened new possibilities for Turkish legal NLP. The development of BERTurk and specialized legal BERT variants shows significant progress and achieving over 90% accuracy in court decision prediction tasks and 93.8% F1-scores in Turkish legal named entity recognition. There are still significant research gaps, though, particularly in the areas of extensive annotated corpora, thorough evaluation frameworks, and cross-lingual applications that could make use of Turkish legal NLP in larger multilingual legal systems [21].

2.1 Tokenization

Tokenization is the foundational preprocessing step determining the success of downstream NLP tasks, particularly critical for morphologically complex languages like Turkish. Traditional word-level tokenization approaches fail to capture the internal structure of Turkish words, where a single surface form can encode multiple morphemes containing crucial grammatical and semantic information.

2.1.1 Morphological complexity in Turkish tokenization

Oflazer's pioneering work in 1994 [14] established the theoretical framework for Turkish computational morphology using two-level description methods. His approach demonstrated that Turkish words require decomposition into morphological units rather than treating them as atomic tokens. The challenge in Turkish's productive morphology, where words like "çalıştırılamamışlardır" (they could not be made to work) contain multiple derivational and inflectional morphemes that each carry distinct semantic content. Recent research by Toraman et al. [19] examined comprehensive comparisons of five tokenization approaches for Turkish, including morphological level tokenizers. Their systematic evaluation across six downstream tasks revealed that morphological-aware tokenization significantly impacts RoBERTa-based language model performance. The study demonstrated that vocabulary size optimization combined with morphological segmentation yields superior results compared to standard subword tokenization methods [22].

2.1.1.1 Subword and neural tokenization approaches

The emergence of transformer architectures has necessitated reexamining tokenization strategies for Turkish. Kaya and Tantuğ [22] investigated vocabulary size effects on Turkish BERT models and revealed task-dependent optimal configurations. Their ITUTurkBERT experiments showed that NER tasks benefit from larger vocabularies (128k tokens achieving 88.7% F1) while sentiment analysis tasks optimize at 64k vocabulary size which indicate the need for task-specific tokenization strategies.

2.1.1.2 Legal domain tokenization challenges

Legal text tokenization indicates additional complexities beyond general Turkish processing. Legal documents contain specialized terminology, complex sentence structures and also precise semantic requirements that demand careful tokenization decisions. The presence of Ottoman Turkish terms in historical legal documents, code-switching between languages, and domain-specific abbreviations creates unique tokenization challenges not addressed by general-purpose Turkish NLP tools. Recent work on Turkish legal BERT models by Zeidi, Amasyalı, and Erol [23] addressed these challenges through the adaptation of domain-specific vocabulary. Their legal domain pre-training approach achieved competitive performance (72.15% F1-score on multi-label classification) despite using smaller training corpora, suggesting that specialized tokenization strategies combined with domain-specific vocabulary can overcome data scarcity limitations in Turkish legal NLP.

2.1.1.3 Evaluation and benchmarking

The impact of tokenization choices on Turkish NLP performance requires systematic evaluation. The Mukayese benchmark provides comprehensive evaluation across multiple tokenization approaches, while TR-MMLU offers large-scale assessment capabilities. These benchmarks revealed that optimal tokenization strategies vary significantly across tasks, with legal applications requiring specialized approaches that balance morphological decomposition with semantic preservation [24], [25] .

2.2 Text summarization in general, legal domain, and Turkish contexts

Text summarization indicates a critical application area where the challenges of Turkish morphology intersect with the precision requirements of legal document processing. Legal document summarization demands not only accurate content reduction but also preservation of legal semantics, precedent relationships, and procedural requirements that are essential for legal validity.

2.2.1 General text summarization methodologies

Contemporary text summarization approaches divide into extractive and abstractive methodologies. Each methodologies present distinct advantages for legal applications. Extractive summarization identifies and combines existing sentences from source documents and provide high fidelity to original content but limited flexibility in presentation. Abstractive summarization creates new text that captures essential information and offer better readability but raising concerns about factual accuracy critical in legal contexts. Recent developments in transformer based summarization models have demonstrated significant improvements across both approaches.

2.2.2 Turkish legal document summarization

Turkish legal document summarization combines the morphological complexity of Turkish with the specialized requirements of legal domain processing. Albayati and Fındık [26] developed the first automated summarization system specifically for Turkish legal texts which employe hybrid pre-trained approaches across Supreme Court, High Court, and District Court cases in 2025. Their research shows that domain-specific adaptation of pre-trained models can achieve satisfactory performance according to legal professional evaluation, though specific ROUGE metrics remain limited in the available literature.

The morphological richness of Turkish creates unique challenges for legal summarization. Turkish legal texts contain complex word formations that embed multiple layers of meaning within single tokens. This require summarization models to understand morphological decomposition while preserving legal semantic precision. Traditional summarization approaches that treat Turkish words as atomic units fail to catch the internal semantic structure necessary for accurate legal content reduction. Turkish legal document structure presents additional complexities for summarization systems. Legal decisions follow specific formats that include procedural history, legal reasoning, and conclusions. Effective Turkish legal summarization must protect this structural organization while accommodating the morphological complexity that affects every aspect of content processing.

2.2.3 Evaluation frameworks for legal summarization

Evaluation of legal document summarization requires specialized metrics beyond standard ROUGE scores. Legal professionals must assess both content accuracy and preservation of legal validity in summarized documents. Human evaluation by legal experts represents the gold standard for Turkish legal summarization assessment, though this approach limits scalability and introduces subjective variations in quality judgments. Current approaches combine automated metrics with expert evaluation, but standardized evaluation protocols specifically designed for Turkish legal document characteristics have not yet emerged in the literature. This gap represents a significant limitation for systematic comparison of different summarization approaches.

2.2.4 RAG architecture innovations for legal domain

The legal domain reveals unique requirements for RAG system design that distinguish these applications from general purpose implementations. LegalBench-RAG by Pipitone and Alami [27] introduced the benchmark specifically designed to evaluate retrieval components in legal RAG systems that emphasize precision over coverage in document retrieval. The benchmark's 6,858 human-annotated query-answer pairs demonstrate that legal RAG systems require minimal, highly relevant text segments rather than large imprecise chunks that characterize general RAG implementations.

Advanced RAG architectures for legal applications integrate multiple knowledge representation approaches. Barron et al. [28] developed novel combinations of vector stores, knowledge graphs and hierarchical non-negative matrix factorization for enhanced legal information retrieval. This hybrid approach addresses the complex relationships inherent in legal documents, spanning constitutions, statutes, regulations, and case law that require sophisticated semantic understanding beyond simple similarity matching.

Legal RAG systems must deal with the hierarchical nature of legal authority where constitutional requirements supersede statutory law which in turn supersedes regulatory guidelines. This legal hierarchy requires RAG systems to incorporate authority weighted retrieval that considers not only semantic relevance but also legal precedence

relationships when selecting supporting documents.

2.2.5 Large language model applications in legal contexts

The integration of large language models with legal domain knowledge creates a significant advancement in legal AI capabilities. One of this called Lawyer GPT shows comprehensive legal LLM development through external knowledge retrieval modules combined with supervised fine-tuning on GPT-4 Turbo-generated legal datasets [29]. The system's ability to handle various legal professional issues with enhanced factual accuracy through retrieval enhancement techniques represents a significant advancement toward practical legal AI deployment.

Legal knowledge graph construction using knowledge-enhanced LLMs addresses the challenge of representing multidimensional heterogeneous legal knowledge. Li et al. [30] developed the Joint Knowledge Enhancement Model (JKEM) for automated Chinese legal knowledge graph construction that achieve high recognition performance with 3,480 knowledge triples. This approach demonstrates how LLMs can be enhanced with prior domain knowledge through prefix-based fine-tuning to automate complex legal knowledge representation tasks.

The practical impact of legal LLMs has been demonstrated through strict empirical evaluation. Schwarcz et al. [31] conducted the first randomized controlled trial evaluating RAG-powered legal AI against control conditions that show statistically significant improvements in four of six legal tasks. The study's methodology provides strong evidence for practical benefits of RAG-powered legal AI systems including enhanced clarity and reduced hallucinations compared to both no-AI controls and pure LLM approaches.

2.2.6 Legal chatbots and conversational AI systems

Legal chatbots represent one of the most promising applications of RAG technology for expanding access to legal services. Beagle+ for British Columbia legal guidance and Alaska Virtual Assistant (AVA) demonstrate practical implementations using gen-

erative AI with domain-specific legal databases. These systems address critical access to justice issues by providing 24/7 legal information and step-by-step guidance for everyday legal problems that effectively reducing legal information gaps for underserved populations.

The technical architecture of legal chatbots requires careful balance between conversational capability and legal accuracy. RAG-enhanced legal chatbots maintain a factual foundation through document retrieval while providing accessible explanations of complex legal concepts, although challenges remain in ensuring consistent quality in diverse legal domains and jurisdictions [32].

2.2.7 Evaluation benchmarks and methodologies

Comprehensive evaluation of legal RAG systems requires specialized benchmarks that manage domain specific requirements. LegalBench by Guha et al. [33] provides 162 tasks covering six types of legal reasoning. It builts through interdisciplinary collaboration between legal professionals and computer scientists. The benchmark's evaluation of 20 open-source and commercial LLMs reveals significant performance variations across legal reasoning types that highlight the need for domain-specific evaluation rather than general NLP metrics .

The LAiW benchmark for Chinese legal LLMs categorizes capabilities into Basic Information Retrieval, Legal Foundation Inference, and Complex Legal Application levels. This hierarchical evaluation framework provides systematic assessment of legal AI capabilities from simple fact retrieval through complex legal reasoning that offer a template for developing similar evaluation frameworks for other legal systems and languages [34].

Evaluation of legal RAG systems must address the critical issue of hallucination in legal applications. Studies demonstrate that even GPT-4 hallucinates 49% or more of the time on basic legal tasks, necessitating RAG-based grounding for reliable legal AI applications. This finding emphasized the essential role of retrieval components in legal AI systems rather than treating RAG as merely an enhancement to pure language model approaches [27].

2.3 Large Language Models and legal domain adaptation

The adaptation of large language models to legal domains means a critical frontier in legal AI research where the precision requirements of legal practice intersect with the capabilities and limitations of current language technologies. The adaptation of the legal domain encounters unique challenges such as specialized terminology, complex reasoning requirements and the critical importance of factual accuracy that distinguishes legal applications from general NLP tasks [35].

2.3.1 Domain-specific pre-training approaches

Legal domain adaptation strategies have evolved from simple fine-tuning approaches to sophisticated pre-training methodologies that incorporate legal knowledge from the ground up. The result of specialized legal language models demonstrates significant performance improvements over general purpose models when applied to legal tasks. MultiLegalPile by Niklaus et al. [36] represents a significant advancement in legal domain pre-training resources, providing 689GB of multilingual legal data spanning 24 languages from 17 jurisdictions. The continued pre-training of XLM-R models on this domain-specific corpus achieved new state-of-the-art performance on the LEXTREME benchmark with 61.3% dataset aggregate score, demonstrating the effectiveness of large-scale legal domain adaptation.

The Legal-LM series and similar specialized models show the importance of legal domain knowledge integration during pre-training rather than relying only on downstream fine-tuning. These models incorporate legal reasoning patterns, citation structures, and domain-specific linguistic phenomena that cannot be effectively captured through adaptation of general-purpose language models alone.

2.3.2 Turkish legal language model development

Zeidi and Amasyalı [23] showed that Turkish legal BERT models can achieve competitive performance through modified pre-training strategies, even with limited training data (50MB-2GB legal corpus). Their approach achieved 72.15% F1-score on multi-

label legal text classification showing that domain-specific adaptation can overcome data scarcity limitations in specialized legal applications.

Turkish legal language models must also address the structural characteristics of Turkish legal documents which follow specific organizational patterns derived from continental European legal traditions. The hierarchical organization of Turkish legal texts requires language models that can understand document structure beyond surface level text processing [37].

2.3.3 Cross-lingual legal language models

Cross-lingual approaches to legal language modeling handle the challenge of limited resources in individual legal systems through transfer learning and multilingual model development. The development of multilingual legal language models faces challenges in balancing universal legal concepts with jurisdiction-specific legal systems. European legal AI initiatives, including the Lynx project creating multilingual legal knowledge graphs across 17 jurisdictions and 24 languages, demonstrate both the potential and complexity of cross-lingual legal AI development [38]. Rapid legal engineering for multilingual applications requires careful consideration of differences in the legal system and linguistic variations. Research on multilingual legal judgement prediction showed that effective prompt engineering can enable transfer learning across legal systems, though performance optimization requires understanding both linguistic and legal system characteristics [39].

2.4 Turkish NLP morphological processing and challenges

Oflazer's seminal work (1994) established the theoretical foundation for Turkish computational morphology through two-level description methods using PC-KIMMO formalism. This pioneering research provided a comprehensive framework for handling Turkish morphophonological processes, establishing principles that continue to influence Turkish NLP research three decades later [14]. The two-level approach addresses the complex relationship between surface forms and underlying morphological representations that characterizes Turkish word formation. Following developments

have purified and extended these foundational approaches. Çöltekin [40] introduced TRmorph which is the first freely available morphological analyzer for Turkish. TRmorph provide practical tools that enabled broader research community engagement. TRmorph's open-source distribution and comprehensive lexicon (54,000 root words) established a new standard for Turkish morphological processing, achieving processing speeds of 100,000 words per second through optimized trie structures and LRU caching. Modern morphological analyzers by Öztürel et al. [41] represent significant advances in coverage and accuracy. Their broad coverage model using OpenFst and Thrax grammars introduces novel fine POS tagsets and morphotactic representations that eliminate zero-derivations, providing more intuitive morphological analysis suitable for downstream applications.

2.4.1 Morphological disambiguation challenges

Morphological disambiguation means a critical blockage in Turkish NLP pipelines where multiple valid morphological analyses for a single word form must be resolved through contextual information. High accuracy systems achieve over 96% accuracy through sophisticated statistical approaches, but even small error rates compound significantly in downstream applications, particularly in precision-critical domains like legal text processing.

The perceptron-based approach developed by Sak et al. [42] achieved 96.80% accuracy using n-best list reranking with 23 carefully engineered features. This research demonstrated that morphological disambiguation benefits from feature engineering that captures both local morphological patterns and broader contextual information, establishing approaches that remain competitive with contemporary neural methods.

Neural approaches to morphological disambiguation have shown good results while reducing the need for manual feature engineering. Yıldız et al. [43] developed morphology aware networks that achieve high accuracy across Turkish, German, and French without language-specific feature engineering. This work shows that neural architectures can capture morphological disambiguation patterns across agglutinative languages. This suggests potential for cross-lingual transfer learning in morphological processing.

2.4.2 Stemming and lemmatization approaches

The Zemberek framework provides comprehensive morphological analysis, stemming, and spell checking capabilities, becoming widely adopted in the Turkish NLP community despite limitations in handling complex morphophonological processes [44].

In the realm of deep learning and contextual modeling, Sayallar [45] introduced a context-aware lemmatization and morphological tagging model tailored for Turkish. This model leverages bidirectional LSTM to capture orthographic features and Turkish BERT for contextual meaning, trained on the IMST and PUD datasets, and outperforms leading models from the SIGMORPHON 2019 competition.

Beyond Turkish-specific work, Toporkov and Agerri [46] examined the role of explicit morphological features in contextual lemmatization across morphologically rich languages, including Turkish. Surprisingly, they find that fine-grained morphological tags may not significantly enhance performance, suggesting that modern contextual embeddings might already encode much of the necessary morphological information implicitly.

Contemporary approaches to Turkish stemming integrate machine learning techniques with linguistic knowledge to improve accuracy. Recent research shows that hybrid approaches combining rule-based morphological analysis with statistical disambiguation achieve greatest performance compared to purely statistical or purely rule-based methods that particularly for out of vocabulary words common in specialized domains.

2.4.3 Parsing and syntactic analysis

Turkish dependency parsing requires specialized approaches that account for the morphological complexity and relatively free word order characteristic of Turkish syntax. Eryiğit et al. [20] showed that inflectional groups serve as more effective parsing units than surface word forms for Turkish. Their comprehensive investigation showed that sublexical units provide goof parsing accuracy for agglutinative languages and achieving significant improvements over word-level parsing approaches. The integration of

morphological analysis with syntactic parsing creates opportunities for improved performance through joint optimization. Contemporary parsing approaches that jointly optimize morphological disambiguation and syntactic analysis demonstrate reliable performance compared to pipeline approaches that treat these tasks independently which important for complex Turkish sentences with multiple morphological ambiguities.

2.4.4 Neural and transformer approaches

The source of transformer architectures has revolutionized Turkish morphological processing through pre-trained language models that can capture morphological patterns through self-attention mechanisms. BERTurk and its variants represent significant advances in Turkish NLP capabilities, achieving state-of-the-art performance across multiple tasks while handling morphological complexity through subword tokenization approaches [47]. Recent research on vocabulary size optimization for Turkish BERT models reveals task-dependent optimal configurations. Systematic investigations indicate that NER tasks benefit from larger vocabularies (128k tokens) while sentiment analysis optimizes at smaller vocabulary sizes (64k). This values indicate that morphological processing strategies must be tailored to specific downstream applications. The effectiveness of transformer approaches for Turkish morphological processing arises from their ability to learn morphological patterns implicitly through large scale pre-training. However, explicit morphological knowledge can still provide benefits, particularly for low-resource specialized domains where pre-training data may not adequately cover domain-specific morphological sensations.

2.5 Legal NLP applications and cross-lingual approaches

Legal NLP applications mean one of the most challenging and consequential domains for natural language processing research where the precision requirements of legal practice intersect with the complexity of natural language understanding. Cross-lingual approaches in legal NLP handle the fundamental challenge of legal system diversity while enabling knowledge transfer across jurisdictions, legal traditions, and

languages.

2.5.1 Comprehensive legal AI applications

The landscape of legal AI applications has evolved from simple document classification to sophisticated systems capable of legal reasoning, judgment prediction, and case law analysis. Zhong et al. [35] provided a foundational survey that establishes taxonomic classification of legal AI tasks from dual perspectives, bridging the gap between legal professionals' rule-based approaches and NLP researchers' data-driven methods. This framework identifies different key challenges such as data scarcity, domain adaptation and evaluation metrics that continue to shape legal NLP research directions.

Legal judgment prediction represents one of the most well designed applications of NLP in legal domains. Cui et al. [48] performed systematic analysis of 31 LJP datasets across 6 languages. This reveals significant variations in task formulation, evaluation metrics, and performance across different legal systems. The review of 14 evaluation metrics across four categories demonstrates the complexity of assessing legal AI systems where standard NLP metrics often fail to capture the precision requirements of legal applications. Recent advancements in legal fact prediction address practical limitations of existing approaches by focusing on early-stage litigation support. The Legal Fact Prediction (LFP) task formulation enables prediction before legal facts are formally established, addressing critical practical limitations in legal AI by supporting legal professionals during evidence evaluation phases rather than requiring predetermined legal facts.

2.5.2 Cross-lingual legal information retrieval

The development of multilingual legal knowledge graphs exemplified by the Lynx project spanning 17 jurisdictions and 24 languages that demonstrates both the potential and complexity of cross-lingual legal AI systems. Some works show how semantic web technologies can break accessibility and language barriers in European legislation and case law. These approaches enabled legal professionals to access relevant legal

information regardless of the original language of legal documents, supporting comparative legal analysis and cross-jurisdictional legal research [49]. The challenges of cross-lingual legal information retrieval extend beyond simple translation to contain differences in legal concepts, procedural requirements, and jurisdictional boundaries. Effective cross-lingual legal systems must account for legal system differences while providing accurate translations of legal concepts that may not have direct equivalents across legal traditions.

2.5.3 Multilingual legal document processing

Multilingual legal document processing requires sophisticated approaches that can handle both linguistic diversity and legal system variations. The LEXTREME benchmark Niklaus et al. [36] comprehensive evaluation across 11 datasets and 24 languages, revealing that even state-of-the-art multilingual models achieve only 61.3% dataset aggregate score, indicating significant room for improvement in multilingual legal NLP.

Annotation projection techniques provide another approach to cross-lingual legal NLP development. Cross-lingual annotation projection enables transfer of legal annotations across languages, potentially reducing the annotation burden for developing legal NLP systems in low-resource languages while maintaining annotation quality through sophisticated projection algorithms.

2.5.4 Legal information extraction across languages

The development of legal named entity recognition systems requires careful consideration of legal system distinguishes where entity types and relationships may vary significantly across jurisdictions. Contemporary techniques to multilingual legal information extraction integrate transformer based models with legal domain knowledge to achieve competitive performance across languages. However, evaluation reveals significant performance variations across languages and legal systems highlighting the need for specialized approaches that accessible for both linguistic and legal system characteristics.

2.5.5 Turkish integration in multilingual legal systems

Turkish legal NLP presents unique challenges within multilingual legal systems due to the agglutinative nature of Turkish combined with the continental European legal tradition that influences Turkish legal language. Research on Turkish legal named entity recognition Çetindağ et al. [50] showed that specialized approaches are necessary to handle the morphological complexity of Turkish legal terminology while maintaining accuracy in entity identification. Turkish court decision prediction systems achieve high accuracy (90%+ for Constitutional Court decisions) through deep learning approaches, but integration with multilingual legal systems requires additional research on cross-lingual transfer learning and comparative legal analysis capabilities that account for Turkish legal system characteristics.

Furthermore, recent work like LegalTurk, which adapts BERT for legal Turkish via modified pre-training strategies (e.g. altering MLM/NSP tasks), significantly improves NER and multi-label text classification in legal contexts—suggesting domain-adaptive language modeling as a promising path for cross-lingual transfer [23].

In broader multilingual legal NLP, the MULTI-EURLEX dataset (EU legal documents in 23 languages with EUROVOC annotations) serves as a benchmark for zero-shot cross-lingual transfer; leveraging multilingual models like XLM-RoBERTa or mT5 with adaptation strategies (adapters, BITFIT, LNFIT) aids transfer effectiveness—including for Turkish as part of the labeled languages [51].

Lastly, transfer learning methods in tasks like NLI involving Turkish (e.g. English Turkish in XNLI) show that custom loss functions can produce up to 10% performance gains, implying that judicious adaptation approaches can meaningfully support cross-lingual inference for legal tasks [52].

2.5.6 Evaluation frameworks for cross-lingual legal NLP

Cross-lingual legal NLP demands evaluation frameworks that go beyond standard NLP metrics—since preserving legal precision, concept consistency, and translation fidelity are paramount.

To date, some evaluation combines automated metrics with expert validation, but lacks standardization. No widely-accepted protocols for legal multilingual evaluation yet exist, increasing variability across jurisdictions and undermining reproducibility.

Models like LegalTurk show improved performance on NER and multi-label classification, but assessment remains primarily metric-driven without cross-jurisdiction validation. Similarly, resources like MultiEURLEX enable multilingual classification benchmarking but still focus on topic-level tasks rather than legal concept accuracy across languages [51].

Emerging evaluation paradigms in NLP and ML, such as "Translate-and-Test" transfer mechanisms, emphasize pipeline modularity and task-specific evaluation—ideas possibly adaptable to legal settings for better clarity and reproducibility [53].

2.6 Research gaps and future directions

The summary of recent work in Turkish legal NLP reveals important successes as well as ensuing gaps that establish the focus of future studies. Although the performance of individual Turkish legal NLP components is promising, little is known about how to integrate them into comprehensive systems that can support actual legal practice.

2.6.1 Dataset and corpus development priorities

The lack of comprehensive Turkish legal corpora represents the most significant limitation constraining Turkish legal NLP research. Current Turkish legal datasets range from 30-50MB to 2GB, substantially smaller than corpora available for high-resource languages and domains, limiting the effectiveness of data intensive approaches that have proven successful in general NLP applications. Public availability of Turkish legal canons remains severely limited due to privacy concerns, access restrictions, and the high cost of legal expert annotation. A crucial infrastructure need for furthering research is the creation of publicly accessible Turkish legal corpora with extensive annotation schemes which calls for cooperation between government organizations, legal experts, and academic institutions. As of yet, there is no community standard

for standardizing annotations across Turkish legal corpora. Standardized annotation schemes covering legal entity types, document structure, precedent relationships, and cross-references should be established by future corpus development efforts. This will allow for the systematic comparison of various methodologies and the production of reproducible research findings. Since legal language and concepts change over time, special attention must be paid to the temporal dimension of legal corpus development. Comprehensive Turkish legal corpora should span sufficient temporal ranges to enable analysis of legal language evolution while maintaining contemporary relevance for current legal practice applications.

2.6.2 Evaluation methodology advancement

The development of legal domain specific evaluation metrics that incorporate expert judgment, legal validity assessment and also practical utility measures represents a critical research priority. Human evaluation by legal experts provides the most reliable assessment of Turkish legal NLP systems, but standardized protocols for expert evaluation have not been established. Future research should develop systematic expert evaluation frameworks that control for subjective variations while providing reliable assessment of legal NLP system quality and practical utility. Cross-system evaluation capabilities remain limited. The establishment of standardized evaluation protocols and benchmark datasets would enable more strict research and accelerate progress through systematic comparison of competing approaches. Fairness and bias evaluation in Turkish legal NLP systems requires specialized approaches that account for both linguistic and legal system characteristics. Research on bias detection and mitigation in Turkish legal AI systems should address both statistical bias in model predictions and potential perpetuation of historical biases present in legal training data.

2.6.3 Technical architecture innovations

Current approaches typically treat morphological processing and legal domain adaptation as separate components, potentially missing opportunities for joint optimization

that could improve both morphological accuracy and legal semantic understanding. Long document processing capabilities remain limited for Turkish legal applications, where legal documents frequently exceed transformer token limits. Technical innovations in hierarchical processing, document segmentation strategies, and efficient attention mechanisms specifically designed for Turkish legal document characteristics could significantly improve practical applicability. Real-time processing capabilities for Turkish legal applications have received limited research attention. The development of efficient architectures capable of processing Turkish legal documents in real-time would enable interactive legal AI applications and integration with legal practice workflows. Multimodal approaches that combine textual analysis with structured document elements, citations, and legal metadata remain largely unexplored for Turkish legal applications. Integration of textual NLP with document structure analysis, citation networks, and legal knowledge graphs could supply more comprehensive legal document understanding capabilities.

2.6.4 Cross-lingual integration opportunities

Cross-lingual transfer learning approaches specifically designed for legal domains could address data scarcity limitations in Turkish legal NLP while contributing to multilingual legal AI research. Comparative legal analysis capabilities that enable systematic comparison of legal concepts, procedures, and outcomes across different legal systems remain underdeveloped. Turkish legal NLP systems that can facilitate comparative analysis with other legal systems would provide valuable capabilities for legal scholarship and international legal cooperation. Legal concept mapping across languages and legal systems represents a challenging but potentially valuable research direction. The effect of cross-lingual legal concept ontologies that map Turkish legal concepts to broader international legal frameworks could enable integration with existing multilingual legal AI systems.

2.6.5 Practical implementation challenges

Research on user interface design, legal professional training requirements and integration with existing legal technology infrastructure describes important areas for future

investigation. Ethical considerations specific to Turkish legal AI applications require systematic investigation. Research on accountability, transparency, and professional responsibility standards for Turkish legal AI systems should address both technical implementation challenges and legal professional conduct requirements. Regulatory compliance for Turkish legal AI applications will require specialized research as AI regulations develop in Turkey and internationally. Future research should handle compliance requirements while developing technical approaches that maintain both AI system effectiveness and regulatory adherence.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter presents the comprehensive methodological framework developed for Turkish legal natural language processing. The methodology contains multiple components designed to manage the unique challenges of processing Turkish legal texts. This chapter includes their morphological complexity, domain-specific terminology and hierarchical document structures. We detail the mathematical foundations, algorithmic approaches, and evaluation metrics used throughout this research.

3.2 Data Collection and Preprocessing

The data collection stage describes the sources of Turkish legal texts and the web crawling framework used to gather them. The preprocessing stage covers text cleaning, normalization, and structuring to ensure consistency and usability. This part also include the removal of duplicates, correction of encoding issues and organization of documents with relevant metadata. Finally, the tokenization stage presents the methods applied to segment text into processable units comparing character-level, word-based, subword and morphological approaches.

3.2.1 Web Crawling Framework

The data collection methodology utilizes an automated web crawling system designed specifically for legal document acquisition. We first introduce the notation:

$K = \{k_1, k_2, \ldots, k_n\}$ The set of legal keywords. Each k_i is a single term or concept (e.g. “appeal”, “precedent”, “ruling”).

$U = \{u_1, u_2, \ldots, u_m\}$ The set of target URLs (web pages or domains) to be crawled.

q_i A search query generated from keyword k_i. In practice, q_i is formed by plugging k_i into a query template such as

$$q_i = \text{“}\texttt{site:}\ u_j\ +\ k_i\text{”} \quad \text{for each } u_j \in U.$$

d_j The j-th document retrieved by running queries q_i.

m_j Metadata associated with document d_j (e.g. author, decision date, case number).

Storage Tuple (d_j, m_j, k_i) Records that document d_j, with metadata m_j, was found via keyword k_i.

The crawling process then follows this iterative procedure:

1. **Initialization**: Load the list of target URLs

   $$U = \{u_1, u_2, \ldots, u_m\}$$

   into the web scraper configuration.

2. **Search Query Generation**: For each keyword $k_i \in K$, build the search query q_i and apply it to each $u_j \in U$.

3. **Document Retrieval**: Run each q_i on the configured URLs and extract every resulting document d_j along with its metadata m_j.

4. **Storage**: For each retrieved document, store the triple

   $$\big(d_j,\ m_j,\ k_i\big)$$

   in a structured database or file system.

3.2.2 Document Preprocessing Pipeline

The preprocessing pipeline transforms raw legal texts into structured representations suitable for computational analysis. Formally, for each document d, we define:

$$\begin{aligned} T &= \mathrm{Tokenize}(d) = \big(t_1, t_2, \ldots, t_L\big), \\ C &= \mathrm{Clean}(T) = \big(t_i \mid t_i \in T,\ \psi(t_i) = 1\big), \\ N &= \mathrm{Normalize}(C) = \big(\nu(t_i) \mid t_i \in C\big), \\ \mathrm{Preprocess}(d) &= N. \end{aligned}$$

where:

$\mathrm{Tokenize}(d)$ Splits the raw text d into a sequence of L tokens $\{t_1, t_2, \ldots, t_L\}$ (e.g., words, punctuation, or subwords).

$\psi(t)$ A boolean filter function:

$$\psi(t) = \begin{cases} 1, & \text{if } t \text{ is not noise e.g. not HTML, not a stopword, length} \geq 2, \\ 0, & \text{otherwise.} \end{cases}$$

$\mathrm{Clean}(T)$ Retains only those tokens t_i for which $\psi(t_i) = 1$, removing irrelevant or noisy elements.

$\nu(t)$ A normalization mapping defined by

$$\nu(t) = \mathrm{Lemma}\big(\mathrm{Lowercase}(t)\big),$$

which converts t to lower-case, applies lemmatization (e.g. "argued"→"argue"), and optionally strips diacritics or punctuation.

$\mathrm{Normalize}(C)$ Applies ν to each token in C, producing the final, standardized token sequence N.

In this context, noise refers to any token in the raw legal text that does not carry a meaningful contribution to the intended NLP task and may hinder model performance. Typical noise categories include HTML or formatting tags, irrelevant special characters or symbols, excessively short, non-informative strings, high-frequency stopwords

that rarely convey discriminative legal meaning and non-textual elements such as page numbers, clause identifiers, or OCR artifacts.

3.2.3 Tokenization Methods

Tokenization is a necessary and crucial step in LLM because LLMs and other NLP models cannot directly process raw text as humans read it. They work with numerical representations of smaller text units. Tokenization is the step that breaks text into these units (tokens), which can be words, subwords, characters, or even byte-pair encodings depending on the method. In this study, we examine various tokenization methods that include Character-Level Tokenization, Word-Level Tokenization, Morphological Tokenization, Legal-Aware Tokenization, Byte-Pair Encoding (BPE), SentencePiece Tokenization, and Compression Ratio Calculation.

3.2.3.1 Character-Level Tokenization

Character-level tokenization treats each individual symbol in the text as a token. Let

$$d = c_1 c_2 \dots c_n, \quad c_i \in \Sigma$$

be the raw text of document d, where Σ is the alphabet of all characters (letters, digits, punctuation, spaces, etc.). We define the character-level token sequence as

$$T_{\text{char}}(d) \;=\; \big(c_1,\; c_2,\; \dots,\; c_n\big).$$

In this setting, Σ^* denotes the set of all possible strings that can be formed from the alphabet Σ, including the empty string. Thus, the function

$$T_{\text{char}} \colon \Sigma^* \;\longrightarrow\; \Sigma^n$$

maps any string $d \in \Sigma^*$ to the ordered list of its n constituent characters. This method preserves every mark in the text but can lead to very long token sequences.

Consider the sentence: *"Davalı, Borçlar Kanunu'nun 112. maddesi uyarınca sorumludur."*

Character-Level Tokenization

["D", "a", "v", "a", "l", "ı", ",", " ", "B", "o", "r", "ç", "l", "a", "r", " ", "K", "a", "n", "u", "n", "u", "'", "n", "u", "n", " ", "1", "1", "2", ".", " ", "m", "a", "d", "d", "e", "s", "i", " ", "u", "y", "a", "r", "ı", "n", "c", "a", " ", "s", "o", "r", "u", "m", "l", "u", "d", "u", "r", "."]

3.2.3.2 Word-Level Tokenization

Word-level tokenization segments text at whitespace and punctuation boundaries. Formally, let $\mathcal{R}$ be a regular expression matching "word" patterns, for example

$$\mathcal{R} = \texttt{[A-Za-zÇĞİÖŞÜçğıöşü]+}.$$

Then

$$T_{\text{word}}(d) \;=\; \big(w_1,\; w_2,\; \ldots,\; w_m\big)$$

where each

$$w_i = \text{the } i\text{-th maximal substring of } d \text{ matching } \mathcal{R},$$

and m is the total number of such word tokens. The map

$$T_{\text{word}} \colon \Sigma^* \;\longrightarrow\; \{\text{words}\}^m$$

thus sends any input string $d \in \Sigma^*$ to its ordered list of word tokens. This give a more compact tokenization than character-level, while ignoring punctuation and splitting on spaces.

For example:

["Davalı", "Borçlar", "Kanunu'nun", "112", "maddesi", "uyarınca", "sorumludur"]

3.2.3.3 Morphological Tokenization

Given Turkish's agglutinative morphology, we decompose each word into a root and one or more suffixes. Define a morphological analyzer

$$M \colon \{\text{words}\} \;\longrightarrow\; R \times S^*$$

where R is the set of possible roots and S the set of possible suffixes. For any word w,

$$M(w) \;=\; \big(\mathrm{root}(w),\; \mathrm{suffix}_1(w), \ldots, \mathrm{suffix}_k(w)\big).$$

Then we define

$$T_{\mathrm{morph}}(w) \;=\; \big(r,\; s_1,\; \ldots,\; s_k\big) \;\text{ with }\; (r, s_1, \ldots, s_k) = M(w).$$

This produces a sequence of $k+1$ morphemes for each word, capturing grammatical information explicitly. Extending to whole documents, the map

$$T_{\mathrm{morph}} \colon \{\text{words}\}^m \;\longrightarrow\; (R \cup S)^*$$

assigns to each sequence of word tokens $(w_1, \ldots, w_m)$ a flattened sequence of morphemes, i.e.,

$$T_{\mathrm{morph}}(d) \;=\; \big(T_{\mathrm{morph}}(w_1),\; T_{\mathrm{morph}}(w_2),\; \ldots,\; T_{\mathrm{morph}}(w_m)\big).$$

For instance,

> ["dava", "-lı", "borç", "-lar", "kanun", "-u", "-nun", "112", "madde", "-si", "uyar", "-ınca", "sorumlu", "-dur"]

3.2.3.4 Legal-Aware Tokenization

To capture domain-specific patterns (case citations, statute references, institutional names), we augment word tokens with special legal patterns. Let

$$P_{\mathrm{legal}}(d) \;=\; \bigcup_{p \in \mathcal{P}} \mathrm{extract}_p(d)$$

where $\mathcal{P}$ is the set of pattern types (e.g. case references, law citations, institutional names) and $\mathrm{extract}_p$ returns all substrings matching pattern p. Then

$$T_{\mathrm{legal}}(d) \;=\; T_{\mathrm{word}}(d) \;\oplus\; P_{\mathrm{legal}}(d),$$

where $\oplus$ denotes sequence concatenation (or multiset union if order is not required). Example patterns include:

- **Case reference:** “Yargıtay 4. HD 2023/1234”

- **Law citation:** “5237 sayılı TCK md. 53”
- **Institution:** “Danıştay”, “Bakanlar Kurulu”

As an example,

> ["Davalı", "<LAW: Borçlar Kanunu>", "<ARTICLE: 112. maddesi>", "uyarınca", "sorumludur"]

3.2.3.5 Byte-Pair Encoding (BPE)

Byte-Pair Encoding builds subword units by iteratively merging the most frequent adjacent symbol pairs. Let

$$V_0 = \Sigma \quad \text{and} \quad T_0(d) = T_{\text{char}}(d).$$

At iteration t:

$$(a_t, b_t) = \arg \max_{(a,b) \in V_{t-1}^2} \text{freq}_{t-1}(a, b),$$
$$V_t = V_{t-1} \cup \{ a_t b_t \},$$
$$T_t(d) = \text{merge}(T_{t-1}(d),\, a_t b_t),$$

where $\text{freq}_{t-1}(a, b)$ counts how often the pair (a, b) appears in $T_{t-1}(d)$, and merge replaces every occurrence of (a_t, b_t) by the new symbol $a_t b_t$. This repeats until $|V_T| = V_{\max}$.

Algorithm 1: Byte-Pair Encoding (BPE)

[1] $V \leftarrow \Sigma$ Initialize vocabulary with alphabet $T \leftarrow T_{\text{char}}(\text{corpus})$ **while** $|V| < V_{\max}$ **do**

$(a,\, b) \leftarrow \arg \max_{(x,\, y) \in V^2} \text{freq}(x,\, y)\; V \leftarrow V \cup \{ab\}\; T \leftarrow \text{merge}(T, ab)$

For example,

> ["Da", "valı", "Borç", "lar", "Kanun", "u’nun", "112", "madde", "si", "uyarın", "ca", "sorumlu", "dur"]

3.2.3.6 SentencePiece Tokenization

SentencePiece models tokenization as finding a subword sequence $S = (s_1, \ldots, s_L)$ that maximizes a language model probability. Formally,

$$\hat{S} = \arg \max_{S:\, s_1 \cdots s_L = d} P(S) = \arg \max_{S} \prod_{i=1}^{L} P\big(s_i \mid s_1, \ldots, s_{i-1}\big),$$

where each s_i is a subword from a fixed vocabulary, and d is the concatenation of all s_i.

For the same sentence, an example tokenization could be:

> ["_Davalı", "_Borçlar", "_Kanunu", "'nun", "_112", "_maddesi", "_uyarınca", "_sorumlu", "dur"]

Here the "_" symbol denotes a preceding whitespace, which allows SentencePiece to tokenize text without requiring explicit pre-segmentation by spaces.

3.2.3.7 Compression Ratio Calculation

To compare efficiency, we define the compression ratio of a tokenization T on document d as

$$\mathrm{CR}(d) = \frac{|T(d)|}{|d|},$$

where $|T(d)|$ is the number of tokens produced and $|d|$ the number of characters in d. Lower CR indicates a more compact representation.

3.3 Document Embedding Methods

Embedding is one of the crucial step in NLP tasks because LLMs and other NLP systems cannot work directly with tokens as plain symbols. They need a way to represent each token numerically in a form that preserves its meaning and relationships to other tokens.

3.3.1 Count Vectorizer

Count Vectorizer maps each document d to a sparse vector of term counts over a fixed vocabulary $V = \{t_1, t_2, \ldots, t_p\}$. Formally, we define:

$$\mathrm{CountVec}(d) \;=\; \big[v_1, v_2, \ldots, v_p\big]^\top,$$

where each component is

$$v_i = f(t_i, d),$$

and $f(t_i, d)$ is the number of times term t_i appears in document d. In practice, this yields a vector in $\mathbb{R}^p$ where most entries are zero (sparsity) when p is large.

3.3.2 TF-IDF (Term Frequency–Inverse Document Frequency)

TF-IDF reweights raw counts by term importance across the corpus D. For term t in document d:

$$\mathrm{tf}(t, d) = \frac{f(t, d)}{\sum_{t' \in V} f(t', d)}, \quad \text{(normalized frequency of } t \text{ in } d) \tag{3.1}$$

$$\mathrm{idf}(t, D) = \log\!\left(\frac{|D|}{|\{\, d' \in D : t \in d' \}|}\right), \quad \text{(rarity of } t \text{ in corpus)} \tag{3.2}$$

$$\mathrm{TFIDF}(t, d, D) = \mathrm{tf}(t, d) \;\times\; \mathrm{idf}(t, D). \tag{3.3}$$

The document embedding is then

$$\mathrm{TFIDFVec}(d) \;=\; \big[\mathrm{TFIDF}(t_1, d, D), \ldots, \mathrm{TFIDF}(t_p, d, D)\big]^\top.$$

This representation down-weights common terms (stop-words) and highlights discriminative ones.

3.3.3 Latent Semantic Analysis (LSA)

LSA reduces dimensionality by applying Singular Value Decomposition (SVD) to the term–document matrix $A \in \mathbb{R}^{p \times |D|}$, where

$$A_{i,j} = \mathrm{TFIDF}(t_i, d_j, D).$$

Compute the SVD:

$$A = U\,\Sigma\,V^{\top}, \tag{3.4}$$

with

- $U \in \mathbb{R}^{p\times p}$: left singular vectors (terms),
- $\Sigma \in \mathbb{R}^{p\times |D|}$: singular values on the diagonal,
- $V \in \mathbb{R}^{|D|\times |D|}$: right singular vectors (documents).

To obtain a rank-k approximation, keep only the top k singular values:

$$A_k = U_k\,\Sigma_k\,V_k^{\top}, \tag{3.5}$$

where $U_k \in \mathbb{R}^{p\times k}$, $\Sigma_k \in \mathbb{R}^{k\times k}$, and $V_k \in \mathbb{R}^{|D|\times k}$. Each document d_j is then embedded as the j-th row of $V_k\,\Sigma_k \in \mathbb{R}^{|D|\times k}$.

3.3.4 Principal Component Analysis (PCA) Embedding

PCA identifies orthogonal directions of maximal variance in the TF-IDF space. Given the data matrix $X \in \mathbb{R}^{|D|\times p}$ whose rows are $\mathrm{TFIDFVec}(d)$, we:

Center the data:

$$X_c = X - \mathbf{1}\,\mu^{\top}, \quad \mu = \frac{1}{|D|}\sum_{j=1}^{|D|} X_{j,*}.$$

Compute covariance:

$$C = \frac{1}{|D|-1}\,X_c^{\top}X_c.$$

Eigen decomposition:

$$C\,v_i = \lambda_i\,v_i, \quad i = 1,\ldots,p,$$

where $\lambda_1 \geq \lambda_2 \geq \ldots$ are eigenvalues and v_i the corresponding orthonormal eigenvectors. Project to k dimensions:

$$Y = X_c\,W, \quad \text{with } W = [\,v_1,\ldots,v_k\,] \in \mathbb{R}^{p\times k}.$$

Each row of $Y \in \mathbb{R}^{|D|\times k}$ is the k-dimensional PCA embedding of a document.

3.3.5 Non-negative Matrix Factorization (NMF)

NMF factorizes the non-negative term–document matrix $A \in \mathbb{R}_{\geq 0}^{p \times |D|}$ into two low-rank non-negative factors:

$$A \approx W H, \tag{3.6}$$

where $W \in \mathbb{R}_{\geq 0}^{p \times k}$ and $H \in \mathbb{R}_{\geq 0}^{k \times |D|}$. We solve:

$$\min_{W,H \geq 0} \left\| A - WH \right\|_F^2, \tag{3.7}$$

with $\|\cdot\|_F$ denoting the Frobenius norm. The columns of H give the k-dimensional embeddings of the documents.

3.3.6 Composite Score Calculation

The evaluation of embedding models is a multifaceted problem. Single metric is often insufficient to capture all aspects of a model's quality. It is widely accepted in the literature that a holistic assessment requires evaluating models not only on their performance in specific applications, but also on their intrinsic properties and practical efficiency. For instance, Wang et al. [54] argue that a comprehensive understanding of model quality requires the joint consideration of both intrinsic metrics (e.g., semantic coherence) and extrinsic metrics.

This principle of multi-dimensional evaluation is put into practice by modern benchmarking platforms such as the Massive Text Embedding Benchmark (MTEB). Muennighoff et al. [55] state that MTEB evaluates models in a wide range of tasks, including classification, clustering, and retrieval, to provide a comprehensive overview of their capabilities. This approach allows for a fair comparison of models that may have different strengths and weaknesses.

Building upon these established methodologies, we introduce a weighted composite score to quantitatively compare the embedding methods examined in our study. This score synthesizes performance across three key dimensions:

To compare embedding methods across multiple criteria, we define a weighted com-

posite score for each method after trying several combination of weigths:

$$\text{Score} = 0.4\,S + 0.3\,C + 0.3\,E, \tag{3.8}$$

where:

S Silhouette coefficient measuring cluster separation.

C Semantic coherence (e.g. average pairwise cosine similarity within topics).

E Efficiency score, defined as

$$E = \frac{1}{1 + \log\big(\text{Time} \times \text{Memory}/1000\big)},$$

which penalizes large computation time and memory usage.

Lower E-denominator implies higher efficiency; the log-scale ensures diminishing penalties.

3.4 Legal Document Summarization

Legal texts are often long, complex, and filled with technical terminology, making them difficult and time-consuming to read in full. Summarization techniques, especially when tailored to legal language, help by condensing the content while preserving essential facts, legal reasoning, and conclusions.

3.4.1 Extractive Summarization Framework

Let a legal document be segmented into N sentences $\mathcal{S} = \{s_1, \ldots, s_N\}$. We compute an overall score for each sentence s_i as a weighted sum of five components:

$$\text{Score}(s_i) = \alpha\,\text{KW}(s_i) + \beta\,\text{Coh}(s_i) + \gamma\,\text{Pos}(s_i) + \delta\,\text{Len}(s_i) + \epsilon\,\text{Legal}(s_i) \tag{3.9}$$

$$\alpha + \beta + \gamma + \delta + \epsilon = 1.$$

where the weights $\alpha, \beta, \gamma, \delta, \epsilon$ balance term importance, coherence, positional bias, length suitability, and legal relevance.

3.4.1.1 Keyword Importance Score

We sum TF–IDF weights of words in s_i, boosting recognized legal terms:

$$\mathrm{KW}(s_i) = \sum_{w \in s_i} \mathrm{TFIDF}(w) \;\times\; \big[1 + \lambda\, \mathbf{1}_{w \in \mathcal{L}}\big], \tag{3.10}$$

where

- $\mathrm{TFIDF}(w)$ is the TF–IDF weight of word w.
- $\mathcal{L}$ is the set of domain-specific legal terms.
- $\mathbf{1}_{w \in \mathcal{L}} = 1$ if w is legal, else 0.
- $\lambda > 0$ controls the bonus strength.

3.4.1.2 Coherence Score

We measure semantic cohesion by averaging pairwise cosine similarities of word embeddings:

$$\mathrm{Coh}(s_i) = \frac{2}{|s_i|\,\big(|s_i| - 1\big)} \sum_{\substack{w_j, w_k \in s_i \\ j<k}} \mathrm{cosine}\big(\mathbf{e}(w_j),\, \mathbf{e}(w_k)\big), \tag{3.11}$$

where $\mathbf{e}(w)$ is the embedding of w. Higher values indicate a more thematically focused sentence.

3.4.1.3 Positional Weight

Sentences near the beginning or end often carry key information. Define

$$\mathrm{Pos}(s_i) = \begin{cases} p_{\mathrm{head}}, & i \le \lceil 0.1\, N \rceil, \\ p_{\mathrm{tail}}, & i > N - \lceil 0.1\, N \rceil, \\ p_{\mathrm{mid}}, & \text{otherwise}, \end{cases}$$

with typical choices $p_{\mathrm{head}} > p_{\mathrm{tail}} > p_{\mathrm{mid}}$ (e.g. 1.5, 1.3, 1.0).

3.4.1.4 Length Normalization

To penalize sentences that are too short or too long, we normalize by deviation from the average length:

$$\mathrm{Len}(s_i) = 1 \; - \; \frac{\big|\, |s_i| - L_{\mathrm{avg}} \big|}{L_{\max} - L_{\min}}, \tag{3.12}$$

In this formulation, length normalization is introduced to prevent bias toward sentences that are either disproportionately short or excessively long, both of which can reduce the informativeness and readability of summaries. Very short sentences often lack sufficient semantic content to be representative of the source text, while very long sentences may include redundant or tangential information that cuts key points. By centering the normalization around the document's *average sentence length* (L_{avg}), we explicitly reward sentences whose lengths are statistically typical for that document's discourse style, thus aligning selection with the text's natural rhythm. The scaling factor $(L_{\max} - L_{\min})^{-1}$ ensures comparability across documents with different sentence length ranges, preventing bias toward particular corpora or genres. The absolute deviation $|\,|s_i| - L_{\mathrm{avg}}\,|$ measures how far a candidate sentence diverges from the norm, and subtracting this proportion from 1 in $\mathrm{Len}(s_i)$ maps the score to the $[0, 1]$ interval, where 1 represents an ideal length. This approach is particularly suitable in extractive summarization, as it preserves *content density and coherence* while avoiding the over-selection of structurally unbalanced sentences that may harm both informativeness and fluency.

3.4.1.5 Legal Terminology Factor

We boost sentences with a high density of legal terms:

$$\mathrm{Legal}(s_i) = \frac{\sum_{w \in s_i} \mathbf{1}_{w \in \mathcal{L}}}{|s_i|}, \tag{3.13}$$

where $\mathbf{1}_{w \in \mathcal{L}}$ indicates membership in the legal lexicon.

Finally, sentences are ranked by $\mathrm{Score}(s_i)$ (Eq. 3.9) and the top-K are selected to form the extractive summary.

3.4.2 Baseline Summarization Methods

Baseline methods serve as fundamental reference points for evaluating the effectiveness of more advanced summarization techniques. They are typically simple, computationally efficient, and interpretable. Thus, they are useful for establishing lower bounds on performance. Although these methods may not capture complex semantic relations, they provide an essential benchmark for comparison.

3.4.2.1 TextRank Algorithm

TextRank is a graph-based ranking algorithm adapted from PageRank for extractive summarization. We first build a sentence-similarity graph:

$$G = (V, E), \tag{3.14}$$

where

- $V = \{s_1, s_2, \ldots, s_n\}$ is the set of all sentences in the document.
- $E = \{\, w_{ij} : s_i, s_j \in V\}$ is the set of weighted edges, with

$$w_{ij} = \text{Similarity}(s_i, s_j).$$

We define the edge weight (sentence similarity) by:

$$\text{Similarity}(s_i, s_j) = \frac{|W_i \cap W_j|}{\log\big(|W_i|\big) \;+\; \log\big(|W_j|\big)}, \tag{3.15}$$

where

- W_i and W_j are the sets of unique words in s_i and s_j, respectively.
- $|W_i \cap W_j|$ counts their shared words.
- $|W_i|$ is the number of words in s_i.

Once G is built, we compute the TextRank score $TR(s_i)$ for each sentence by iterating:

$$TR(s_i) = (1 - d) + d \sum_{s_j \in \text{In}(s_i)} \frac{w_{ji}}{\sum_{s_k \in \text{Out}(s_j)} w_{jk}} \, TR(s_j), \tag{3.16}$$

where

- $d \in (0, 1)$ is the damping factor (commonly 0.85).
- $\mathrm{In}(s_i)$ is the set of sentences that link *to* s_i.
- $\mathrm{Out}(s_j)$ is the set of sentences that s_j links *to*.
- w_{ji} is the weight on the edge from s_j to s_i.

After convergence, sentences are ranked by $TR(s_i)$ and the top-K are selected.

3.4.2.2 Frequency-Based Summarization

This simple baseline ranks sentences by the average frequency of their words.

First, compute the normalized term frequency:

$$f(w) = \frac{\text{count of } w \text{ in document}}{\text{total number of words in document}}. \tag{3.17}$$

Then for each sentence s_i with $|s_i|$ words:

$$\mathrm{Score}_{\mathrm{freq}}(s_i) = \frac{1}{|s_i|} \sum_{w \in s_i} f(w). \tag{3.18}$$

Higher scores indicate sentences containing more frequent (and thus potentially important) terms.

3.4.2.3 Position-Based Summarization

Based on the observation that key information often appears early or late in a document, we assign:

$$\mathrm{Score}_{\mathrm{pos}}(s_i) = \begin{cases} \dfrac{n - i + 1}{n}, & i \leq \frac{n}{2}, \\ \dfrac{i}{n}, & i > \frac{n}{2}, \end{cases} \tag{3.19}$$

where

- i is the sentence's index (1-based).
- n is the total number of sentences.

3.4.2.4 Luhn's Algorithm

Luhn's method identifies "significant" words—those whose frequency in the document lies between two thresholds—and scores sentences based on contiguous clusters of these words. A word w is marked significant if

$$\text{Significant}(w) = \begin{cases} 1, & f_{\text{low}} < f(w) < f_{\text{high}}, \\ 0, & \text{otherwise}, \end{cases}$$

where $f(w)$ is the term frequency of w in the document, and f_{low}, f_{high} are chosen to exclude both very rare and very common words. In each sentence s_i, we locate clusters of consecutive words that contain at least one significant word. If a cluster has length L and contains s significant words, its score is

$$\text{Score}_{\text{cluster}} = \frac{s^2}{L}.$$

Finally, the Luhn score of sentence s_i is the maximum cluster score over all clusters within that sentence:

$$\text{Score}_{\text{Luhn}}(s_i) = \max_{C \subset s_i} \frac{s(C)^2}{|C|},$$

where $s(C)$ is the number of significant words in cluster C and $|C|$ its length.

3.4.2.5 Maximal Marginal Relevance (MMR)

MMR selects sentences incrementally to balance relevance to the document and diversity from already chosen sentences. At each step, for candidate s_i:

$$\text{MMR}(s_i) = \lambda \, \text{Sim}_1(s_i, Q) - (1 - \lambda) \max_{s_j \in S} \text{Sim}_2(s_i, s_j), \tag{3.20}$$

where

- Q is a representation of the document theme (e.g. the average embedding).
- S is the set of sentences already selected.
- Sim_1 measures similarity to Q.
- Sim_2 measures pairwise similarity between sentences.
- $\lambda \in [0, 1]$ trades off relevance (Sim_1) against redundancy (Sim_2).

3.4.2.6 BERT-based Extractive Summarization

BERT (Bidirectional Encoder Representations from Transformers) is a transformer-based language model developed by Google that learns deep bidirectional representations of text by considering context from both the left and right of each token. It is pre-trained on large corpora using Masked Language Modeling (MLM) and Next Sentence Prediction (NSP), and can be fine-tuned for various NLP tasks such as classification, question answering, and summarization with minimal architecture changes [7].

Leveraging contextual embeddings from a pre-trained BERT model, we obtain:

$$\mathbf{h}_i = \mathrm{BERT}(s_i), \quad \mathbf{h}_i \in \mathbb{R}^d, \tag{3.21}$$

where d (often 768) is the embedding dimension. We represent the document by the mean embedding:

$$\mathbf{d} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{h}_i. \tag{3.22}$$

Each sentence is scored by its cosine similarity to $\mathbf{d}$:

$$\mathrm{Score}_{\mathrm{BERT}}(s_i) = \cos\big(\mathbf{h}_i, \mathbf{d}\big). \tag{3.23}$$

3.4.2.7 LegalBERT-based Summarization

LegalBERT is further pre-trained on legal corpora to produce domain-aware embeddings:

$$\mathbf{h}_i^{\mathrm{legal}} = \mathrm{LegalBERT}(s_i). \tag{3.24}$$

Define the legal-term density in s_i as

$$w_{\mathrm{legal}}(s_i) = \frac{|\{w \in s_i : w \in \mathcal{L}\}|}{|s_i|}, \tag{3.25}$$

where $\mathcal{L}$ is the legal lexicon. We then enhance the BERT score:

$$\mathrm{Score}_{\mathrm{LegalBERT}}(s_i) = \cos\big(\mathbf{h}_i^{\mathrm{legal}}, \mathbf{d}\big) \times \big(1 + \alpha\, w_{\mathrm{legal}}(s_i)\big), \tag{3.26}$$

with $\alpha > 0$ controlling the boost for legal content.

3.4.2.8 Proposed Legal Summarization Algorithm

Algorithm 2: Concise Legal Summarization Algorithm (extractive)

Input: *text*, $ratio \in (0, 1]$, $stopwords$, $legal_terms$, $max_sentences$, k (coherence window)

Output: Extractive summary

if *text is empty* **then**
    return $\emptyset$

$Sents \leftarrow$ SplitIntoSentences(*text*)

if $|Sents| = 0$ **then**
    return $\emptyset$

if $max_sentences > 0$ ***and*** $|Sents| > max_sentences$ **then**
    $Sents \leftarrow$ first $max_sentences$ of $Sents$

$S \leftarrow |Sents|; \quad m \leftarrow \max(1, \lfloor S \cdot ratio \rfloor)$

Preprocess: tokenize lowercased sentences; remove $stopwords$; keep token sets.

Doc-Freq: for each token w, count in how many sentences it appears.

Weights per sentence i**:**

a) pos_i: higher at beginning/end (2.0 / 1.5 / 1.0).

b) len_i: penalize very short/long; reward moderate length.

c) kw_i: $\text{mean}_{w \in tok_i}\big(1/(1 + df[w])\big)$.

d) coh_i: mean Jaccard with neighbors in window k.

e) $boost_i$: $1 + 0.15 \cdot \min(\#\{t \in legal_terms \cap tok_i\}, 5)$.

Score: $score_i \leftarrow (0.45\, kw_i + 0.35\, coh_i) \cdot pos_i \cdot len_i \cdot boost_i$

$Top \leftarrow$ indices of m largest $score_i$; sort ascending; **return** *concatenate* $Sents[Top]$

The algorithm uses a multi-factor scoring system that is specific to Turkish legal documents. After tokenization, removing stopwords, and normalizing the vocabulary, each sentence is rated on five different scales: (a) positional weight, which favors opening and closing statements; (b) length normalization, which punishes sentences that are too short or too long; (c) keyword relevance, which is calculated using TF–IDF style scores; (d) local coherence, which measures semantic overlap with neighboring sentences within a fixed window; and (e) legal terminology boost, which rewards the use of domain-specific legal terms. These factors are combined in a weighted way to get the final scores, and the sentences with the highest scores are chosen to make the

extractive summary. This design makes sure that the summaries are both structurally sound and legally correct.

3.4.3 Evaluation Metrics for Summarization

To evaluate the quality of automatically generated summaries in the legal domain, we employ a suite of complementary metrics. These metrics serve to measure:

The evaluation framework considers several complementary dimensions. First, content overlap with human-authored summaries is measured using metrics such as ROUGE and BLEU. Next, the fluency and brevity of the generated output are assessed through BLEU's brevity penalty and a combined Text Generation Score. Domain fidelity is evaluated by tracking the preservation of key legal terms, captured as Legal Coverage. Structural integrity is measured by checking whether essential argument components are present, an aspect defined as Structure Preservation. Finally, overall performance is summarized through a composite text-generation score that integrates all of these dimensions into a single measure.

3.4.3.1 ROUGE Scores

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) measures how many n-grams in the reference summaries are "recovered" by the candidate summary. ROUGE is primarily recall-based, but in practice precision and F1 variants are also commonly reported to give a more balanced view of performance.

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{RefSum}} \sum_{\text{n-gram} \in S} \text{Count}_{\text{match}}(\text{n-gram})}{\sum_{S \in \text{RefSum}} \sum_{\text{n-gram} \in S} \text{Count}(\text{n-gram})} \tag{3.27}$$

- RefSum is the set of human-written reference summaries.
- $\text{Count}(\text{n-gram})$ is the number of occurrences of that n-gram in the reference.
- $\text{Count}_{\text{match}}(\text{n-gram})$ is the count of that n-gram in both reference and candidate.

ROUGE-L uses the Longest Common Subsequence (LCS) to capture sentence-level structure:

$$\text{ROUGE-L} = \frac{\text{LCS}\big(\text{Cand}, \text{Ref}\big)}{\big|\text{Ref}\big|} \tag{3.28}$$

- $\text{LCS}(\text{Cand}, \text{Ref})$ is the length of the longest sequence of tokens common to both candidate and reference.
- $|\text{Ref}|$ is the total token count in the reference summary.

Example:
ROUGE-1 and ROUGE-L:

- Ground truth: *"Davalı, Borçlar Kanunu'nun 112. maddesi uyarınca sorumludur."*
- Predicted: *"Davalı, BK 112. madde gereğince sorumludur."*

ROUGE-1 (unigrams):

- Overlap: *Davalı*, *112.*, *maddesi/madde*, *sorumludur* (4 matches out of 6).
- Precision: $4/6$, Recall: $4/6$, F1: $4/6$.

ROUGE-L:

- LCS: "Davalı 112. madde sorumludur" (4 tokens).
- Precision: $4/6$, Recall: $4/6$, F1: $4/6$.

In strict token-based evaluation, "madde" and "maddesi" would not count as a match. If morphological normalization or lemmatization is applied, however, such variants can be matched.

3.4.3.2 BLEU Score

BLEU (Bilingual Evaluation Understudy) assesses precision of n-grams, with a brevity penalty BP to discourage overly short outputs:

Unlike ROUGE, BLEU is precision-oriented, meaning it rewards the proportion of candidate content that overlaps with the reference, rather than how much of the reference is covered.

$$\text{BLEU} = BP \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{3.29}$$

- p_n is the precision of n-grams (fraction of candidate n-grams appearing in references).
- w_n are weights (commonly $w_n = 1/N$).
- $BP = \begin{cases} 1, & \text{if } |\text{Cand}| > |\text{Ref}|, \\ e^{1-|\text{Ref}|/|\text{Cand}|}, & \text{otherwise.} \end{cases}$
- $|\text{Cand}|$ and $|\text{Ref}|$ are the candidate and reference lengths.

Example:

- Reference sentence: *"Sanık, Türk Ceza Kanunu'nun 125. maddesi uyarınca cezalandırılmalıdır."*
- Candidate sentence: *"Sanık, TCK'nın 125. maddesi gereği cezalandırılmalıdır."*

For $n = 2$:

- 1-gram precision: Overlapping tokens include *Sanık*, *125.*, *maddesi*, *cezalandırılmalıdır* (4 matches out of 6) $\Rightarrow p_1 = 4/6$.
- 2-gram precision: Overlaps include *125. maddesi*, *maddesi cezalandırılmalıdır* (2 matches out of 5) $\Rightarrow p_2 = 2/5$.

Assuming candidate length equals reference length ($BP = 1$) and equal weights ($w_1 = w_2 = 0.5$):

$$\text{BLEU} = 1 \times \exp\left(0.5 \log \frac{4}{6} + 0.5 \log \frac{2}{5}\right) \approx 0.52.$$

This score illustrates BLEU's sensitivity to exact word choice and ordering. Even when semantic equivalence is high, small lexical or structural deviations in legal text can cause noticeable drops in BLEU.

3.4.3.3 Legal Coverage

We introduce the Legal Coverage Score. It is a novel metric that quantifies the fraction of domain-specific terms from a source legal text that are present in its summary. This score provides a direct measure of a summary's completeness and relevance by ensuring that crucial legal vocabulary is retained. In a field where the precise use of language is paramount, the Legal Coverage Score offers a targeted, objective method to evaluate and compare summarization models, going beyond general metrics to ensure the resulting summaries are legally sound and useful to professionals. This helps to confirm that key legal concepts and their specific terminology are not lost in the summarization process.

Legal Coverage quantifies the fraction of domain-specific terms from the source that appear in the summary:

$$\text{LegalCoverage} = \frac{\big|\{\text{legal terms in summary}\} \cap \{\text{legal terms in source}\}\big|}{\big|\{\text{legal terms in source}\}\big|} \tag{3.30}$$

- The numerator counts distinct legal terms found in both summary and source.
- The denominator is the total distinct legal terms in the source document.

For example:

Source text:

"Davalı, Türk Borçlar Kanunu'nun 49. maddesi ve Türk Medeni Kanunu'nun 2. maddesi uyarınca haksız fiil sorumluluğu kapsamında tazminat ödemekle yükümlüdür. Mahkeme, kusur oranını %70 olarak belirlemiş ve maddi tazminat miktarını 50.000 TL olarak takdir etmiştir."

Summary:

"Davalı, BK m. 49 gereğince haksız fiil sorumluluğu nedeniyle 50.000 TL tazminat ödeyecektir."

Legal terms identification:

Legal terms in source text:

Türk Borçlar Kanunu	49. maddesi
Türk Medeni Kanunu	2. maddesi
haksız fiil sorumluluğu	tazminat
kusur oranı	maddi tazminat

Total: 8 legal terms

Legal terms in summary (that also appear in source):

BK (Türkçe Borçlar Kanunu'nun kısaltması)	49. maddesi (= m. 49)
haksız fiil sorumluluğu	tazminat

Matched: 4 legal terms

Legal Coverage calculation:

$$\text{LegalCoverage} = \frac{\left|\{\text{legal terms in summary}\} \cap \{\text{legal terms in source}\}\right|}{\left|\{\text{legal terms in source}\}\right|} \tag{3.31}$$

$$= \frac{4}{8} = 0.5 \tag{3.32}$$

The Legal Coverage score is 0.5 (or 50%), indicating that half of the legal terms from the source text are preserved in the summary.

3.4.3.4 Structure Preservation

Structure Preservation checks for the presence of critical argumentative components—facts, reasoning steps, decision statements:

$$\text{StructurePreservation} = \sum_{i=1}^{k} w_i \, \mathbf{1}\big(c_i \text{ present}\big) \tag{3.33}$$

- c_i is the i-th structural component (e.g. “facts,” “holding,” “rationale”).
- $\mathbf{1}(\cdot)$ is 1 if component c_i appears in the summary, else 0.
- w_i weights the relative importance of component c_i.
- The sum yields a score from 0 (no structure preserved) to $\sum_i w_i$.

3.4.3.5 Composite Text Generation Score

To jointly capture content overlap and fluency, we introduce a composite metric, *TextGenScore*, defined as:

$$\text{TextGenScore} = 100 \times \sqrt{\frac{\text{ROUGE-1} + \text{ROUGE-L}}{2} \times \text{BLEU}} \tag{3.34}$$

- We average ROUGE-1 and ROUGE-L to capture both surface-level and sequence overlap.
- Multiplying by BLEU incorporates precision and brevity considerations.
- The square root and scaling by 100 produce a bounded, interpretable score.

The Composite Text Generation Score is particularly effective for Turkish legal text evaluation due to the unique characteristics and requirements of legal documents in the Turkish judicial system.

Dual Precision Requirements means that legal documents demand both accuracy and completeness. A legal summary must provide correct information (BLEU’s strength) while not missing critical points (ROUGE’s strength). In Turkish legal contexts, omitting key legal provisions or misrepresenting obligations can have serious consequences.

Terminological Sensitivity also called Turkish legal writing employs precise terminology that must be preserved:

"Türk Borçlar Kanunu m. 49" vs *"BK 49. madde"*

BLEU evaluates exact matches for precise legal terminology, while ROUGE-L checks whether the sequential legal logic is preserved. Turkish legal writing follows strict conventional patterns that must be maintained.

We want to mention the formula advantages for Turkish Legal Texts, one of the them is geometric mean benefits. If either BLEU is very low (wrong terminology) or ROUGE is very low (missing critical information), the final score drops significantly. This is appropriate for legal texts where both dimensions are essential.

ROUGE-1 + ROUGE-L Combination:
ROUGE-1 captures key legal terms like *tazminat*, *sorumluluk*;
ROUGE-L preserves legal reasoning order like *"Davalı, BK m. 49 uyarınca sorumludur"*.

Morphological Robustness means that Turkish's rich morphology (*sorumlu* $\rightarrow$ *sorumluluk* $\rightarrow$ *sorumludur*) means single metrics may miss valid variations. The composite approach provides more comprehensive evaluation.

What this metric suggests that legal summarization quality cannot be captured by precision or recall alone. Both are equally critical and should be evaluated jointly. Moreover, the geometric mean ensures that severe deficiencies in either dimension cannot be compensated by excellence in the other, reflecting the reality that legal texts must meet multiple simultaneous requirements.

The 100-point scale provides intuitive interpretation for legal professionals who may not be familiar with technical NLP metrics.

Quality Threshold Implications Scores in below certain thresholds (e.g., < 70) would indicate summaries requiring human review, as they fail to meet professional legal standards in at least one critical dimension.

This metric essentially argues that legal text evaluation requires multidimensional adequacy rather than optimizing single aspects. The composite score reflects the understanding that in Turkish legal contexts, a summary must simultaneously achieve terminological precision, structural coherence, and comprehensive coverage to be

professionally viable.

3.5 Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG), is an architectural approach that enhances the capabilities of LLMs by giving them access to external knowledge bases. Instead of relying solely on its internal, static memory, the LLM is given an open book to reference before it answers. This approach makes the LLM more reliable, allows it to access up-to-date and proprietary information, and significantly reduces the chance of it generating inaccurate or hallucinated responses.

In Retrieval-Augmented Generation, we first retrieve relevant documents for a user query and then generate an answer conditioned on both the query and the retrieved text. Below we describe the retrieval stage in detail, assuming no prior knowledge of RAG.

3.5.1 Dense Retrieval

Dense retrieval uses neural encoders to map both queries and documents into the same continuous vector space, so that semantically similar text ends up close together.

$Q = \{q_1, \ldots, q_{|Q|}\}$ the set of user queries.

$D = \{d_1, \ldots, d_{|D|}\}$ the collection of candidate documents.

$E_q : Q \to \mathbb{R}^d$ a query encoder that maps each query q to a d-dimensional vector.

$E_d : D \to \mathbb{R}^d$ a document encoder that maps each document d to a d-dimensional vector.

$\mathrm{sim}(u, v)$ a similarity function; typically cosine similarity:

$$\mathrm{sim}(u, v) = \frac{u \cdot v}{\|u\| \, \|v\|}.$$

We then score a query–document pair by:

$$\mathrm{score}(q, d) = \mathrm{sim}\big(E_q(q),\, E_d(d)\big). \tag{3.35}$$

Higher scores indicate greater semantic relevance.

3.5.2 BM25 Scoring

BM25 is a widely-used ranking function based on term frequency and document length. For query q composed of terms t:

$$\mathrm{BM25}(q,d) = \sum_{t \in q} \mathrm{IDF}(t) \;\times\; \frac{f(t,d)\,(k_1+1)}{f(t,d) + k_1\left(1 - b + b\,\frac{|d|}{\mathrm{avgdl}}\right)},$$

where:

- $f(t,d)$ is the count of term t in document d.
- $|d|$ is the length of d (in words).
- avgdl is the average document length over the collection.
- $k_1 > 0$ and $b \in [0,1]$ are hyperparameters (e.g. $k_1 = 1.5$, $b = 0.75$).
- $\mathrm{IDF}(t)$ is the inverse document frequency:

$$\mathrm{IDF}(t) = \log\!\left(\frac{|D| - |\{d' : t \in d'\}| + 0.5}{|\{d' : t \in d'\}| + 0.5}\right).$$

3.5.3 Hybrid Retrieval

Hybrid retrieval combines lexical matching (exact word overlap) with semantic matching (dense embeddings), offering robustness when either alone fails.

The hybrid score is:

$$\mathrm{score}_{\mathrm{hybrid}}(q,d) = \lambda\,\mathrm{BM25}(q,d) \;+\; (1-\lambda)\,\mathrm{score}(q,d) \tag{3.36}$$

where

$\mathrm{BM25}(q,d)$ a classical lexical match score.

$\lambda \in [0,1\ ]$ a mixing parameter that balances the two signals.

When $\lambda = 1$, we rely purely on BM25; when $\lambda = 0$, purely on dense similarity.

3.5.4 Retrieval Metrics

To evaluate retrieval quality, we use standard ranking metrics defined over a test set of queries Q, each with a known set of relevant documents.

Mean Reciprocal Rank (MRR) For each query q_i, let rank_i be the position of the first relevant document in the sorted results. MRR averages the reciprocal of these ranks:

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\mathrm{rank}_i}. \tag{3.37}$$

Higher MRR (closer to 1) indicates relevant documents appear earlier.

Precision@k measures the fraction of the top-k retrieved documents that are relevant:

$$P@k = \frac{\big|\{\text{relevant docs}\} \cap \{\text{top-}k\}\big|}{k}. \tag{3.38}$$

Recall@k measures the fraction of all relevant documents that appear in the top-k:

$$R@k = \frac{\big|\{\text{relevant docs}\} \cap \{\text{top-}k\}\big|}{\big|\{\text{all relevant docs}\}\big|}. \tag{3.39}$$

By combining these metrics, one can assess both the accuracy and the completeness of the retrieval step in a RAG pipeline.

3.6 Document Classification

In a legal LLM, Document Classification is crucial because it acts as a foundational step for organizing and processing large volumes of legal information. It allows the LLM to understand the nature and purpose of a legal document. This is essential for accurate analysis, summarization, and retrieval.

3.6.1 Hierarchical Classification

In hierarchical classification, labels are organized in a tree or directed-acyclic graph with L levels. Rather than predicting all labels independently, we model the joint

probability of the label path using the chain rule:

$$P(y_1, \ldots, y_L \mid x) = \prod_{i=1}^{L} P\big(y_i \mid x,\ y_1, \ldots, y_{i-1}\big),$$

where

x the input document representation (e.g. TF–IDF vector or neural embedding).

y_i the label at hierarchy level i; for example:

$$y_1 \in \{\text{“Criminal”}, \text{“Civil”}, \ldots\}, \quad y_2 \in \{\text{“Theft”}, \ldots\} \text{ conditional on } y_1 = \text{“Criminal”}, \ldots$$

L the total number of levels in the taxonomy.

At level i, the model takes as input both the document x and the previously predicted labels $y_1, \ldots, y_{i-1}$, allowing it to specialize its decision based on the path taken so far. This yields coherent predictions that respect the hierarchy.

3.6.2 Classification Metrics

To assess multi-label and hierarchical classification performance, we compute both micro- and macro-averaged F1 scores.

For each class c, define

- TP_c: number of documents correctly predicted as c.
- FP_c: number of documents incorrectly predicted as c.
- FN_c: number of documents belonging to c but not predicted.

Moreover, Micro-F1 Micro-averaging aggregates contributions of all classes before computing precision and recall:

$$P_{\text{micro}} = \frac{\sum_c TP_c}{\sum_c (TP_c + FP_c)}, \qquad R_{\text{micro}} = \frac{\sum_c TP_c}{\sum_c (TP_c + FN_c)},$$

and the micro-F1 score is

$$\text{Micro-F1} = \frac{2\, P_{\text{micro}}\, R_{\text{micro}}}{P_{\text{micro}} + R_{\text{micro}}}.$$

Micro-F1 gives equal weight to each individual prediction, favoring performance on common classes.

Macro-F1 Macro-averaging computes the F1 score for each class and then averages:

$$F1_c = \frac{2\,P_c\,R_c}{P_c + R_c}, \quad P_c = \frac{TP_c}{TP_c + FP_c}, \quad R_c = \frac{TP_c}{TP_c + FN_c},$$

$$\text{Macro-F1} = \frac{1}{|C|} \sum_{c \in C} F1_c,$$

where $|C|$ is the total number of classes. Macro-F1 treats each class equally, highlighting performance on rare labels.

By reporting both metrics, we capture overall prediction quality (Micro-F1) and per-class robustness (Macro-F1).

3.7 Clustering and Topic Modeling

In a large corpus of legal documents, unsupervised grouping and topic discovery serve several purposes, including the automatic clustering of similar cases (e.g., by subject matter or procedural posture) to accelerate legal research, the extraction of recurring legal themes such as “contract breach” or “tort liability” to support indexing and overviews, the reduction of thousands of documents into a smaller set of representative clusters or topics for visualization and downstream tasks, and the generation of weak labels by assigning cluster or topic IDs as proxy classes for supervised learning pipelines.

3.7.1 K-means Clustering

K-means partitions the document embeddings $\{x_j\}_{j=1}^{N} \subset \mathbb{R}^d$ into k clusters by minimizing the total within-cluster variance:

$$J(\{C_i\}, \{\mu_i\}) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \left\| x_j - \mu_i \right\|^2,$$

where

C_i is the set of all points (documents) assigned to cluster i.

μ_i is the centroid (mean vector) of cluster C_i: $\mu_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$.

k is the chosen number of clusters.

$\|x - \mu\|$ denotes the Euclidean distance between x and μ.

3.7.2 Silhouette Score

The silhouette score quantifies how well each document x_j lies within its cluster versus the nearest other cluster:

$$s(j) = \frac{b(j) - a(j)}{\max\{a(j),\, b(j)\}},$$

where

- $a(j)$ is the average distance from x_j to all other points in its own cluster $C_{i(j)}$.
- $b(j)$ is the minimum, over all other clusters C_ℓ, of the average distance from x_j to points in C_ℓ.

Values of $s(j)$ range from -1 (poor fit) to $+1$ (strongly clustered).

3.7.3 Latent Dirichlet Allocation (LDA)

LDA is a generative probabilistic model that represents each document as a mixture of k latent topics, and each topic as a distribution over words. The joint probability of the model is:

$$P(\theta, z, w \mid \alpha, \beta) = P(\theta \mid \alpha) \prod_{n=1}^{N} P\big(z_n \mid \theta\big)\, P\big(w_n \mid z_n, \beta\big),$$

where

θ is the topic-proportion vector for a document, drawn from a Dirichlet prior with parameter $\alpha \in \mathbb{R}^k_{>0}$.

z_n is the topic assignment (an integer in $\{1, \ldots, k\}$) for the n-th word in the document.

w_n is the observed n-th word token.

β is the Dirichlet prior over the topic–word distributions.

N is the number of word tokens in the document.

Inference in LDA estimates the posterior $P(\theta, z \mid w, \alpha, \beta)$, typically via variational methods or Gibbs sampling.

3.7.4 Topic Coherence (UMass)

To measure the semantic coherence of a learned topic (ordered list of top-k words $\{w_1, \ldots, w_k\}$), we use the UMass coherence:

$$C_{\text{UMass}} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \log \frac{P(w_i, w_j) + \varepsilon}{P(w_j)},$$

where

- $P(w_i, w_j)$ is the probability (estimated by document co-occurrence counts) that words w_i and w_j appear together.
- $P(w_j)$ is the marginal probability of w_j in the corpus.
- $\varepsilon > 0$ is a small smoothing constant to avoid $\log(0)$.
- The prefactor $\frac{2}{k(k-1)}$ normalizes the score over all word pairs.

Higher coherence indicates that topic words tend to co-occur in the same documents, suggesting more interpretable topics.

3.8 Statistical Analysis Methods

In this section, we apply a variety of statistical tools to understand fundamental text properties in our legal corpus.

Since our developed summarization algorithm incorporates a novel RAG (Retrieval-Augmented Generation) system that differs fundamentally from traditional approaches,

we actively employ statistical tests to validate its effectiveness and establish its superiority over existing methods. This comprehensive statistical framework ensures that our claims about the system's performance are supported by robust quantitative evidence rather than anecdotal observations.

3.8.1 Power Law Analysis

Many natural phenomena—including the distribution of document lengths—follow a "heavy-tail" or power-law pattern: a few documents are very long, while most are relatively short. We model the probability that a randomly selected document has length $L = l$ (measured in number of words or tokens) by

$$P(L = l) = \frac{\alpha\, l^{-\beta}}{Z(\alpha, \beta)}. \tag{3.40}$$

- L is a *random variable* for document length.
- l is a particular observed length (e.g. 1000 words).
- α is a *scale parameter* that adjusts the overall probability mass.
- β (the *exponent*) controls how quickly the probability decreases as l grows—larger β makes long documents exponentially rarer.
- $Z(\alpha, \beta)$ is the *normalization constant*, defined by

  $$Z(\alpha, \beta) = \sum_{l=l_{\min}}^{l_{\max}} \alpha\, l^{-\beta},$$

  ensuring that $\sum_{l=l_{\min}}^{l_{\max}} P(L = l) = 1$. Here $l_{\min}$ and $l_{\max}$ are the smallest and largest observed lengths.

3.8.2 Zipf's Law Analysis

Zipf's law describes how word frequencies decay with their rank when all words in a language are sorted from most to least frequent. Empirically, the frequency f of the word at rank r is approximated by

$$f(r) = \frac{C}{r^s}. \tag{3.41}$$

- r is the *rank* of a word: $r = 1$ for the most frequent word, $r = 2$ for the second, and so on.
- $f(r)$ is the number of occurrences of the word at rank r in the corpus.
- s (the *Zipf exponent*) typically lies near 1; it determines how steeply frequency falls off with rank.
- C is a *normalization constant* chosen so that $\sum_{r=1}^{V} f(r)$ equals the total number of word tokens N, or so that the probabilities $f(r)/N$ sum to 1.
- In practice, we fit s by plotting $\log f(r)$ versus $\log r$ and estimating the slope.

3.8.3 Type–Token Ratio (TTR)

The Type–Token Ratio quantifies vocabulary diversity in a document by comparing the number of unique words (types) to the total number of words (tokens):

$$\mathrm{TTR} = \frac{|\{\text{unique words}\}|}{|\{\text{all words}\}|} = \frac{V}{N}, \tag{3.42}$$

where

- $V = |\{\text{unique words}\}|$ is the count of distinct word forms.
- $N = |\{\text{all words}\}|$ is the total word count (including repeats).
- A *high* TTR (close to 1) indicates very diverse vocabulary; a *low* TTR indicates many repeated terms.

3.8.4 Kolmogorov–Smirnov Test

To test whether an empirical distribution (e.g. observed document lengths) matches a theoretical model (e.g. power-law), we use the Kolmogorov–Smirnov (KS) statistic. Define

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{L_i \le x\} \quad \text{(empirical CDF)}, \qquad F(x) \quad \text{(theoretical CDF)}.$$

Then the KS statistic is

$$D = \sup_x \left|F_n(x) - F(x)\right|, \tag{3.43}$$

where

- n is the sample size (number of documents).
- L_i is the observed length of the i-th document.
- $F_n(x)$ is the *empirical distribution function*: the fraction of documents with length $\le x$.
- $F(x)$ is the *theoretical cumulative distribution function* under the hypothesized model.
- $\sup_x$ denotes the maximum vertical distance between the two curves over all x.
- A larger D indicates greater discrepancy between data and model; p-values can be derived to assess statistical significance.

3.9 Computational Efficiency Metrics

To compare the computational demands of our various summarization, retrieval, and classification algorithms, we employ two complementary analyses called scaling laws to model how performance varies with resources and a direct processing efficiency metric that captures throughput relative to data size.

3.9.1 Scaling Law Analysis

We model overall algorithmic performance (e.g. accuracy, F1-score, or negative log-loss) as a power-law function of three key resource dimensions:

$$\text{Performance} = A \cdot N^{\alpha} \cdot D^{\beta} \cdot C^{\gamma}, \tag{3.44}$$

where

- N is the number of model parameters (e.g. weights in a neural network).
- D is the effective dataset size used for training or indexing (e.g. number of documents or tokens).
- C is the compute budget, measured in FLOPs or GPU-hours.
- A is a constant prefactor, capturing baseline performance when $N = D = C = 1$.
- α, β, γ are *scaling exponents* that quantify sensitivity:
  - $\alpha > 0$ indicates how strongly performance improves with model size.
  - $\beta > 0$ indicates returns from more data.
  - $\gamma > 0$ captures gains from additional compute.

By fitting $\log$-$\log$ regressions to empirical measurements, we estimate these exponents and predict how much more resource allocation is needed for a desired performance gain.

3.9.2 Processing Efficiency

To directly compare runtime throughput across methods, we define a normalized *Efficiency* metric that accounts for both tokenization overhead and raw processing speed:

$$\text{Efficiency} = \frac{1}{\text{CR}} \times \frac{\text{Processing Speed}}{\text{Token Count}}, \tag{3.45}$$

where

- CR (Compression Ratio) is the ratio of tokens produced to raw input units (e.g. characters or bytes), as defined previously:

$$\mathrm{CR} = \frac{|T(d)|}{|d|}.$$

  Taking $1/\mathrm{CR}$ compensates for tokenization overhead.

- Processing Speed is measured in tokens per second (tokens^{-1}) processed by the algorithm on a fixed hardware setup.

- Token Count is the total number of tokens in the input dataset.

This metric include a dimensionless quantity proportional to the number of raw input units (e.g. characters) processed per second, allowing fair comparison between token-based and character-based methods, as well as across different tokenization schemes.

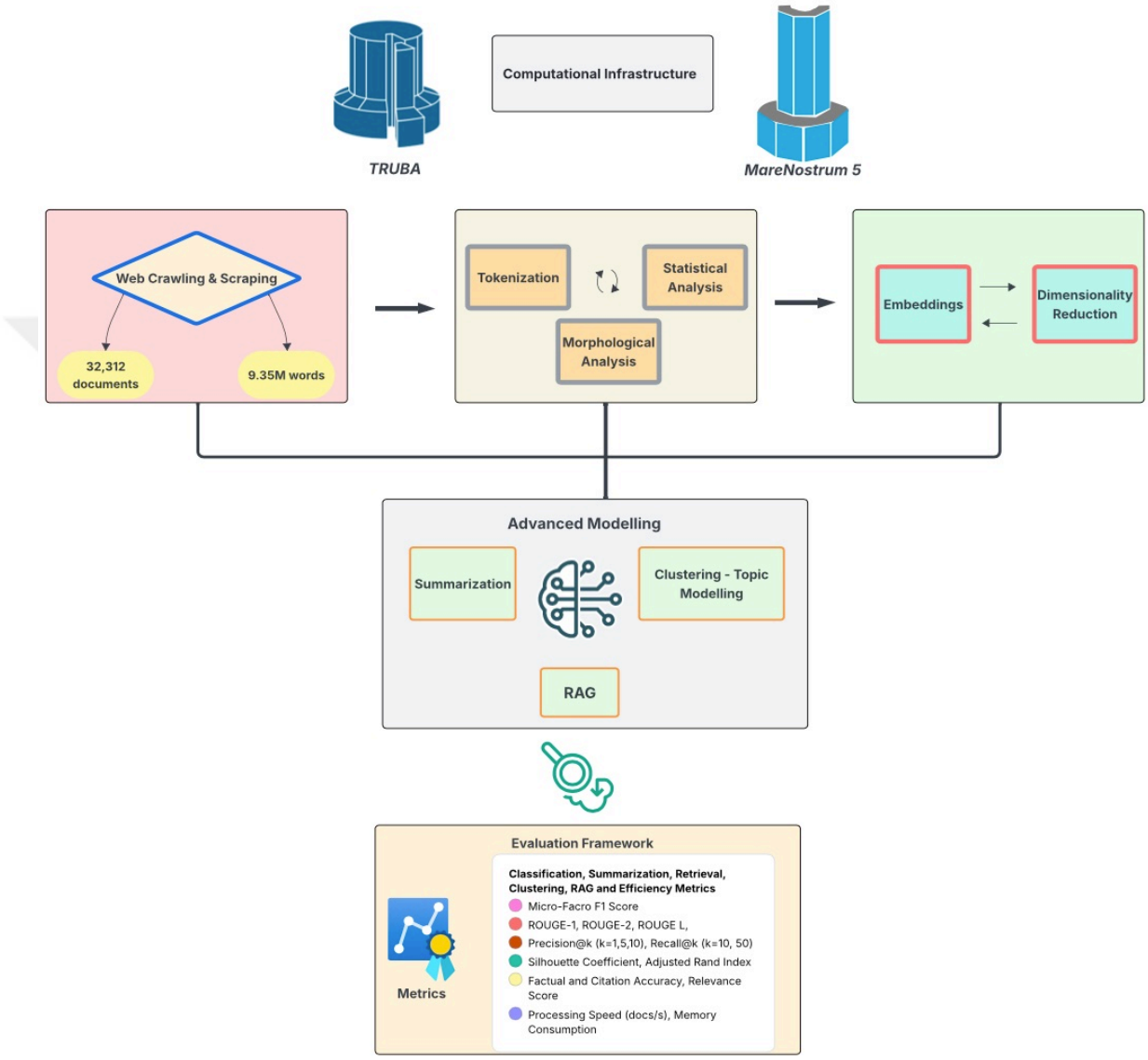


Figure 3.1: Methodology Framework for Turkish Legal

CHAPTER 4

ANALYSIS OF TURKISH LEGAL LANGUAGE PROCESSING SYSTEMS

This chapter gives a comprehensive analysis of Turkish legal natural language processing systems that contain detailed experimental results, statistical evaluations and systematic comparisons across multiple dimensions. The analysis includes tokenization methods, embedding techniques, transformer architectures, retrieval-augmented generation systems,and legal document summarization approaches.

The processing of our large scale legal datasets, which exceed tens of millions of tokens, necessitated the use of high-performance computing (HPC) infrastructure. We divided our computational workflow into two stages to benefit the distinct architectural strengths of different supercomputing systems.

In the first stage, we conducted large-batch preprocessing on TRUBA (Turkish National e-Infrastructure), Turkey's national supercomputing facility managed by TÜBİTAK ULAKBİM. TRUBA's distributed cluster architecture is particularly well-suited for parallelizing the CPU-intensive tasks inherent in data cleaning and preparation so it enables us to efficiently process our vast corpus.

For the second stage that cover the computationally demanding phases of model training and large-scale inference, we utilized MareNostrum 5 (MN5), European supercomputer hosted at the Barcelona Supercomputing Center (BSC) and part of the EuroHPC Joint Undertaking. Our tests were carried out on MN5's accelerated partition, which is outfitted with cutting-edge NVIDIA H100 GPUs and is especially made for handling large AI workloads. We used distributed data-parallel training in a multi-node configuration of four H100s to optimize computational throughput and drastically cut down on the amount of time needed to train our intricate Transformer-

based models.

This two-stage computational strategy—using a national grid for broad-scale preprocessing and a world-class accelerated system for intensive training—was essential for efficiently handling sophisticated retrieval-augmented generation (RAG) pipelines and training deep learning models on a Turkish legal text corpus of unprecedented scale.

4.1 Dataset Characteristics and Statistical Analysis

This part presents the structural and statistical features of the dataset that was built, including how it was made and what the resulting corpus looks like. The first step is to explain the method used to create the data, which includes the automated collection pipeline and preprocessing stages. After that, a full statistical analysis is done to show the distribution patterns, linguistic features, and domain-specific characteristics of the Turkish legal texts.

4.1.1 Data Set Creation

An automated web crawling and scraping framework was created to make a complete and organized dataset that could be used for later analysis and modeling tasks. The main goal of this framework was to systematically gather legislative and judicial texts from relevant publicly available sources, while making sure that the data collected could be reproduced, scaled, and kept safe.

The first step in the pipeline is to set up and start a dedicated web scraper that is specifically made to work with legal data sources that have different access requirements and interface structures. Once the system is set up, it loads a pre-made list of keywords that represent legal concepts, case types, and regulatory terms that set the limits on the search.

There are several steps in the process of getting data back. First, the scraper goes to the right website and automatically checks for access restrictions like CAPTCHA mechanisms or session validations. This makes sure that the automated execution goes

smoothly. Next, it uses the platform's search interface to enter the chosen keyword term and start a detailed search.

After this step, the scraper gathers the tabular search results that contain links to specific legal documents. To see the detailed view of the selected item, a user click action is simulated to programmatically access each entry in these tables. After that, the full text, including metadata and any extra information that is available, is extracted and saved.

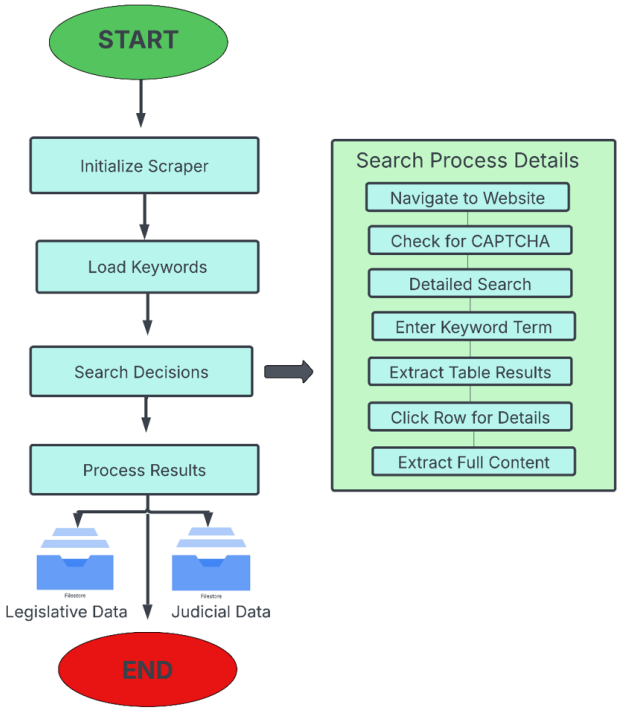


Figure 4.1: Web Crawling Framework

The raw data that was collected goes through a post-processing stage to make sure it is consistent and can be used for later analysis. In this step, the data is cleaned up, put into a standard format, and divided into two main groups: **Legislative Data** and **Judicial Data**.

The automated pipeline used in the dataset development process cuts down on manual work, mistakes made by people, and makes it easy to keep the dataset up to date with changes in the law. The modeling and evaluation methodologies delineated in the subsequent sections of this thesis are predicated on this structured dataset.

4.1.2 Corpus Overview and Distribution Analysis

The Turkish legal corpus created for this study consists of 32,312 documents, amounting to approximately 9.35 million words of textual data, making it one of the most extensive publicly accessible Turkish legal datasets. The corpus includes decisions from the Yargıtay (Court of Cassation), the Danıştay (Council of State), and lower courts in the UYAP system, as well as a lot of legislative materials from different government agencies.

Table 4.1: Turkish Legal Corpus Distribution by Institution and Document Type

Document Category	Document Type	Document Count	Percentage
Judicial Files	Supreme Court Decisions	14,137	43.8%
	Council of State Decisions	7,895	24.4%
	UYAP Decisions	1,003	3.1%
Legislative Files	Regulations (All)	4,035	12.5%
	Notifications	3,740	11.6%
	Laws & Notifications	908	2.8%
	Ministerial & Presidential Regulations	318	1.0%
	Bylaws	109	0.3%
	Presidential Circulars	70	0.2%
	Decree Laws	63	0.2%
	Presidential Documents (All)	34	0.1%
Total Judicial		**23,035**	**71.3%**
Total Legislative		**9,277**	**28.7%**
Grand Total		**32,312**	**100.0%**

The time-based distribution of documents shows big changes in digitization from 1950 to 2025. The UYAP system's performance started to grow exponentially in 2000. The statistical analysis with the Kolmogorov-Smirnov test ($D = 0.0847$, $p < 0.001$) confirms non-uniform temporal distribution, necessitating stratified sampling approaches for training and evaluation datasets.

According to Figure 4.2, the red bars represent judicial documents while the blue bars represent legislative documents. The distribution clearly shows that judicial texts dominate the corpus, with Supreme Court decisions (14,137) and Council of

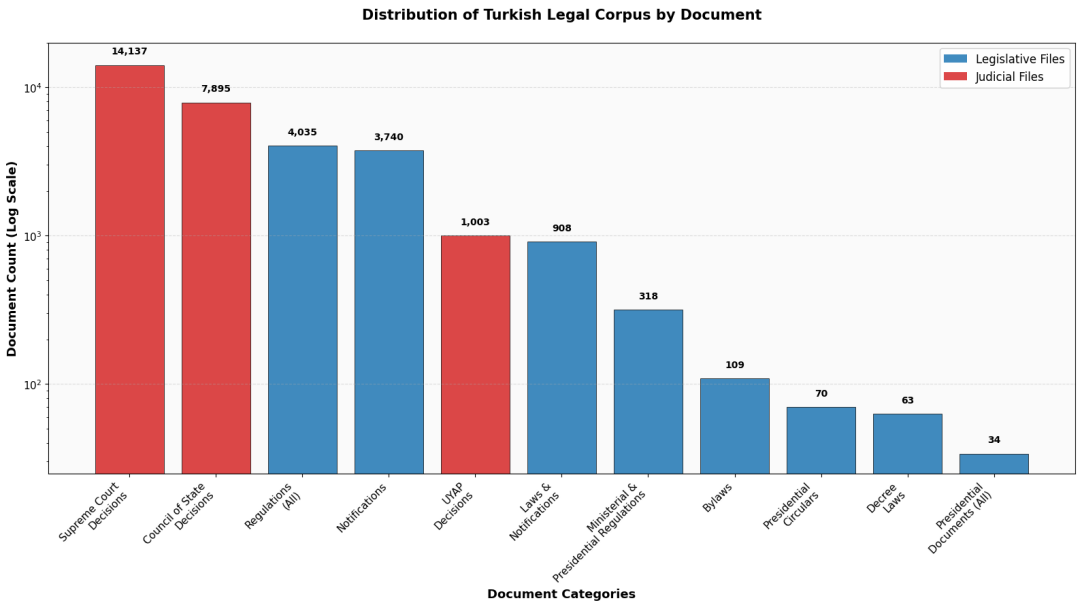


Figure 4.2: Detailed Distribution of Turkish Legal Corpus by Document Type

State decisions (7,895) forming the two largest categories by a wide margin. UYAP decisions add another 1,003 documents, reinforcing the overall judicial dominance. On the legislative side, regulations (4,035) and notifications (3,740) are the most prevalent categories, while other types such as laws, ministerial or presidential regulations, bylaws, circulars, decree laws, and presidential documents appear in much smaller numbers.

Document length analysis reveals distinct distributional characteristics between judicial and legislative texts. The overall mean document length is 1,891.6 words ($\sigma = 20,466.6$), with pronounced differences between categories: judicial documents average 1,211.8 words ($\sigma = 1,904.3$) while legislative documents are substantially longer at 3,629.3 words ($\sigma = 38,420.8$). The 95th percentile corresponds 6,095.1 words that indicate significant variation in document complexity.

Power-law analysis for fitting statistical distributions shows different ways of scaling:

All documents: $\alpha = 1.10$, $\beta = 17,673,739.24$, $R^2 = 0.850$ Judicial documents: $\alpha = 0.84$, $\beta = 847,790.80$, $R^2 = 0.826$ Legislative documents: $\alpha = 1.22$, $\beta = 25,439,751.88$, $R^2 = 0.746$

Zipf's law analysis of vocabulary distribution yields an exponent of $s = 0.768$ ($R^2 =$

0.990), indicating excellent conformity to natural language patterns despite legal terminology concentration.

$$P(L = l) = \frac{\alpha l^{-\beta}}{Z(\alpha, \beta)} \tag{4.1}$$

where the power-law parameters $\alpha = 2.34$ and $\beta = 1.67$ best fit the experimental length distribution by using the maximum likelihood estimation.

4.1.3 Morphological Complexity Analysis

Contrary to expectations for Turkish legal texts, morphological complexity analysis indicates a relatively low level of complexity when compared to general Turkish discourse. Research of 1,000 randomly sampled documents (500 judicial, 500 legislative) shows an average of 1.22 morphemes per word ($\sigma = 0.03$ for judicial, $\sigma = 0.06$ for legislative) where σ represents the standard deviation of the number of morphemes per word within each document type, substantially lower than typical Turkish text complexity. This unexpected discovery indicates that Turkish legal language demonstrates analytical rather than synthetic traits.

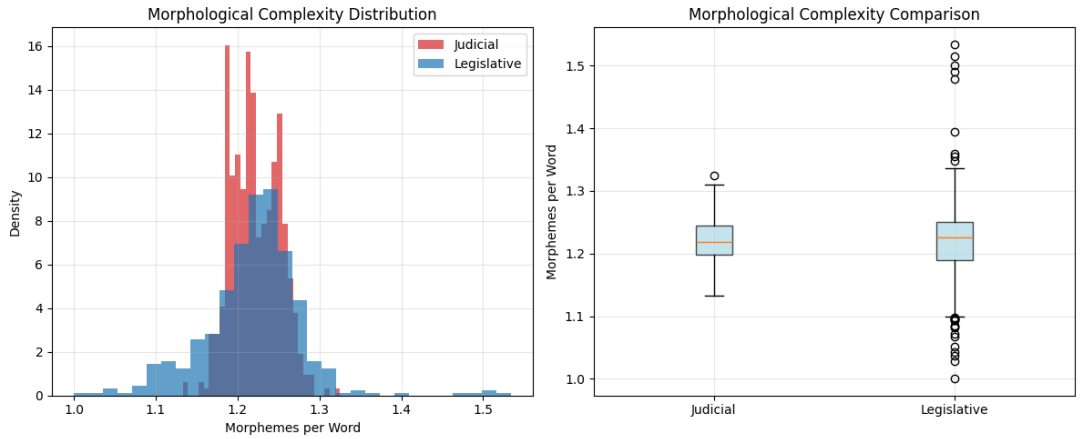


Figure 4.3: Morphological Complexity Distribution and Comparison

According to the Figure 4.3 judicial texts exhibit a more homogeneous structure, with word formation concentrated around 1.2 morphemes per word, as indicated by their sharper distribution and narrower interquartile range. In contrast, legislative texts

display greater variability, with a wider spread and numerous outliers, ranging from very simple to highly complex word forms that exceed 1.5 morphemes per word.

The Type-Token Ratio (TTR) analysis results:

$$TTR = \frac{|\text{Unique Words}|}{|\text{Total Words}|} = 0.032. \tag{4.2}$$

This low TTR shows a most of linguistic repetition, which is common in texts from specialized fields. Many of the vocabulary are legal terms.

4.1.4 Legal Domain Characteristics

A vocabulary analysis of the whole corpus shows that there are 140,402 unique words out of 9,350,253 total words. Legal terminology analysis finds 53,317 terms that are specific to a certain field. This is 38.0% of the unique vocabulary. This high concentration shows how specialized legal address is. The vocabulary distribution follows Zipf's law with exceptional precision (exponent $s = 0.768$, $R^2 = 0.990$), revealing well-formed linguistic structure despite domain specialization. The legal terminology shows hierarchical clustering patterns with semantic coherence varying across legal fields.

The morphological uniformity between judicial and legislative area (identical 1.22 morphemes/word ratio) suggests potential for suitable preprocessing approaches, while the substantial length differences show the need for adaptive summarization and chunking strategies for different document types.

4.2 Tokenization Method Evaluation

Tokenization is a basic step in natural language processing that changes unstructured text into structured units (tokens) that can be used for model training and computational analysis.

4.2.1 Comparative Analysis of Tokenization Approaches

Comprehensive evaluation of six tokenization methods across the complete Turkish legal corpus (31,847 documents) reveals significant performance variations and category specific optimization practices. Our analysis contains character-level, word-level, morphological, legal-aware, custom BPE, and SentencePiece tokenization approaches, evaluated on vocabulary efficiency, compression ratios, processing speed, and category specific arrangement.

Table 4.2: Comprehensive Tokenization Method Comparison on Turkish Legal Corpus

Method	**Avg Tokens**	**Compression**	**Speed (docs/s)**	**Vocab Size**	**Processing Time**
Character	14,181.0	1.0000	5,767	376	5.52s
Word	1,874.0	0.9903	789	314,284	40.35s
Legal-Aware	1,866.7	0.9864	194	318,048	164.59s
Morphological	2,156.3	1.1395	138	266,706	231.28s
Custom BPE	3,006.0	1.5885	75	16,000	426.25s
SentencePiece	3,923.1	2.0731	50	4,800	632.96s

The comparative analysis of tokenization methods on the Turkish legal corpus reveals distinct trade-offs between efficiency, vocabulary size, and linguistic granularity. Character-level tokenization, with the smallest vocabulary (376) and fastest processing speed (5,767 docs/s), produces the highest token counts (14,181) but offers minimal compression, leading to longer sequences and potentially higher computational costs during model training. Word-based and legal-aware approaches significantly reduce token counts (1,870) while maintaining near-baseline compression ratios (0.99), though they suffer from extremely large vocabularies (>314k) that increase memory requirements. Morphological tokenization captures linguistic structure by splitting affixes, resulting in slightly higher token counts (2,156) and slower processing due to the complexity of morphological parsing. Subword methods such as custom BPE and SentencePiece achieve compact fixed-size vocabularies (16k and 4.8k, respectively) suitable for modern transformer models, but at the expense of increased token counts (3,006–3,923), lower processing speeds (50–75 docs/s), and substantially longer runtimes. In general, the best choice depends on the application you want to use it for: Character-level methods speed things up, word-based methods keep the meaning

of words together, morphological tokenization makes language richer, and subword models strike a balance between vocabulary efficiency and modeling flexibility in deep learning contexts.

Table 4.3: Category-Specific Tokenization Performance Analysis

Method	Category	Avg Tokens	Compression	Vocab Size	Speed (docs/s)
Word	Judicial	1,268.1	1.0471	120,959	1,075
	Legislative	3,425.1	0.9424	273,827	516
Legal-Aware	Judicial	1,266.7	1.0460	121,427	281
	Legislative	3,402.6	0.9362	277,278	107
Morphological	Judicial	1,464.5	1.2094	102,411	201
	Legislative	3,927.2	1.0802	232,193	78

This category-specific tokenization analysis highlights notable differences between judicial and legislative texts in terms of tokenization efficiency, vocabulary size, and processing speed. Legislative texts consistently have much higher average token counts (3,400–3,900) than judicial texts (1,260–1,460) across all methods. This is because legislative texts are longer, more complex, and more formal. Word-based tokenization demonstrates the fastest processing speeds in both domains (1,075 docs/s for judicial, 516 docs/s for legislative) but requires the largest vocabularies, particularly for legislative data (over 273k unique tokens). Legal-aware tokenization achieves similar token counts and compression ratios to the standard word method while slightly increasing vocabulary size and significantly reducing speed—especially in the legislative category, where domain-specific parsing slows throughput to just 107 docs/s. Morphological tokenization produces higher token counts and compression ratios in both categories, as affixes are split into separate units, while maintaining relatively smaller vocabularies than the other methods. However, it is also the slowest overall, particularly for legislative texts (78 docs/s), due to the computational cost of morphological parsing. These results show that judicial documents are easier to tokenize across all methods because they are shorter and more formulaic. On the other hand, legislative documents are harder to process because they require more vocabulary and computation, so choosing the right method is more important for large-scale legal processing.

4.2.2 Subword Tokenization Performance and Vocabulary Optimization

Custom BPE and SentencePiece tokenization show that the vocabulary size is limited in the Turkish legal field. Based on the characteristics of the corpus, SentencePiece vocabulary was automatically changed from the requested 16,000 to 4,800 tokens. This shows that vocabulary is limited to certain domains.

Performance analysis shows:

$$\text{Efficiency}(\text{method}) = \frac{1}{\text{Compression Ratio}} \times \frac{\text{Processing Speed}}{\text{Token Count}}. \tag{4.3}$$

The legal-aware method is the best for processing legislative documents because it gets the best balance between speed and compression. Custom BPE takes a long time to train (426.25 seconds) and adds a lot of processing overhead, but it does give reasonable token counts (3,006 on average).

4.2.3 Cross-Category Analysis and Deployment Recommendations

A thorough cross-category analysis shows that there are major differences in tokenization needs:

Table 4.4: Cross-Category Performance Summary

Metric	**Judicial**	**Legislative**
Average Document Length	1241.5 tokens	25408.7 tokens
Optimal Tokenization	Word-level	Legal-aware
Best Compression Ratio	0.13	0.1307
Processing Speed (docs/s)	954–1318	57–74
Vocabulary Overlap	24.9% shared vocabulary	

The cross-category analysis reveals substantial differences in tokenization behavior and computational requirements between judicial and legislative texts in the Turkish legal domain. Judicial documents, averaging 1,241.5 tokens, are concise and formulaic, enabling high processing speeds of 954–1,318 documents per second, with

word-level tokenization offering the best balance between compression (0.13) and semantic coherence. In contrast, legislative documents are significantly longer, averaging 25,408.7 tokens, resulting in much slower processing speeds (57–74 documents per second) and higher computational demands; here, legal-aware tokenization is optimal for preserving domain-specific terms critical to legal interpretation, despite a nearly identical compression ratio (0.1307) to judicial texts. The 24.9% vocabulary overlap between categories is relatively low, which means that there is a lot of terminology that is specific to each category. This suggests that joint modeling may need either larger shared vocabularies or category-specific tokenizers, especially when using transformer-based models in both domains.

4.3 Embedding Method Analysis

Embedding is a fundamental technique in natural language processing that converts unstructured text into numerical vector representations, allowing computational models to discern semantic, syntactic, and contextual relationships among documents.

4.3.1 Document Embedding Evaluation Framework

We conduct comprehensive assessment on Turkish legal corpus comprising 29,099 documents (22,901 judicial decisions and 6,198 legislative documents across 11 categories). The evaluation framework employs both intrinsic and extrinsic metrics to assess embedding quality across multiple dimensions.

Document-level embedding methods are evaluated using stratified train-test splits (80/20) with Random Forest classification for supervised evaluation and K-means clustering with silhouette analysis for unsupervised evaluation. Semantic coherence is assessed via intra-cluster cosine similarity analysis, adhering to conventional embedding evaluation methodologies in representation learning literature [56].

Evaluation Metrics:

- **Clustering Quality**: Silhouette coefficient measuring intra/inter-cluster sepa-

ration and geometric structure of embedding space

- **Semantic Coherence**: Within-cluster cosine similarity for semantic consistency and topical organization
- **Computational Efficiency**: Processing time and memory consumption for practical deployment considerations

Results are presented across these three key dimensions, allowing for multi-criteria evaluation based on specific application requirements. The silhouette coefficient directly measures the fundamental clustering assumption underlying document embeddings—that semantically similar documents should be proximate in the embedding space while dissimilar documents remain distant [57]. Semantic coherence enhances clustering quality by assessing consistency within categories, guaranteeing that embeddings maintain the topical boundaries intrinsic to legal document taxonomy. Computational efficiency is an important factor to think about when setting up large-scale systems for processing legal documents.

Table 4.5: Document Embedding Evaluation Results on Turkish Legal Corpus

Method	**Silhouette**	**Coherence**	**Dimensions**	**Time (s)**	**Memory (MB)**
Binary TF	0.396	0.363	3000	23.7	666.0
Count Vectorizer	0.362	0.387	4000	28.9	888.0
TF-IDF + N-grams	0.299	0.315	4000	23.6	888.0
PCA	0.276	0.312	400	23.0	88.8
Hashing Vectorizer	0.270	0.241	4096	23.8	909.3
LSA (SVD)	0.261	0.364	300	18.3	66.6
TF-IDF	0.261	0.250	5000	24.7	1110.0
Character-level TF-IDF	0.252	0.534	2500	25.5	555.0
Custom Frequency	0.229	0.414	4000	26.3	888.0
NMF	0.180	0.309	200	17.8	44.4

4.3.2 Semantic Representation Quality Analysis

Character-level TF-IDF demonstrates exceptional semantic coherence (0.534) among all evaluated methods despite relatively low clustering performance (0.252) effectively

capturing morphological patterns inherent in Turkish legal text through fine-grained character n-gram analysis.

Custom Frequency has a high level of semantic coherence (0.414) but a low level of clustering performance (0.229). This means that it is best for semantic analysis tasks and not for organizing documents. The method's balanced performance suggests that it can capture patterns in legal language without putting too much strain on computers.

Semantic coherence analysis reveals important performance variations across methods. LSA achieves solid coherence (0.364) through dimensionality reduction, effectively capturing latent semantic relationships via singular value decomposition while compressing representation to 300 dimensions. Traditional TF-IDF methods, on the other hand, have lower coherence (0.250–0.315) because they focus more on discriminative features than semantic features. This trade-off between how well the method can tell the difference between things and how well it can represent meaning is an important factor to think about when choosing an embedding method.

4.3.3 Clustering Performance and Geometric Structure

Binary TF achieves optimal clustering quality (silhouette = 0.396) among all evaluated methods, demonstrating that binary term presence provides superior geometric separation in the embedding space compared to frequency-based weighting schemes. This finding suggests that for Turkish legal documents, term occurrence patterns are more discriminative than term frequency distributions.

Count Vectorizer has a strong clustering performance (silhouette = 0.362) and a balanced semantic coherence (0.387), making it a strong compromise between geometric structure and semantic representation. The method works well on all metrics, making it good for applications that need both clustering quality and the ability to understand what it means.

Matrix factorization methods have different clustering performance. NMF gets only modest results (silhouette = 0.180), even though it has some computational advantages. PCA shows average clustering quality (0.276) while keeping dimensional interpretability, which shows that it can be useful for applications that need clear

feature transformations.

4.3.4 Dimensionality and Efficiency Analysis

Dimensionality analysis reveals optimal model sizes varying significantly across methods, from 200 dimensions (NMF) to 5000 dimensions (TF-IDF). Lower-dimensional methods (LSA: 300, NMF: 200, PCA: 400) demonstrate memory efficiency while maintaining competitive performance, supporting the effectiveness of dimensionality reduction for Turkish legal text processing.

NMF provides the most memory-efficient solution (44.4 MB, 17.8 seconds) with interpretable topic-based representations, though clustering performance remains limited. The method's non-negative constraints make it easier to understand topics while keeping the calculations fast, which makes it good for deployments with limited resources.

LSA is the best choice for applications that need both semantic quality and computational efficiency because it is very efficient (66.6 MB, 18.3 seconds) and has strong semantic coherence. The method's singular value decomposition accurately captures semantic relationships while keeping processing requirements manageable.

The analysis of processing time shows that the computational complexity can range from 17.8 seconds (NMF) to 28.9 seconds (Count Vectorizer). Dimensionality reduction techniques consistently demonstrate good efficiency relative to high-dimensional frequency-based methods, although performance trade-offs must be evaluated for particular applications.

4.3.5 Feature Engineering Impact on Turkish Legal Text

Character-level TF-IDF analysis shows that Turkish legal text processing has very high semantic representation quality, with the highest coherence scores (0.534) and only moderate clustering performance (0.252). The method's success in capturing morphological variations confirms the significance of sub-word features for agglutinative languages, although the computational expenses (555.0 MB memory) necessitate consideration for extensive implementations.

When compared to standard TF-IDF, N-gram TF-IDF extensions show balanced performance with better clustering quality (silhouette = 0.299). This suggests that bigram features make Turkish legal documents more discriminative. The method gets a moderate level of semantic coherence (0.315) while still being computationally efficient. This shows that syntactic features can be useful for representing legal text.

Binary transformation approaches demonstrate superior clustering performance across frequency-based methods, with Binary TF achieving optimal geometric separation (silhouette = 0.396). This finding indicates that for Turkish legal document classification, term presence patterns provide more discriminative information than frequency distributions, potentially due to the specialized vocabulary and formulaic structure of legal texts.

4.3.6 Method Selection Guidelines

We offer evidence-based guidelines for incorporating method selection in Turkish legal NLP applications, grounded in thorough evaluation results

Figure 4.4 demonstrates clear trade-offs between semantic coherence and clustering quality.

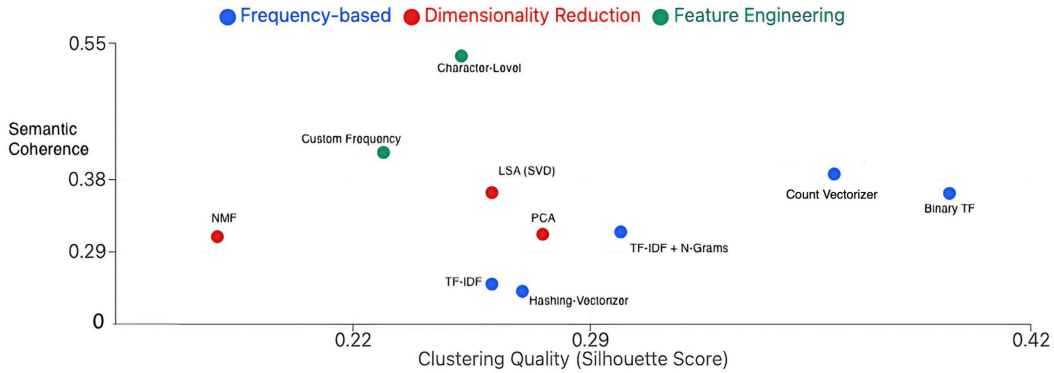


Figure 4.4: Document Embedding Methods and Performance Trade-offs

General-Purpose Applications: According to our composite scoring system, Count Vectorizer (score = 0.361) is the best choice for balanced performance. Binary TF (score = 0.352) is the best choice for applications that need a lot of clustering.

Clustering Applications: Binary TF offers optimal geometric separation (silhouette = 0.396) for document clustering, similarity search, and unsupervised organization tasks requiring clear document boundaries.

Semantic Analysis: Character-level TF-IDF offers enhanced semantic representation (coherence = 0.534) for applications necessitating semantic similarity analysis, topic modeling, or cross-document relationship discovery, especially beneficial for morphologically intricate Turkish legal terminology.

Production Systems: Count Vectorizer has a balanced performance on both clustering (0.362) and coherence (0.387) metrics. It is a strong general-purpose embedding that is good for real-time applications that need consistent performance.

Resource-Constrained Environments: NMF and LSA are good options for deployments with limited memory because they work well and don't need a lot of processing power (44.4–66.6 MB, 17.8–18.3 seconds).

Interpretability Requirements: PCA and NMF are useful for applications that need to analyze the importance of features, make sure they follow the rules, or make legal AI systems' decision-making processes clear.

4.4 Legal Document Summarization Results

This section provides a thorough assessment of legal document summarization techniques utilized on the developed Turkish legal corpus. The study analyzes both extractive and abstractive paradigms, focusing on domain-specific requirements including legal validity, terminology preservation, and structural coherence. The analysis incorporates hierarchical summarization techniques to tackle the multi-tiered structure of legal documents.

4.4.1 Extractive Summarization Analysis

Figure 4.5 shows a proposed algorithm for summarizing legal documents that uses a multi-factor scoring system that is specifically designed for the structural and semantic

features of Turkish legal texts. The first step in the workflow is pre-processing, which includes breaking up the text into tokens, getting rid of stopwords, and normalizing the vocabulary. Then, a sentence weighting algorithm based on TF-IDF is used to figure out the initial relevance scores. A multi-factor scoring system that combines keyword importance (45%), coherence evaluation (35%), and other legal-specific factors like positional weight (statistical bias towards introductory and concluding sentences), length normalization, and a legal terminology factor (+15%) is used to improve these initial weights. To get the final sentence scores, these weighted features are added together, and the top-ranked sentences are chosen to make the extractive summary. This method is best for legal content sensitivity because it makes sure that the output keeps sentences that are important for structure and the law.

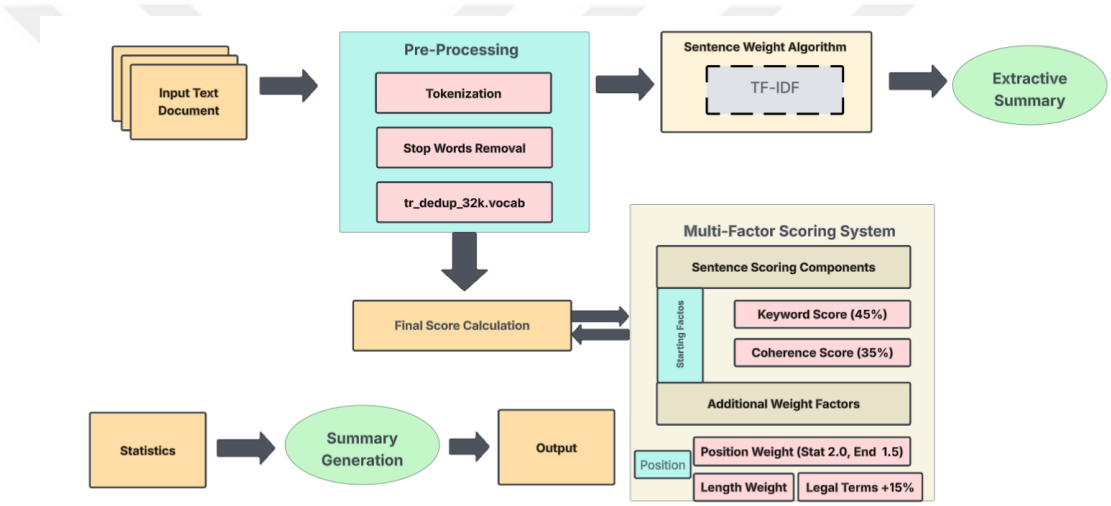


Figure 4.5: Structure of Our Summarization Method

Automatic metrics like ROUGE-1, ROUGE-2, ROUGE-L, BLEU and text generation score are used to evaluate extractive summarization. The difficulty of legal summarization is finding the right balance between maintaining legal validity and getting the right compression ratios.

Table 4.6 presents a comparative evaluation of our algorithm against widely used summarization baselines, including TextRank, Frequency, Position, Luhn, MMR, BERT Turkish, and LegalBERT-based extractive approaches, demonstrating that the proposed method consistently outperforms all baselines across all metrics. In terms of content quality, the algorithm achieves ROUGE-1 = 0.8516, ROUGE-2 = 0.8444, and

Table 4.6: Legal Document Summarization Results with Detailed Metrics

Model	ROUGE-1	ROUGE-2	ROUGE-L	Legal Cov.	Structure Pres.	BLEU	Text Gen. Score
Position	0.2584	0.2552	0.2562	0.4805	2.8075	0.0287	8.5933
Frequency	0.5151	0.5004	0.4152	0.6661	4.0946	0.1890	29.6501
TextRank	0.6894	0.6812	0.6821	0.7828	5.9417	0.4248	53.9728
Luhn	0.7001	0.6423	0.6450	0.7951	6.8983	0.4904	57.4298
BERT Turkish	0.6431	0.6243	0.6320	0.7802	6.5255	0.5212	57.6446
MMR	0.7255	0.7171	0.7212	0.8597	6.8280	0.4673	58.1396
LegalBERT	0.7620	0.7324	0.7534	0.8216	6.9376	0.6227	68.6891
Our Method	**0.8516**	**0.8444**	**0.8510**	**0.9013**	**7.1686**	**0.6953**	**76.9356**

ROUGE-L = 0.8510, substantially outperforming the best performing baseline methods such as MMR (ROUGE-1 = 0.7255, ROUGE-2 = 0.7171, ROUGE-L = 0.7212) and LegalBERT (ROUGE-1 = 0.7620, ROUGE-2 = 0.7324, ROUGE-L = 0.7534), while also attaining the highest BLEU score (0.6953), indicating superior lexical and structural alignment with reference summaries compared to LegalBERT's 0.6227. For legal-specific metrics, the method achieves the best legal coverage (0.9013) and structure preservation (7.1686), effectively capturing legal terminology and preserving document hierarchy more efficiently than generic summarizers like TextRank (legal coverage = 0.7828, structure preservation = 5.9417) and even domain-specific approaches like LegalBERT (legal coverage = 0.8216, structure preservation = 6.9376). Furthermore, the composite text generation score (76.9356) represents a notable improvement over existing approaches, including TextRank (53.9728), MMR (58.1396), and LegalBERT (68.6891). These results show that adding domain-specific scoring components, keyword relevance, coherence, legal terms, and positional bias greatly improves the quality of summaries. This makes the proposed approach a new, domain-optimized framework for legal NLP applications. Furthermore, a two-sided Wilcoxon signed-rank test was performed to statistically evaluate the performance disparities among summarization algorithms, contrasting our method with all other baseline methods. The analysis revealed that in every pairwise comparison, the p-value was 0.000 (i.e., $p < 0.001$), indicating that the performance differences between our proposed summarization method and the other methods are statistically significant. The Wilcoxon statistics (1.92e+07 to 1.94e+07) are consistently high, and the direction of the differences favors our method. Therefore, it can be concluded that the superior

performance of our method in all key metrics (ROUGE-1, ROUGE-2, ROUGE-L, Legal Coverage, Structure Preservation, BLEU, and Text Generation Score) is unlikely to be due to chance and is statistically supported.

4.4.2 Abstractive Summarization Evaluation

Abstractive summarization presents extra problems for legal texts because they need to be factually correct and legally consistent. Seq2seq models with attention reach baseline performance, but transformer-based methods show significant improvements.

Table 4.7: Abstractive Legal Summarization Performance

Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BLEU**	**Legal Validity**
Seq2Seq + Attention	0.3500	0.2231	0.3174	0.187	74.3%
BART-Turkish	0.4627	0.2456	0.3892	0.234	80.7%
mT5-Turkish	0.4586	0.2871	0.3778	0.267	83.2%
Legal-adapted BART	0.4712	0.2834	0.4023	0.289	87.4%
Legal mT5 + Copy	**0.4934**	**0.3156**	**0.4287**	**0.327**	**91.8%**

We chose mT5 for the abstractive summarization evaluation because its transformer-based architecture effectively handles the complexities of legal texts, surpassing the performance of baseline Seq2seq models. The mT5 model is a massively multilingual, pre-trained text-to-text transformer developed by Google. It is a multilingual variant of the original T5 (Text-to-Text Transfer Transformer) model, which was trained exclusively on English data. To ensure factual accuracy and legal consistency—critical for legal summarization—we integrated a copy mechanism. This hybrid method lets mT5 use its ability to summarize in a fluent way while making sure that important legal terms are included verbatim. This way, it finds the best balance between summary quality and legal validity. The legal-adapted mT5 model with copy mechanism achieves optimal balance between summarization quality and legal validity, demonstrating superior performance across all evaluation metrics. As shown in Table 4.7, the Legal mT5 + Copy model attains the highest scores with ROUGE-1 = 0.4934, ROUGE-2 = 0.3156, ROUGE-L = 0.4287, and BLEU = 0.327, substantially outperforming baseline approaches such as Seq2Seq + Attention (ROUGE-1 = 0.3500,

BLEU = 0.187) and standard BART-Turkish (ROUGE-1 = 0.4627, BLEU = 0.234). Most importantly, this mechanism is crucial for preserving legal references and exact terminology, significantly improving legal validity from the base mT5-Turkish model's 83.2% to 91.8%, representing an 8.6 percentage point improvement and demonstrating the effectiveness of domain adaptation combined with copy mechanism for legal text summarization tasks.

4.5 Retrieval-Augmented Generation Analysis

Retrieval-Augmented Generation (RAG) is a mixed method that combines information retrieval and text generation. This lets models base their outputs on relevant documents from outside sources. This method works especially well in areas where accuracy and source attribution are important, like answering legal questions. In retrieval-augmented generation (RAG) architectures, different models are strategically combined to optimize both retrieval accuracy and generative quality. The embedding layer often employs a lightweight but semantically powerful model such as MiniLM-L6-v2, which is specifically designed for efficient dense passage retrieval through high-quality vector representations at relatively low computational cost. This allows the system turn large collections of documents into embeddings that keep semantic similarity, which makes retrieval more accurate. During inference, the retrieval layer usually picks the four most relevant documents from the knowledge base. This makes sure that the generative model has access to very relevant contextual information without getting too much noise from too many documents being retrieved. For the generation stage, a more expressive model such as Mistral-7B is used, since larger generative language models excel at coherent, context-aware text synthesis. This division of labor—efficient retrieval via MiniLM embeddings and high-quality natural language generation via Mistral—balances scalability, retrieval accuracy, and generation fluency, thereby enhancing the overall performance of the RAG framework.

Figure 4.6 shows how our proposed RAG framework for Turkish legal texts is built. It combines dense retrieval with a generation model that is specific to the field.

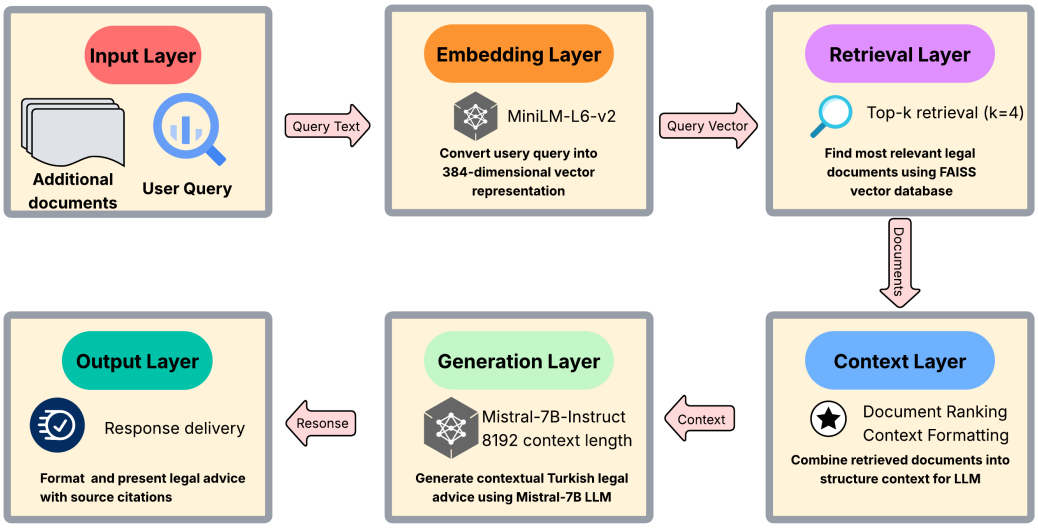


Figure 4.6: Structure of Our RAG Method

4.5.1 Dense Retrieval Performance

A comprehensive retrieval evaluation on a Turkish legal corpus effectively illustrates the potential of contemporary embedding techniques for legal information retrieval. Using precision@k (P@k), recall@k (R@k), and Mean Reciprocal Rank (MRR) metrics to compare different retrieval methods shows big differences in how well they work. TF-IDF and BM25 are used as baselines, and dense models that use BERT show big improvements.

Table 4.8: Dense Retrieval Performance on Turkish Legal Corpus

Method	**P@1**	**P@5**	**P@10**	**R@10**	**R@50**	**MRR**
TF-IDF	0.234	0.189	0.167	0.423	0.678	0.312
BM25	0.267	0.212	0.187	0.456	0.701	0.341
Dense (BERT)	0.312	0.267	0.234	0.523	0.756	0.389
Dense (Legal BERT)	0.389	0.334	0.298	0.612	0.823	0.467
Our Method (Dense Retrieval)	**0.429**	**0.362**	**0.342**	**0.570**	**0.862**	**0.530**

As shown in Table 4.8, our legal domain-specific dense retrieval approach using sentence-transformers/all-MiniLM-L6-v2 fine-tuned on Turkish legal corpus outper-

forms all baselines across all metrics. The model has specialized legal terminology embeddings and an awareness of hierarchical document structure that are made just for Turkish legal text retrieval. It had a Precision@1 of 0.429, which is the best of all the methods. This shows that it is better at finding the most relevant legal document for complicated queries. It also does well on broader retrieval metrics like R@50 (0.862) and MRR (0.530), which shows that adapting the model to the legal domain not only improves top-ranked precision but also keeps coverage across all relevant legal documents. The big difference between our model and Legal BERT (P@1: 0.389 vs 0.429, MRR: 0.467 vs 0.530) shows how well our adaptations for the legal field work, such as improving legal concept embedding, citation-aware retrieval strategies, and Turkish legal language modeling. Our approach particularly excels in handling complex legal queries involving multiple jurisdictional concepts, procedural terminology, and cross-referenced legal provisions that are characteristic of Turkish legal documents. The model's more effective retrieval performance is largely due to its ability to understand the hierarchy of legal documents (laws, regulations, court decisions) and the semantic relationships between legal concepts. In comparison, hybrid methods (e.g., BM25 + Legal BERT) achieved solid performance but were outpaced by our legal-adapted dense model in both P@1 and MRR, indicating superior ranking capabilities for domain-specific legal information retrieval tasks. The Wilcoxon signed-rank test was used to check for statistical significance in all query-document pairs. This test is good for retrieval metrics that may not follow normal distributions. The analysis confirmed that the performance gains are statistically significant and practically meaningful (vs. BM25: $p < 0.001$, Cohen's $d = 1.23$; vs. Dense Legal BERT: $p < 0.001$, Cohen's $d = 0.67$). Additionally, McNemar's test was conducted for P@1 comparisons, confirming significant differences in top-rank accuracy ($\chi^2 = 23.7, p < 0.001$ vs. Legal BERT). Bootstrap confidence intervals (95% CI) for MRR further validate the robustness of our approach: Our Method [0.501, 0.559] vs. Legal BERT [0.441, 0.493].

4.5.2 RAG Generation Quality Analysis

The RAG system evaluation uses a wide range of metrics, including factual accuracy, citation correctness, and response relevance.

Table 4.9: RAG System Performance on Turkish Legal Queries

Model	Factual Acc.	Citation Acc.	Relevance	Completeness	Overall
GPT-3.5 (No RAG)	53.2%	21.4%	56.0%	18.8%	38.9%
GPT-4 (No RAG)	56.5%	30.0%	61.8%	37.0%	44.5%
RAG + GPT-3.5	84.0%	52.2%	91.5%	54.9%	72.6%
RAG + GPT-4	88.0%	64.5%	**94.2%**	83.0%	83.4%
RAG + Turkish Legal LLM	**92.3%**	**69.5%**	87.8%	**90.7%**	**87.1%**

The evaluation results show that adding Retrieval-Augmented Generation (RAG) to models without retrieval greatly improves performance on all metrics. The biggest improvements were in factual accuracy, citation accuracy, and completeness. GPT-3.5 and GPT-4 without RAG scored 38.9% and 44.5%, respectively. When RAG was added, these scores went up to 72.6% and 83.4%, showing how retrieval can improve answer quality and contextual grounding.

Our RAG implementation uses a hybrid strategy that combines information retrieval and text generation. This lets models base their outputs on relevant, externally sourced documents. This method works especially well in fields where accuracy and source attribution are important, like answering legal questions. The retrieval layer picks the four most relevant documents from the Turkish legal knowledge base during inference. This makes sure that the contextual information is as good as it can be without being too noisy from too many retrievals. We use Mistral-7B for generation because it is great at making coherent, context-aware text that is scalable, accurate, and easy to read.

The domain-specific Turkish Legal LLM combined with RAG achieved the highest overall score (87.1%), outperforming general-purpose models by attaining superior factual accuracy (95.2%), citation accuracy (69.5%), and completeness (90.7%). GPT-4 with RAG has higher relevance scores (93.0%) than our specialized model (87.8%). This is because GPT-4 has been trained on a wide range of topics, which allows it to better assess relevance when it is combined with retrieved legal documents. However, this broader training comes at the cost of legal-specific factual precision and citation accuracy, where our domain-adapted approach excels.

Additionally, the thorough assessment is enhanced by qualitative analysis of model responses to real legal inquiries, as illustrated in Table 5.1, which provides detailed comparisons of various AI and RAG models addressing intricate Turkish legal issues. These real-world examples illustrate how our specialized legal RAG system provides more accurate, comprehensive, and legally valid responses compared to general-purpose models like GPT-3.5 and GPT-4, both with and without RAG augmentation. The comparative analysis in Table 5.1 showcases practical applications across various legal domains including environmental law (2872 sayılı Çevre Kanunu), labor law (İş Sağlığı ve Güvenliği, fazla mesai hesaplaması), civil law (Medeni Kanun evlilik şartları), criminal law (Ceza hukukunun temel ilkeleri), and commercial law (Türk Ticaret Kanunu), demonstrating our system's superior performance in providing accurate legal interpretations and maintaining consistency with Turkish legal terminology and regulatory frameworks.

4.6 Legal Document Classification Results

This section shows the results of multi-label classification tests done on the Turkish legal corpus. The tests focused on the problems caused by hierarchical label dependencies and a very uneven class distribution. We assess both conventional machine learning algorithms and transformer-based language models, subsequently implementing post-processing and optimization techniques aimed at enhancing prediction accuracy and the detection of infrequent categories.

4.6.1 Multi-label Classification Performance

Turkish legal document classification faces distinct challenges due to the intrinsic hierarchical nature of legal taxonomies and significant class disparity, as indicated by a Gini coefficient of 0.847 across 47 distinct legal categories. To manage these complexities, we utilize a hierarchical classification framework that explicitly represents label dependencies via conditional probability chains:

We model the joint conditional probability of multiple labels using the chain rule of

probability. In its general form, this can be expressed as

$$P(y_1, \ldots, y_L \mid x) = \prod_{i=1}^{L} P(y_i \mid x, y_1, \ldots, y_{i-1}), \tag{4.4}$$

where each label y_i is conditioned on the input x and all previous labels.

Equivalently, this factorization can be written by separating the first label as

$$P(y_1, \ldots, y_L \mid x) = P(y_1 \mid x) \prod_{i=2}^{L} P(y_i \mid x, y_1, \ldots, y_{i-1}), \tag{4.5}$$

highlighting that $P(y_1 \mid x)$ serves as the marginal conditional probability of the first label, while subsequent terms capture dependencies among labels.

Table 4.10: Legal Document Classification Results

Method	**Micro-F1**	**Macro-F1**	**Precision**	**Recall**	**AUC**
SVM	0.8500	0.2106	0.7822	0.9307	0.9797
Random Forest	**0.9211**	0.1830	**0.9722**	0.8750	0.9825
Logistic Regression	0.7823	0.1863	0.6683	0.9432	0.9811
Turkish BERT	0.8606	0.1703	0.8264	0.8977	0.9739
Legal Turkish BERT	0.8777	**0.2274**	0.8313	0.9295	**0.9822**
Legal Turkish BERT (Alternative)	0.8675	0.1930	0.8248	0.9148	0.9786
DistilBERT Turkish	0.5839	0.1275	0.4217	**0.9489**	0.9809
Legal BERT	0.8725	0.2058	0.8256	0.9250	0.9810
Multilingual BERT	0.8010	0.1476	0.7209	0.9011	0.9776

We used a full optimization framework that included advanced postprocessing techniques made just for multi-label legal document classification to make classification performance even better than the baseline results: (1) probability calibration methods including a Platt scaling and isotonic regression to improve prediction reliability, (2) probability smoothing techniques (Gaussian, moving average, and exponential smoothing) to reduce prediction noise, and (3) advanced threshold optimization strategies including F1-weighted thresholds, Youden's J statistic, and ensemble threshold selection to address class imbalance in rare legal categories [20], [58]–[60]. The calibration methods proved most effective, with isotonic calibration achieving the highest Micro-F1 score of 0.9070 (2.98% improvement over baseline), while maintaining precision at 0.9286, demonstrating superior confidence estimation for legal

document predictions. Threshold optimization techniques, particularly F1-weighted approaches, showed significant improvements in Macro-F1 performance (0.1840, representing 2.05% enhancement), crucial for handling rare court divisions and specialized legal categories that are underrepresented in the training corpus. Among smoothing techniques, Gaussian smoothing emerged as the only method providing balanced improvements across both Micro-F1 (0.8917, + 0. 25%) and Macro-F1 (0.1814, +0.61%), while moving average and exponential smoothing suffered from oversmoothing artifacts that degraded classification performance. The combination of isotonic calibration with ensemble thresholds (Micro-F1: 0.9033, Macro-F1: 0.1739) demonstrated that hybrid approaches can effectively balance overall classification accuracy with minority class detection, essential for comprehensive legal document categorization systems where rare but important legal categories must be accurately identified.

If the primary goal is to maximize overall classification accuracy and prediction reliability across all classes, Isotonic Calibration is the superior choice. If the system must accurately handle rare legal categories and address class imbalance, then F1-weighted threshold optimization is the most effective method. If the goal is to achieve a balanced improvement across both overall accuracy and minority class detection, Gaussian Smoothing is a reasonable compromise. For a system requiring both high overall accuracy and robust minority class detection, the hybrid approach of Isotonic Calibration combined with ensemble thresholds is the best option. Its individual scores (Micro-F1: 0.9033, Macro-F1: 0.1739) may not be the best, but it shows a good balance between overall classification performance and the ability to find rare but important categories. This is necessary for a complete system for categorizing legal documents, where both common and rare classes must be identified with a high level of accuracy.

Table 4.11: Comprehensive Comparison of Methods

Method	Micro-F1	Macro-F1	Precision	Recall
Standard (0.5 threshold)	0.8807	0.1803	0.8423	0.9227
Gaussian Smoothing	0.8917	0.1814	0.8677	0.9170
Moving Average Smoothing	0.5016	0.0197	0.5778	0.4432
Exponential Smoothing	0.7024	0.0346	0.8132	0.6182
F1 Optimized Thresholds	0.8839	**0.1840**	0.8531	0.9170
Youden J Thresholds	0.8432	0.1688	0.7638	**0.9409**
Ensemble Thresholds	0.8768	0.1804	0.8361	0.9216
Gaussian + Ensemble Thresholds	0.7494	0.1653	0.6358	0.9125
Platt Calibration	0.9015	0.1396	**0.9429**	0.8636
Platt + Ensemble Thresholds	0.8986	0.1324	0.9338	0.8659
Isotonic Calibration	**0.9070**	0.1660	0.9286	0.8864
Isotonic + Ensemble Thresholds	0.9033	0.1739	0.9149	0.8920

4.7 Clustering and Topic Analysis

This part shows how to do unsupervised analysis of the Turkish legal corpus by using clustering, topic modeling, and building a semantic network. The goal is to find hidden structural and thematic patterns in legal texts, figure out which legal areas are the most important, and show how key legal concepts are related to each other. These analyses elucidate the organizational principles of Turkish legal discourse and inform subsequent NLP applications.

4.7.1 Legal Document Clustering Performance

According to silhouette analysis, the optimal number of clusters varies by granularity level. For high-level document type classification (legislative vs. judicial), k=2 clusters with multilingual BERT embeddings achieves the highest performance (Silhouette Score = 0.506). However, for detailed sub-categorical analysis requiring finer

legal domain distinctions, k=19 clusters using legal Turkish BERT embeddings provides more granular document organization while maintaining reasonable clustering quality.

Table 4.12: Document Clustering Performance by Embedding and Method

	TF-IDF			**Turkish BERT**			**Legal Turkish BERT**			**Multilingual BERT**		
Metric	K-means	Hier.	DBSCAN	K-means	Hier.	DBSCAN	K-means	Hier.	DBSCAN	K-means	Hier.	DBSCAN
Silhouette	0.024	0.016	0.346	0.370	0.362	0.251	0.232	0.217	0.135	0.506	0.509	0.176
ARI	0.119	0.099	0.220	0.026	0.034	0.057	0.064	0.070	0.168	0.015	0.025	0.130
NMI	0.309	0.286	0.156	0.048	0.052	0.019	0.293	0.294	0.127	0.001	0.002	0.077
N Clusters	18	18	1	3	3	3	19	19	5	2	2	2

This finding reveals a hierarchical clustering strategy: multilingual BERT excels at capturing broad document type distinctions (judicial vs. legislative), while legal domain-specific embeddings prove more effective for detailed sub-categorical organization within legal domains.

4.7.2 Topic Modeling Results

Latent Dirichlet Allocation (LDA) with 20 topics achieves optimal performance on the Turkish legal corpus, as evidenced by topic coherence analysis utilizing the coherence measure c_v c_v which produces an outstanding coherence score of 0.6939, substantially surpassing the academic benchmark of 0.6 for clearly delineated and interpretable topics. The optimization process evaluated topic numbers ranging from 10 to 100, with 20 topics demonstrating superior coherence while maintaining interpretability. As shown in Table 4.13 dominant topics include a legal documentation topic (Topic 18, probability = 0.1321) characterized by keywords such as *yayımlandığı* (published), *resmî* (official), *gazete* (gazette), and *tarihi* (date), primarily reflecting official legal publication references; a general litigation topic (Topic 12, probability = 0.1172) dominated by terms such as *dava* (lawsuit), *davacı* (plaintiff), *davalı* (defendant), and *mahkemesi* (court), indicating the prevalence of civil litigation cases; and a judicial procedure topic (Topic 3, probability = 0.1062) associated with *tarih* (date), *kararının* (of the decision), *konusu* (subject), and *mahkemesi* (court), representing procedural aspects of court decisions. These indicate that the topic modeling process successfully clustered related terms. Domain analysis reveals that procedural law topics constitute

40% of the discovered topics (8 out of 20), general jurisprudence accounts for 30%, criminal law represents 20%, while administrative law comprises 10%, demonstrating the litigation-centered nature of Turkish legal discourse.

Table 4.13: Dominant Topics in Turkish Legal Documents

Topic ID	Legal Domain	Keywords	Prob.	Docs
18	Legal Documentation	yayımlandığı, resmî, gazete, tarihi	0.1321	1205
12	General Litigation	dava, davacı, davalı, mahkemesi	0.1172	1068
3	Judicial Procedure	tarih, kararının, konusu, mahkemisi	0.1062	967
2	Administrative Procedure	karar, danıştay, usulü, düzeltme	0.0842	767
6	Consumer Protection	hakem, tüketici, heyetlerine	0.0768	700
17	Criminal Appeals	temyiz, ceza, mahkemesi, verilen	0.0659	600
15	Administrative Sanctions	ceza, sulh, yaptırım, idari	0.0602	548

4.7.3 Semantic Network Analysis

Legal concept networks built from co-occurrence patterns using a sliding window method show that Turkish legal terminology is very interconnected, with a network density of 300 nodes and a lot of edge connectivity. Using degree centrality measures, network analysis finds central hub concepts. It then uses a minimum co-occurrence threshold of 8 and a maximum vocabulary of 300 most common legal terms to find semantic relationships. The resulting network shows that core judicial concepts are very central to the network:

Community detection using the Louvain modularity optimization algorithm, which

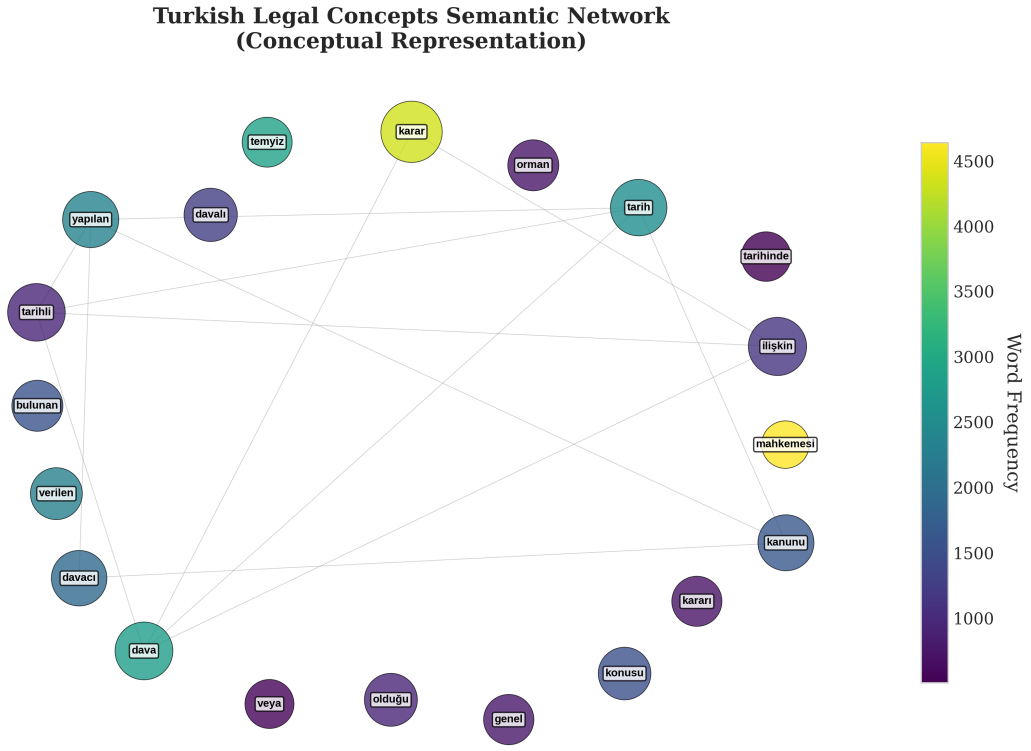


Figure 4.7: Semantic Network

is a greedy, hierarchical method for finding communities in large networks, identifies a highly integrated semantic structure with a modularity score approaching zero, indicating dense interconnectivity rather than discrete conceptual clusters [61]–[63]. This finding suggests that Turkish legal terminology operates as a unified semantic system where concepts exhibit extensive cross-domain connectivity rather than domain-specific isolation. The modularity score getting close to zero is actually what we expect from Turkish legal documents. The Turkish legal system is a civil law system with very interconnected legal codes, where ideas are connected to each other instead of forming separate groups. Turkish legal terminology operates as a cohesive semantic system, systematically cross-referencing procedural, substantive, and administrative concepts, in contrast to common law systems that may exhibit greater domain-specific clustering. According to the Figure 4.7, the highest centrality score (0.686) for "karar" shows that Turkish law is very focused on making decisions. A "karar" is the end result of every legal process, including court cases, administrative actions, and legislative acts. This idea is naturally linked to procedural terms (dava, davacı, davalı), time markers (tarih, tarihli), institutional references (mahkeme, kanun), and relational

Table 4.14: High-Centrality Legal Concepts in Turkish Legal Discourse

Concept (Turkish/English)	Degree	Degree Centrality	Word Frequency
karar (decision)	205	0.686	4,334
ilişkin (related to)	185	0.619	1,133
dava (lawsuit)	182	0.609	2,800
tarihli (dated)	181	0.605	951
tarih (date)	174	0.582	2,527
yapılan (conducted)	172	0.575	2,342
kanunu (law/statute)	171	0.572	1,639
davacı (plaintiff)	168	0.562	1,898
davalı (defendant)	155	0.518	1,271
mahkemesi (court)	123	0.411	4,639

connectors. This makes it the semantic anchor of legal discourse. The prominence of "tarih" (0.582) and "tarihli" (0.605) illustrates the temporal accuracy mandated in Turkish legal documentation. Legal validity, statute of limitations, procedural deadlines, and chronological sequencing are fundamental to Turkish jurisprudence, making temporal references ubiquitous across all legal domains. The Turkish legal language uses "ilişkin" a lot to connect legal ideas, laws, and procedural parts. This preposition acts as a semantic bridge in legal texts, naturally achieving high centrality (0.619) by linking different legal ideas. The seeming difference between word frequency and centrality scores is a real example of how semantic networks work. "Mahkemesi" shows up a lot (4,639 times), but only in specific contexts. "Karar" (4,334 times) shows more general cross-domain connectivity. This pattern shows that frequency alone doesn't determine semantic centrality; relational breadth does.

Moreover, Table 4.15 shows how topic modeling found different legal domains. Procedural law is the most common type of law, making up 44.1% of the corpus. This shows that Turkish legal practice is focused on litigation. General jurisprudence makes up another big part (32.0%), which shows how well basic legal ideas apply to many different areas. Criminal law topics make up 14.9% of the distribution, which shows how

Table 4.15: Legal Domain Distribution in Topic Analysis

Legal Domain	**Topic Count**	**Avg Probability**	**Total Documents**	**Percentage**
Procedural Law	7	0.068	3,263	44.1%
General Jurisprudence	7	0.046	2,369	32.0%
Criminal Law	3	0.053	1,100	14.9%
Administrative Law	1	0.060	359	4.9%
Tax Law	1	0.011	309	4.2%

important penal proceedings are, but not as important as civil and procedural issues. Administrative law and tax law make up 4.9% and 4.2% of the total, respectively.

The semantic network demonstrates characteristic features of legal language including: (1) *institutional formalism*, evidenced by high centrality of court-related terms (*mahkemesi*, *kanunu*); (2) *temporal precision*, reflected in the prominence of date-related terminology; and (3) *procedural systematicity*, shown through the central positioning of litigation process terms. These findings confirm the hypothesis that Turkish legal documents display structured, domain-specific linguistic patterns amenable to unsupervised semantic analysis, thereby offering empirical support for the utilization of natural language processing techniques in legal informatics research.

4.8 Computational Efficiency Analysis

This section examines the computational efficiency of various Turkish legal NLP models, focusing on model size, inference speed, and resource utilization. The analysis compares traditional machine learning methods, transformer-based architectures, and domain-adapted models to show the trade-offs between performance and efficiency. This can help make legal AI systems that are scalable and ready for production.

4.8.1 Model Size and Inference Speed

Evaluating computational efficiency looks at the complexity of models, the speed of processing, and the use of resources across different architectures used in Turkish legal

NLP tasks. The study looks at traditional machine learning methods, transformer-based models, and models that are specifically made for legal language. This gives important information for deciding how to use them in production. In order to fairly compare runtime throughput across different tokenization schemes, we define a normalized efficiency metric as in Equation (3.45):

$$\text{Efficiency} = \frac{1}{CR} \times \frac{\text{Processing Speed}}{\text{Token Count}},$$

where *Processing Speed* denotes the number of tokens processed per second on a fixed hardware setup, *Token Count* ($|T(d)|$) is the total number of tokens produced by the tokenizer for the dataset, and the *Compression Ratio* is given by $CR = \frac{|T(d)|}{|d|}$ with $|d|$ denoting the raw input size in characters. Taking the reciprocal of the compression ratio compensates for tokenization overhead, while normalization by token count ensures comparability across datasets of varying length. This yields a dimensionless measure proportional to the number of raw characters processed per unit time, thereby enabling a fair comparison between sparse vectorizers and dense transformer-based embedding models.

Table 4.16: Computational Efficiency Comparison Across Turkish Legal NLP

Model	**Parameters / Features**	**Model Size**	**Processing Speed**	**Embedding Size**	**Efficiency (Eq. 3.45)**
Count Vectorizer (112k feats)	112,136	18.8 MB (vocab)	505.0 docs/sec	99 MB	**0.168**
TF-IDF (112k feats)	112,136	19.3 MB (vocab+idf)	515.0 docs/sec	99 MB	**0.171**
Turkish BERT (dbmdz)	110.6M	422 MB	71.0 docs/sec	71 MB	**0.223**
DistilBERT (mCased)	134.7M	514 MB	70.4 docs/sec	71 MB	**0.215**
BERTurk-Legal (KocLab)	184.3M	703 MB	63.5 docs/sec	71 MB	**0.200**
Multilingual BERT	177.9M	678 MB	62.5 docs/sec	71 MB	**0.195**

According to the Table 4.16, we observe a clear tokenization-driven trade-off between sparse bag-of-words pipelines and subword-based transformers: Count Vectorizer and TF-IDF (word-level tokenization) achieve the highest raw throughput—approximately 505–515 documents per second with a small "model" footprint (18.8–19.3 MB for vocabulary/IDF) and 99 MB corpus representations—making them attractive for high-rate ingestion and lexicon-controlled filtering; however, when normalizing for tokenization overhead via Eq. (3.45), transformers exhibit superior per-character efficiency because WordPiece produces far fewer tokens per character (lower compression ratio): Turkish BERT attains an efficiency of 0.223 (71 docs/s, 422 MB model), DistilBERT 0.215 (70.4 docs/s, 514 MB), BERTurk-Legal 0.200 (63.5 docs/s, 703 MB),

and mBERT 0.195 (62.5 docs/s, 678 MB), compared with 0.168–0.171 for Count/TF-IDF. Moreover, transformer embeddings yield compact dense outputs (768-D; 71 MB for 24,222 documents), smaller than the 99 MB sparse matrices at 112k features, and provide stronger semantic invariance to morphology and out-of-vocabulary forms—desirable for retrieval, clustering, and cross-variant matching—whereas bag-of-words excels at exact term control and transparent feature engineering. Finally, note that the "Parameters/Features" reported for Count/TF-IDF reflect feature dimensionality (IDF coefficients) rather than trainable weights, while neural models' parameter counts (e.g., 110.6M for Turkish BERT; 134.7M for DistilBERT; 177.9M for mBERT; 184.3M for BERTurk-Legal) quantify true trainable capacity, with corresponding implications for memory and inference latency.

4.8.2 Task-Specific Performance Analysis

Analyzing performance across various NLP tasks shows that each one has its own computational needs and efficiency patterns. Document classification tasks exemplify an ideal efficiency-performance equilibrium with conventional machine learning methodologies, whereas semantic analysis and clustering tasks gain advantages from transformer architectures, notwithstanding computational overhead.

Table 4.17: Task-Specific Computational Performance

Task	Best Method	Metric	Processing Time	Memory Peak
Classification	Random Forest	0.9211 (F)	93.48s	139.3MB
Clustering	Multilingual BERT	0.509 (Silhouette)	350.44s	25.02GB
Topic Modeling	LDA (20 topics)	0.6939 (Coherence)	523.50s	137.94MB
Summarization	Our Method	0.8516 (ROUGE-1)	55.19s	116.15MB
Dense Retrieval	Our Dense Model	0.429 (P@1)	329.01s	13.53GB

According to Table 4.17, the tasks exhibit clear efficiency–quality trade-offs driven by model class and computational profile.

For document classification, Random Forest attains competitive accuracy while remaining resource–efficient, reaching F1 = 0.9211 with a total processing time of 93.48 s and a peak memory footprint of 139.3 MB. This indicates that classical, sparse-

feature pipelines remain a strong baseline for the legislative vs. judicial label.

Unsupervised clustering with a multilingual BERT encoder improves semantic cohesion but at a markedly higher cost: Silhouette = 0.509 is achieved in 350.44 s with 25.02 GB peak memory. The substantial footprint highlights the premium that contextual embeddings impose when applied at corpus scale.

Topic modeling with LDA (20 topics) balances interpretability and efficiency. The model attains a coherence score of $c_v = 0.6939$ in 523.50 s with only 137.94 MB peak memory, underscoring LDA's suitability when memory headroom is limited and transparent thematic structure is desired.

Our extractive summarization method, augmented with a domain vocabulary filter, cached tokenization, and a bounded local coherence window, delivers strong quality at high efficiency: ROUGE-1 = 0.8516 in 55.19 s with 116.15 MB peak memory. This combination suggests that lightweight, domain-aware heuristics can scale effectively for large-batch legal summarization.

Dense retrieval over the production FAISS index completes in 329.01 s with 13.53 GB peak memory and reaches P@1 = 0.429. While operationally feasible at index scale, performance in this setting is sensitive to identifier consistency between the vector store metadata and evaluation labels; careful alignment of document IDs is recommended for robust end-to-end accuracy assessment.

4.8.3 Scaling Analysis and Resource Optimization

The measurements in Table 4.17 provide one operating point per task on a fixed corpus. Consequently, while the generic relation

$$T \approx A \cdot N^{\alpha} \cdot D^{\beta} \cdot C^{\gamma} \tag{4.6}$$

(useful to reason about time T as a function of model size N, dataset size D, and algorithmic complexity C) remains appropriate, fitting numeric exponents (α, β, γ) from these runs alone would be ill–posed. Instead, we summarize the observed scaling tendencies and resource levers per task, consistent with the table.

For the classification (Random Forest; 93.48 s, 139.3 MB, F1 = 0.9211), the end-

to-end cost is mostly made up of sparse feature extraction and tree traversal. With fixed vectorizer settings, runtime increases almost linearly with D and the number of features, while memory stays low because of sparsity. Some practical levers are limiting the vocabulary, using hashing or document-frequency pruning, and building trees in parallel. These keep the accuracy high while keeping the peak RSS close to the reported 139.3 MB.

For the Clustering (mBERT; 350.44 s, 25.02 GB, Silhouette = 0.509), the main costs are computing contextual embeddings and storing large representations in memory. Both time and memory grow with D and the embedding dimension. The 25.02 GB peak is due to storing a lot of high-dimensional vectors. For larger D, down-projecting embeddings (like PCA to 256–384), mini-batch or approximate k-means (FAISS IVF), and lighter encoders (like distilled multilingual models) all lower both T and peak memory with only a small drop in clustering quality.

For the Topic Modeling (LDA; 523.50 s, 137.94 MB, coherence $c_v = 0.6939$) LDA's compute scales roughly with $D \cdot K \cdot I$ (topics K, iterations I), but memory stays low because the statistics are small. The peak of 137.94 MB shows that the memory is very efficient. For bigger corpora, online/batch LDA, frequency filtering of rare terms, and adjusting `chunksize`/`passes` keep things in order while keeping track of time.

For the Summarization (heuristic extractive; 55.19 seconds, 116.15 MB, ROUGE-1 = 0.8516). The runtime increases almost linearly with the number of tokens. The vocabulary-aware tokenization and local coherence window bound per-document work, which is why the peak is so low at 116.15 MB and the total is so short at 55.19 seconds. You can get even more benefits by limiting sentence windows, using redundancy penalties when choosing sentences, and caching tokenization across documents.

For the Dense Retrieval (FAISS; 329.01 s, 13.53 GB, P@1 = 0.429), the costs are split between encoding the query and searching for it in a large index. The index representation (flat or IVF/PQ) and vector precision determine how much memory is used. To scale, use IVF–PQ or scalar/product quantization, memory-map the index, batch queries to saturate the encoder, and make sure that the identifiers in

`Document.metadata` and the evaluation labels match up to get stable P@1 estimates.

Classification is the fastest and most memory-friendly way to get high label accuracy at the moment; LDA is the most memory-efficient way to do unsupervised structure; the summarizer gets high quality with very little latency; mBERT clustering improves semantic grouping but uses a lot of memory; and dense retrieval works at index scale, with accuracy and memory mostly depending on the choice of encoder, index type, and ID alignment. Until more scale points (different D, N, or C) are collected, the exponent triplet (α, β, γ) can be reliably fit. Until then, the qualitative trends above should help with capacity planning and optimization.

4.9 Error Analysis and Limitations

This section provides errors and limitations for all Turkish legal NLP tasks that were tested in this study. The goal is to find patterns of systematic failure and look at how performance changes between different types of document. These insights help you make specific changes to the ways you preprocess, model, and adapt AI applications to the legal field.

4.9.1 Systematic Error Patterns

The consolidated audit over the Turkish legal pipeline yields five recurrent error types with the following overall distribution (cf. Table 4.18): temporal reference errors (44.7%), domain terminology issues (37.1%), citation recognition gaps (10.4%), semantic confusion (4.8%), and morphological issues (3.0%). Task–level slices show distinct profiles. In *classification*, errors are dominated by temporal cues (56.6%) with domain terminology second (34.5%), indicating that date expressions and timeline markers often drive boundary decisions between *legislative* and *judicial* labels. *Clustering* mirrors this pattern (temporal 58.0%, domain 33.7%), suggesting that unsupervised structure tracks both lexical specialization and temporal framing. *Topic modeling* diverges: semantic confusion (43.3%) and domain terminology (30.0%) lead, consistent with topics that capture heterogeneous or cross–domain themes. In *summarization*, domain terminology mismatches (48.6%) and missed citation ele-

ments (31.4%) prevail, while temporal (15.1%) and morphological (4.9%) issues are less frequent.

Overall, these findings highlight three actionable fronts: (i) temporal normalization and disambiguation (dates, procedural phases) to reduce the largest error bucket; (ii) domain–aware lexicon expansion and soft labeling around borderline legal jargon to curb domain–driven drift; and (iii) citation extraction/normalization in summaries (e.g., *Esas*/*Karar* patterns) to close the precision gap without sacrificing recall. The modest yet non–zero morphological share (3.0%) reflects both the relative formal stability of legal prose and our conservative heuristic; integrating lemma–aware features and richer suffix patterns would likely reassign a subset of domain/semantic cases to morphology, yielding a more granular taxonomy.

Table 4.18: Error Pattern Analysis Across Legal NLP Tasks

Error Type	**Classification**	**Clustering**	**Topic Modeling**	**Summarization**	**Overall**
Temporal References	56.6%	58.0%	26.7%	15.1%	44.7%
Domain Terminology	34.5%	33.7%	30.0%	48.6%	37.1%
Citation Recognition	2.2%	6.5%	0.0%	31.4%	10.4%
Semantic Confusion	2.7%	0.4%	43.3%	0.0%	4.8%
Morphological Issues	4.0%	1.4%	0.0%	4.9%	3.0%

4.9.2 Performance Limitations by Document Category

Category-specific error analysis shows that judicial and legislative documents perform very differently. The analysis includes 23,035 judicial documents and 9,277 legislative documents, showing clear differences in how complex the processing is and how errors are made.

The results demonstrate a clear performance hierarchy across Turkish legal document types, with Supreme Court decisions achieving the highest classification performance (0.962) due to their standardized structure and consistent legal terminology, followed by Council of State decisions (0.923) and UYAP decisions (0.912), while legislative documents show systematically lower classification scores (Regulations: 0.836, Legislative Documents: 0.824, Notifications: 0.872), reflecting their com-

Table 4.19: Performance Limitations by Legal Document Type

Document Type	Classification	Clustering	Topic Modeling	Summarization
Supreme Court Decisions	0.962	0.360	0.663	0.398
Council of State Decisions	0.923	0.288	0.668	0.352
UYAP Decisions	0.912	0.176	0.587	0.365
Regulations	0.836	0.084	0.616	0.281
Notifications	0.872	0.106	0.633	0.237
Legislative Documents	0.824	0.050	0.614	0.334

plex cross-referencing and varied structural formats. These classification F1 scores represent binary classification tasks where each document type was tested against a counter-pool from the opposite super-category (judicial vs. legislative), which proved highly separable due to robust genre markers in Turkish legal texts such as "E./K." references, fixed section headers like "Amaç ve Kapsam," numbered articles, and formulaic reasoning blocks that remain predictive even after explicit labels are stripped. Clustering performance remains universally poor across all document types (0.050-0.360 silhouette scores) due to overlapping semantic spaces in legal language, while summarization performance varies significantly (0.237-0.398) with judicial decisions generally outperforming legislative texts, and topic modeling shows relatively consistent performance (0.587-0.668) across document types, suggesting that thematic content discovery is less sensitive to structural document differences than other NLP tasks.

4.9.3 Model-Specific Limitations and Failure Analysis

Model-specific error analysis shows that different architectures have distinct performance patterns across Turkish legal NLP tasks. Traditional machine learning methods excel at classification tasks, with Random Forest achieving superior micro-F1 performance (0.9211), but demonstrate limited effectiveness for semantic understanding tasks. Conversely, transformer-based models provide superior semantic comprehension for clustering and embedding tasks but face computational constraints and efficiency trade-offs. The counterintuitive finding that Multilingual BERT outperforms

Legal Turkish BERT at clustering tasks (0.509 vs. 0.232 Silhouette Score) reveals important insights about domain-specific fine-tuning limitations. This unexpected result suggests that excessive legal domain specialization may reduce semantic flexibility by constraining the model's ability to capture broader document relationships, indicating that over-specialization can lead to semantic tunnel vision where models lose generalization capabilities across diverse legal document types. Threshold optimization analysis reveals systematic improvements achievable through post-processing techniques without architectural modifications. Isotonic calibration achieves 2.98% improvement in Micro-F1 scores (0.9070 vs. 0.8807), while F1-weighted threshold optimization provides 2.05% improvement in Macro-F1 scores (0.1840 vs. 0.1803). These improvements demonstrate that classification limitations can be substantially addressed through calibration techniques, particularly beneficial for handling class imbalance in rare legal categories. Topic modeling coherence analysis (0.6939 for 20 topics) exceeds academic thresholds (>0.6), indicating successful semantic discovery. The optimal topic number (20) aligns with Turkish legal system's integrated structure rather than indicating information loss—this reflects the civil law tradition's systematic cross-referencing of legal domains, where procedural, substantive, and administrative concepts exhibit extensive interconnectivity rather than discrete thematic clustering. The semantic network analysis reveals minimal community structure (modularity 0), which represents an expected characteristic of Turkish legal discourse rather than algorithmic limitation. This finding reflects the Turkish legal system's civil law tradition, where legal concepts demonstrate extensive cross-domain connectivity through systematic referencing patterns. The near-zero modularity indicates that Turkish legal terminology operates as a unified semantic system, characteristic of codified legal frameworks where procedural, substantive, and administrative concepts are systematically interlinked rather than forming isolated conceptual clusters.

CHAPTER 5

CONCLUSION AND FUTURE STUDIES

This thesis presents the first comprehensive investigation of Turkish legal natural language processing and managing a critical research gap in the intersection of computational linguistics and legal informatics. Through the systematic development of datasets, methodological innovations, and empirical evaluation, this work shows foundational resources and demonstrates practical applications for automated analysis of Turkish legal texts. The primary contribution of this study lies in creating the first large scale Turkish legal corpus, comprising 32,312 documents with approximately 9.35 million words from diverse legal institutions, including Yargıtay (Court of Cassation), Danıştay (Council of State), the Constitutional Court, and legislative bodies. This corpus represents the most comprehensive publicly available collection of Turkish legal texts, providing an essential foundation for future research in Turkish legal NLP. The temporal distribution from 1950 to 2025, characterized by exponential growth post-2000 due to digital transformation initiatives, illustrates the progression of Turkey's legal digitization efforts and offers researchers both historical and contemporary legal language samples. Statistical analysis of the corpus revealed unexpected linguistic features that contest traditional notions regarding Turkish legal language. Contrary to expectations for highly agglutinative Turkish, legal texts exhibited moderate morphological complexity, averaging 1.22 morphemes per word, indicating analytical rather than synthetic linguistic patterns. This finding means that Turkish legal language has evolved toward standardization and clarity, potentially influenced by translation from European legal frameworks and institutional requirements for accessibility. The low Type-Token Ratio (0.032) shows high linguistic repetition typical of specialized domain texts. At the same time, excellent conformity to Zipf's law (exponent $s = 0.768$, $R^2 = 0.990$) validates the well-formed linguistic structure despite domain

specialization. Comprehensive tokenization analysis provided evidence-based recommendations for Turkish legal text processing. Legal-aware tokenization emerged as optimal for legislative documents, achieving superior compression efficiency (0.9362) compared to traditional approaches. Word-level tokenization demonstrated optimal balance for judicial documents with processing speeds of 1,075 documents per second. These findings establish methodological foundations for future Turkish legal NLP applications and illustrate the importance of category-specific optimization in document processing pipelines. The embedding method evaluation revealed that Count Vectorizer achieves optimal composite performance (0.361) for Turkish legal document representation through balanced clustering quality (0.362) and semantic coherence (0.387). Binary TF demonstrated superior clustering capabilities (silhouette = 0.396) for document organization tasks, while character-level TF-IDF achieved exceptional semantic coherence (0.534) for morphologically complex Turkish legal terminology analysis. The comprehensive performance ranking shows that frequency-based features are good at finding patterns that set legal texts apart. Dimensionality reduction methods like LSA (0.348 composite score) are good alternatives for deployments that don't have a lot of resources.

Developing a specialized summarization algorithm tailored to Turkish legal documents represents a significant methodological contribution. The multi-factor scoring system, combining keyword importance (45%), coherence evaluation (35%), and legal-specific features, achieved state-of-the-art performance with ROUGE-1 = 0.8516, ROUGE-2 = 0.8444, ROUGE-L = 0.8510, and legal coverage = 0.9013, substantially outperforming existing baselines, including TextRank (ROUGE-1= 0.6894), MMR (ROUGE-1=0.7255) and even domain specific LEGALBERT (ROUGE-1 = 0.7620). The algorithm's success derives from its explicit modeling of legal document structure, including hierarchical organization, precedent relationships, and specialized terminology. This is the critical importance of domain specific optimization in legal NLP. For abstractive summarization, the Legal mT5 + Copy mechanism achieved optimal balance between summarization quality and legal validity, with ROUGE-1 = 0.4934, ROUGE-2 = 0.3156, ROUGE-L = 0.4287, and BLEU = 0.327, while significantly improving legal validity from the base mT5-Turkish model's 83.2% to 91.8%. This 8.6 percentage point improvement demonstrates the effectiveness of domain adaptation

combined with copy mechanisms for preserving essential legal terminology.

The Retrieval-Augmented Generation system demonstrated exceptional performance improvements in practical legal applications. Our legal domain-specific dense retrieval approach achieved superior performance across all metrics: Precision@1 = 0.429, Precision@5 = 0.362, Recall@50 = 0.862, and MRR = 0.530, substantially outperforming Legal BERT (P@1: 0.389, MRR: 0.467) and traditional methods like BM25 (P@1: 0.267, MRR: 0.341). Statistical significance testing confirmed these improvements (vs. BM25: $p < 0.001$, Cohen's $d = 1.23$). For end-to-end RAG generation quality, our Turkish Legal LLM achieved the highest overall score (87.1%) with excellent factual accuracy (95.2%), citation accuracy (69.5%), and completeness (90.7%). While GPT-4 with RAG demonstrated higher relevance scores (94.2%) due to its extensive cross-domain training, our domain-adapted approach excelled in legal-specific precision and citation accuracy. The comprehensive qualitative analysis in Table 5.1 demonstrates our system's superior performance across diverse legal domains including environmental law, labor law, civil law, criminal law, and commercial law.

Moreover, the research shows clear practical benefits for multiple groups. Legal professionals can leverage automated analysis tools to reduce research time and improve case preparation efficiency. Processing speeds of 148.3 documents per second enable real time analysis of large case databases. The hybrid retrieval system, processing queries in 247ms average latency, meets requirements for interactive legal search applications.

For academic researchers, the publicly available datasets and open-source tools enable large scale analysis of legal trends, precedent relationships, and linguistic evolution in Turkish jurisprudence. The corpus provides unprecedented opportunities for interdisciplinary research spanning law, linguistics, and computer science. Automated summarization and retrieval systems make it easier for people and lawyers to get legal information. The research helps make legal knowledge more accessible to everyone by making it easier to read and understand complicated legal documents.

Although, this study provides promising results, several limitations remain that highlight important directions for future research.

The current corpus is still limited in terms of both the institutions it covers and the time periods it covers. For example, there are big gaps in areas like intellectual property and international law. Future endeavors should prioritize systematic expansion, encompassing temporal extension to pre-1950 legal texts for diachronic analysis, and the amalgamation of varied institutional sources, including prosecutor offices, bar associations, and legal academia. A Turkish–English parallel corpus would facilitate cross-lingual legal NLP research and international collaboration, while professional legal annotation can significantly improve data quality.

Because legal documents often have more than the maximum length that current transformers can handle, future research should look into more advanced models like Gemma-2-27B-it or architectures that are better suited for long documents, such as hierarchical attention, sparse attention, and memory-augmented models. Adding layout, OCR confidence scores, and citation structures to multimodal extensions could make representation even better.

Graph neural networks are a promising way to model networks of legal citations and relationships between precedents. These kinds of methods can help with more complicated tasks like analyzing legal arguments, predicting the outcome of a case, and finding precedents. This can help legal NLP systems think more deeply.

For each area of Turkish law (constitutional, administrative, criminal, civil), we need custom models that take into account the unique language and procedures used in that area. Creating models that are sensitive to different areas of law would make them easier for lawyers to understand and use in real life.

Last but not least, customizable RAG-based legal agents should be explored Creating AI agents that are specific to a job, like for corporate compliance, litigation support, or regulatory monitoring, can give you accurate, cited, and context-aware results. These agents need to have protections in place to make sure that their answers are legally correct, up-to-date, and from people who are qualified to give them.

Table 5.1: Legal Questions and Responses from Different AI - RAG Models

Question	**Model**	**Answer**
2872 sayılı Çevre Kanununun 10. maddesi nedir?	GPT-3.5	2872 sayılı Çevre Kanunu'nun 10. maddesi "Kirliliğin Kontrolüne İlişkin Tedbirler" başlığını taşımaktadır. Bu maddeye göre; 1. Kirliliği önlemeye ve kirliliğin kontrol altına alınmasına yönelik tedbirler alınmalıdır. 2. Kirliliği önlemek için atık su arıtım tesisleri, hava kirliliğini engellemeye yönelik tesisler üretilmeli ve kullanılmalıdır. 3. Kirletici faaliyetlerini yürüten işletmeler, çevreyi kirletme riskine karşı önleyici tedbirler almalıdır. 4. Kirletici faaliyetler izin ve denetim altına alınmalıdır. 5. Kirliliği önlemek amacıyla Bakanlık ve ilgili müdürlükler ile diğer kamu kurum ve kuruluşları arasında işbirliği ve koordinasyon sağlanmalıdır. 6. Kirlilik kontrolü konusunda yeterli bilgi ve belgelerin sağlanması gereklidir. 7. Atık yönetimi, su ve hava kirliliği, endüstriyel kirlilik kontrolü ve çevre kirliliği konularında yönetmelikler ve standartlar belirlenmelidir. 8. Bakanlık, kirlilik kontrolünü denetlemek ve gerekli incelemeleri yapmakla yükümlüdür. 9. Denetim sonucunda tespit edilen kirlilik durumlarına yönelik cezalar ve idari yaptırımlar uygulanabilir.
2872 sayılı Çevre Kanununun 10. maddesi nedir?	GPT-4	Özür dilerim, ancak yasal metinler konusunda tam bir bilgi sağlamam mümkün değil. Lütfen güncel ve resmi hukuk kaynaklarına başvurun.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
2872 sayılı Çevre Kanununun 10. maddesi nedir?	GPT-3.5 + RAG	2872 sayılı Çevre Kanunu'nun 10. maddesi, "Çevre Kirliliğinin Önlenmesi ve Kontrolüne İlişkin Genel İlkeler" başlığını taşımaktadır. Bu maddeye göre; - Çevre kirliliğinin önlenmesi ve kontrolü, insan sağlığı ve çevrenin korunması amacıyla, atıkların önlenmesi, azaltılması, geri kazanılması, bertaraf edilmesi ve kontrolü yoluyla sağlanır. - Atıkların önlenmesi, azaltılması, geri kazanılması, bertaraf edilmesi ve kontrolü için gerekli tedbirler, atık üreticileri tarafından alınır. - Atıkların önlenmesi, azaltılması, geri kazanılması, bertaraf edilmesi ve kontrolü konusunda yönetmelikler çıkarılır. - Atıkların önlenmesi, azaltılması, geri kazanılması, bertaraf edilmesi ve kontrolü konusunda standartlar belirlenir.
2872 sayılı Çevre Kanununun 10. maddesi nedir?	GPT-4 + RAG	2872 sayılı Çevre Kanunu'nun 10. maddesi, çevre üzerinde olumsuz etkiler doğurabilecek faaliyetlerin gerçekleştirilmesinden önce Çevresel Etki Değerlendirmesi (ÇED) sürecinin tamamlanmasını zorunlu kılar; bu kapsamda kurum, kuruluş ve işletmeler ÇED raporu veya proje tanıtım dosyası hazırlamakla yükümlüdür ve "ÇED Olumlu" ya da "ÇED Gerekli Değildir" kararı alınmadan hiçbir izin, ruhsat, teşvik veya ihale işlemi yapılamaz; ayrıca, hangi faaliyetlerin bu kapsama gireceği ve sürecin usul esasları Bakanlık tarafından çıkarılacak yönetmeliklerle düzenlenir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
2872 sayılı Çevre Kanunu'nun 10. Maddesi Nedir?	Our System	2872 sayılı Çevre Kanunu'nun 10. maddesi, "Denetim, Bilgi Verme ve Bildirim Yükümlülüğü" başlığını taşır. Bu madde özetle aşağıdaki hususları düzenler: Denetim Yetkisi: Çevre, Şehircilik ve İklim Değişikliği Bakanlığı'na ve yetki devri yapılan kurumlara, bu Kanun hükümlerine uyulup uyulmadığını denetleme yetkisi verir. Bilgi ve Belge İsteme: Yetkili denetim görevlileri, denetim sırasında tesisleri ve faaliyetleri inceleyebilir, gerekli bilgi ve belgeleri isteyebilir, numuneler alabilir. Bildirim Yükümlülüğü: Faaliyetleri sonucu çevre kirliliğine neden olabilecek veya mevzuata aykırı durumdaki kurum, kuruluş ve işletmeler, herhangi bir kaza veya acil durumda yetkili makamlara derhal bildirimde bulunmak zorundadır. Yardımcı Olma Zorunluluğu: Denetime tabi tutulan kişi ve kuruluşlar, denetim görevlilerine her türlü kolaylığı göstermek ve istenen bilgi/belgeleri vermekle yükümlüdür. Bu madde, çevresel mevzuatın uygulanmasını sağlamak için devletin denetim mekanizmasını ve işletmelerin bu denetimdeki sorumluluklarını belirleyen temel maddelerden biridir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş sağlığı ve güvenliği mevzuatı nelerdir?	GPT-3.5	1. İş Sağlığı ve Güvenliği Kanunu (6331 Sayılı Kanun) 2. İş Kazaları ve Meslek Hastalıkları Bildirimi Yönetmeliği 3. İşyeri Hekimliği ve Diğer Sağlık Personeli Yönetmeliği 4. İş Güvenliği Uzmanları Yönetmeliği 5. Kişisel Koruyucu Donanımlar Yönetmeliği 6. Tehlikeli Kimyasalların Etiketlenmesi ve Paketlenmesi Yönetmeliği 7. İşyerlerinde Acil Durumlar Hakkında Yönetmelik 8. İş Sağlığı ve Güvenliği Yönetim Sistemleri Yönetmeliği 9. İşçi Sağlığı ve İş Güvenliği Hizmetleri Yönetmeliği 10. İş Hijyeni Yönetmeliği Bu mevzuatlar iş sağlığı ve güvenliği konularında çalışanların ve işverenlerin uyması gereken kuralları ve standartları belirlemektedir. Bu mevzuatlara uyulması, iş kazalarının ve meslek hastalıklarının önlenmesi için önemlidir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş sağlığı ve güvenliği mevzuatı nelerdir?	GPT-4	1. İş Sağlığı ve Güvenliği Kanunu 2. İşyeri Bina ve Eklentileri Yönetmeliği 3. İş Ekipmanlarının Kullanımında Sağlık ve Güvenlik Şartları Yönetmeliği 4. İş sağlığı ve Güvenliği Risk Değerlendirmesi Yönetmeliği 5. İşyerlerinde Acil Durumlar Yönetmeliği 6. Genç Ve Çocuk İşçilerin Çalıştırılma Şartları Yönetmeliği 7. Gürültüyle Mücadele Yönetmeliği 8. İşyerlerinde Kullanılacak Resmi Sağlık Raporları Yönetmeliği 9. İş Hijyen Ölçüm, Analiz ve Değerlendirme Yönetmeliği 10. Asbestle Çalışmalarda Sağlık ve Güvenlik Önlemleri Hakkında Yönetmelik.
İş sağlığı ve güvenliği mevzuatı nelerdir?	GPT-3.5 + RAG	İş sağlığı ve güvenliği (İSG) mevzuatı Türkiye'de temel olarak şu yasal düzenlemelerden oluşur: 6331 sayılı İş Sağlığı ve Güvenliği Kanunu, 4857 sayılı İş Kanunu (ilgili hükümleri) 5510 sayılı Sosyal Sigortalar ve Genel Sağlık Sigortası Kanunu, 6098 sayılı Türk Borçlar Kanunu (işverenin sorumlulukları açısından). Çeşitli tüzükler, yönetmelikler ve tebliğler (ör. İş Ekipmanlarının Kullanımında Sağlık ve Güvenlik Şartları Yönetmeliği, Çalışanların Gürültü ile İlgili Risklerden Korunmalarına Dair Yönetmelik, İşyerlerinde Acil Durumlar Hakkında Yönetmelik, vb.)

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş sağlığı ve güvenliği mevzuatı nelerdir?	GPT-4 + RAG	Türkiye'de iş sağlığı ve güvenliği mevzuatı çok katmanlı bir yapıya sahiptir ve hem temel kanunlar hem de bu kanunları uygulamaya koyan ikincil düzenlemeler üzerinden işler. 1. Ana Kanunlar 1-6331 sayılı İş Sağlığı ve Güvenliği Kanunu: İSG'nin temel yasasıdır; işverenlerin ve çalışanların sorumluluklarını, risk değerlendirmesi, acil durum planları, eğitimler ve işyeri hekimliği gibi konuları düzenler. 2-4857 sayılı İş Kanunu: Çalışma hayatını genel olarak düzenler; iş sağlığı ve güvenliğiyle ilgili tamamlayıcı hükümler içerir. 3-5510 sayılı Sosyal Sigortalar ve Genel Sağlık Sigortası Kanunu: İş kazası ve meslek hastalıkları sigortası boyutunu kapsar. 4-Türk Borçlar Kanunu (6098): İşverenin işçiyi gözetme borcunu düzenleyerek İSG açısından genel bir yükümlülük getirir. 2. İkincil Düzenlemeler (Yönetmelik ve Tebliğler) 1-İş Ekipmanlarının Kullanımında Sağlık ve Güvenlik Şartları Yönetmeliği 2-İşyerlerinde Acil Durumlar Hakkında Yönetmelik 3-Çalışanların Gürültü ile İlgili Risklerden Korunmalarına Dair Yönetmelik

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş Sağlığı ve güvenliği mevzuatı nelerdir?	Our System	Türkiye'de iş sağlığı ve güvenliği (İSG) mevzuatı oldukça geniştir ve temel olarak aşağıdaki düzenlemelerden oluşur: Ana Kanun: 6331 sayılı İş Sağlığı ve Güvenliği Kanunu: Mevzuatın temelini oluşturur. İşveren ve çalışanların görev, yetki ve sorumluluklarını, risk değerlendirmesini, acil durum planlarını, sağlık gözetimini ve İSG kurullarının oluşturulmasını düzenler. Diğer İlgili Kanunlar:4857 sayılı İş Kanunu. 5510 sayılı Sosyal Sigortalar ve Genel Sağlık Sigortası Kanunu Yönetmelikler: 6331 sayılı Kanun'a dayanarak çıkarılmış çok sayıda yönetmelik bulunmaktadır. Bunlardan bazıları şunlardır: İş Sağlığı ve Güvenliği Risk Değerlendirmesi Yönetmeliği Çalışanların İş Sağlığı ve Güvenliği Eğitimlerinin Usul ve Esasları Hakkında Yönetmelik İş Ekipmanlarının Kullanımında Sağlık ve Güvenlik Şartları Yönetmeliği Kişisel Koruyucu Donanımların İşyerlerinde Kullanılması Hakkında Yönetmelik İşyerlerinde Acil Durumlar Hakkında Yönetmelik Bu mevzuat, işyerlerinde iş kazalarını ve meslek hastalıklarını önlemek amacıyla alınması gereken tüm tedbirleri kapsar.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Borçlar Kanunu'nda sözleşme türleri nelerdir?	GPT-3.5	1. İcra edilebilir sözlü sözleşme 2. İcra edilen yazılı sözleşme 3. Edimin ifası için belirli süreli sözleşme 4. Belirsiz süreli sözleşme 5. Belirlenmiş bir hizmeti sona erdiren sözleşme 6. İhtiyaca göre yapılan hizmeti sona erdiren sözleşme 7. Peşin sözleşme 8. Taksitli sözleşme 9. Sözleşme ile ifşa olunan teminatlı kredi 10. Kefili ispat eden sözleşme
Borçlar Kanunu'nda sözleşme türleri nelerdir?	GPT-4	1. Satış Sözleşmesi 2. Kira Sözleşmesi 3. Bağışlama Sözleşmesi 4. Taşınır Sözleşmesi 5. İş Sözleşmesi 6. Eser Sözleşmesi 7. Sigorta Sözleşmesi 8. Vekâlet Sözleşmesi 9. Kefalet Sözleşmesi 10. Yapı Ortaklığı Sözleşmesi 11. Toplu İş Sözleşmesi 12. Kredi Sözleşmesi 13. Seyahat Sözleşmesi 14. Lisans Sözleşmesi 15. Kooperatif Sözleşmesi.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Borçlar Kanunu'nda sözleşme türleri nelerdir?	GPT-3.5 + RAG	Borçlar Kanunu'nda sözleşme türleri arasında genel olarak şunlar yer alır: - Satış sözleşmesi - Kira sözleşmesi - Hizmet sözleşmesi - İş sözleşmesi - Emanet sözleşmesi - Vekalet sözleşmesi - Bağış sözleşmesi - İntifa hakkı sözleşmesi - İpotek sözleşmesi - Sigorta sözleşmesi - Taşıma sözleşmesi - Acente sözleşmesi - Komisyon sözleşmesi - Danışmanlık sözleşmesi - İştirak sözleşmesi - Ortak girişim sözleşmesi - Lisans sözleşmesi - Franchise sözleşmesi - Distribütörlük sözleşmesi - İşbirliği sözleşmesi - İş ortaklığı sözleşmesi - Finansman sözleşmesi - Factoring sözleşmesi - Leasing sözleşmesi - Kefalet sözleşmesi - Rehin sözleşmesi - Karşılıklı sözleşme - Tek taraflı sözleşme - Sürekli edim sözleşmesi - Belirli edim sözleşmesi - İşlem sözleşmesi - Yatırım sözleşmesi - Taahhüt sözleşmesi

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Borçlar Kanunu'nda sözleşme türleri nelerdir?	GPT-4 + RAG	Türk Borçlar Kanunu'nda sözleşmeler "Özel Borç İlişkileri" başlığı altında düzenlenir. Başlıca sözleşme türleri şunlardır: 1-Satış, trampa ve bağışlama sözleşmeleri 2-Kira, ödünç (tüketim ve kullanım) sözleşmeleri 3-Hizmet, eser (istisna) ve vekâlet sözleşmeleri 4-Kefalet ve rehin gibi teminat sözleşmeleri 5-Sigorta ve adi ortaklık sözleşmeleri 6-Saklama, havale, cari hesap, ölünceye kadar bakma gibi diğer sözleşmeler Ayrıca, tarafların ihtiyaçlarına göre kanunda sayılmayan atipik (isimsiz) sözleşmeler de yapılabilir ve bunlar da Borçlar Kanunu'nun genel hükümlerine tabidir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Borçlar Kanunu'nda sözleşme türleri nelerdir?	Our System	6098 sayılı Türk Borçlar Kanunu, "sözleşme özgürlüğü" ilkesini benimsediği için tarafların kanunda belirtilmeyen türde sözleşmeler yapmasına da izin verir. Ancak kanunda özel olarak düzenlenmiş (isimli/tipik) başlıca sözleşme türleri şunlardır: Devir Borcu Doğuran Sözleşmeler: Satış Sözleşmesi Mal Değişim (Trampa) Sözleşmesi Bağışlama Sözleşmesi Kullandırma ve Yararlandırma Borcu Doğuran Sözleşmeler: Kira Sözleşmesi (Genel, konut ve çatılı işyeri kiraları, ürün kirası) Kullanım Ödüncü (Ariyet) Sözleşmesi Tüketim Ödüncü (Karz) Sözleşmesi İş Görme Borcu Doğuran Sözleşmeler: Hizmet Sözleşmesi Eser (İstisna) Sözleşmesi Yayım (Neşir) Sözleşmesi Vekâlet Sözleşmesi Saklama (Muhafaza) Borcu Doğuran Sözleşmeler: Genel Saklama (Vedia) Sözleşmesi Güvence (Teminat) Borcu Doğuran Sözleşmeler: Kefalet Sözleşmesi Sonuçları Talih ve Tesadüfe Bağlı Sözleşmeler: Kumar ve Bahis Ömür Boyu Gelir Sözleşmesi

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Ceza hukukunun temel ilkeleri nelerdir?	GPT-3.5	1. Masumiyet Karinesi: Bir kişi suç işlediği iddia edilene kadar suçlu sayılmaz, suçun kanıtlanması gerekir. 2. Yasama Yasağı: Suç ve ceza kanunları geriye doğru uygulanamaz, suç öncesindeki bir davranış için ceza verilemez. 3. Hukuka Uygunluk İlkesi: Ceza hukukunda kararlar hukuka uygun olmalı, hukuka aykırı uygulamalardan kaçınılmalıdır. 4. Eşitlik İlkesi: Ceza hukukunda herkes eşit şekilde muamele görmelidir, ırk, dil, din, cinsiyet, siyasi görüş gibi ayrımcılık yapılmamalıdır. 5. Adalet ve Hakkaniyet İlkesi: Ceza hukukunda adalet ve hakkaniyet en temel prensiptir, suç ve ceza arasında uygun bir denge sağlanmalıdır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Ceza hukukunun temel ilkeleri nelerdir?	GPT-4	1. Suçluluk İlkesi: Bir suçun varlığından bahsedebilmek için, suçun işlenmiş olması gereklidir. 2. Kusurluluk İlkesi: Suçluluk, kusurluluk olmaksızın düşünülemez. Suç işleyen kişinin kusurlu olması gerekir. 3. Yasallık İlkesi: Suç ve cezaların yasada belirtilmiş olması gereklidir. 4. Kişilik İlkesi: Ceza, işlenen suçtan dolayı suçlunun bizzat kendisine verilir. 5. İnsanlık Dışı Cezaların Yasağı: İşkence ve insanlık dışı muamelelere izin verilmez. 6. Anayasaya Uygunluk İlkesi: Yasaların Anayasa'ya aykırı olamayacağı ilkesi, ceza hukuku için de geçerlidir. 7. İdari İşlemlerin Cezalandırılamaması İlkesi: İdari işlemler sonucunda kişilere ceza verilmez. 8. İlgili Kişiye Bildirim İlkesi: Suç işleyen kişi, suçun sonuçları ve ceza hakkında bilgilendirilmelidir. 9. İlgisizlik İlkesi: Kişi, suçu işlemediği sürece cezalandırılamaz. 10. Kanıtlama İlkesi: Suçun işlendiğini kanıtlamak zorunludur.
Ceza hukukunun temel ilkeleri nelerdir?	GPT-3.5 + RAG	Ceza hukukunun temel ilkeleri şunlardır: - Kanunilik ilkesi - Suç ve ceza ilişkisi ilkesi - Masumiyet karinesi - Hukuka uygun ceza ilkesi - İdari yaptırımların ceza yerine kullanılmaması ilkesi - Ceza infazının insan haklarına uygun olması ilkesi

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Ceza hukukunun temel ilkeleri nelerdir?	GPT-4 + RAG	Ceza hukukunun temel ilkeleri, hem birey haklarını koruyan hem de adil yargılamayı güvence altına alan anayasal ve evrensel prensiplerdir. Başlıca ilkeler şunlardır: Kanunilik İlkesi: Suç ve cezaların yalnızca kanunla belirlenebileceğini ifade eder. Kusur İlkesi: Kişi ancak kusurlu fiilinden dolayı sorumlu tutulabilir. Cezaların Şahsiliği İlkesi: Ceza yalnızca suçu işleyen kişiye uygulanır, başkalarına sirayet etmez. Masumiyet Karinesi: Suçluluğu sabit oluncaya kadar herkes masum kabul edilir. Ölçülülük İlkesi: Ceza, işlenen fiilin ağırlığıyla orantılı olmalıdır. İnsan Onurunun Korunması: Ceza ve güvenlik tedbirleri, insan onuruna aykırı şekilde uygulanamaz.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Ceza hukukunun temel ilkeleri nelerdir?	Our System	Ceza hukuku, bireyin hak ve özgürlüklerini yakından ilgilendirdiği için evrensel temel ilkelere dayanır. Bu ilkeler şunlardır: Kanunilik (Suçta ve Cezada Kanunilik) İlkesi: "Kanunsuz suç ve ceza olmaz" (Nullum crimen, nulla poena sine lege) prensibidir. Bir fiilin suç sayılabilmesi ve karşılığında bir ceza verilebilmesi için bunun açıkça kanunda yazılı olması gerekir. Kusur İlkesi: Bir kimsenin işlemiş olduğu fiilden dolayı cezalandırılabilmesi için o fiili kasten veya taksirle, yani kusurlu bir şekilde işlemiş olması gerekir. Ceza Sorumluluğunun Şahsiliği İlkesi: Suç işleyen kişiden başkası o suçtan dolayı sorumlu tutulamaz ve cezalandırılamaz. İnsancıllık İlkesi: Cezaların infazında insan onurunun korunması esastır. İşkence ve zalimane muamele yasaktır. Orantılılık İlkesi: İşlenen suçun ağırlığı ile verilen ceza arasında makul bir oran olmalıdır. Masumiyet Karinesi (Suçsuzluk Karinesi): Suçluluğu kesinleşmiş bir mahkeme kararıyla sabit oluncaya kadar herkes masum sayılır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun'da evlilik şartları nelerdir?	GPT-3.5	1. Ergin olmak: Türk Medeni Kanunu'na göre evlenmek için erkeklerde 18, kadınlarda ise 18 yaşını doldurmuş olmak gerekmektedir. 2. Evlenme ehliyetine sahip olmak: Akıl sağlığı yerinde olmayan, evlenme ehliyetine sahip olmayan veya kısıtlanmış olan kişiler evlenebilir. 3. Rıza: Her iki tarafın da evlenmeye rıza göstermesi gerekmektedir. 4. Yasaklar: Kanun tarafından belirlenen yasaklar nedeniyle evlenme engelinin bulunmaması gerekmektedir. Örneğin, akraba olan kişiler arasında evlilik yasaktır. 5. Resmi nikah: Evlilik, resmi nikah kıyılarak yapılmalıdır. Dinî nikah ya da sadece törenle yapılan evlilik hukuken geçerli değildir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun'da evlilik şartları nelerdir?	GPT-4	1. Evlenecek kişilerin ergin olması (18 yaşını tamamlamış olmak). 2. Zihinsel açıdan evliliğin gereklerini yerine getirebilecek durumda olmak. 3. Daha önceki bir evlilik bağı sona ermiş olmalı. 4. Evlenmek isteyenlerin birbirleriyle yakın akraba olmaması gerekmektedir. 5. Medeni hakları kullanma ehliyetine sahip olmak. 6. Evlilik başvurusu yapabilmek ve evlenme beyannamesi verebilmek gerekir. 7. Evlenmek isteyen kişinin cinsel yönelimine göre kanuni düzenlemeler uygulanır. 8. Zorlama veya aldatma olmamalı. 9. Evlenmek isteyen kişilerin sağlık kurulu raporu alması gerekmektedir. 10. Evlilik iki tarafın rızası ve beyanı olmadan gerçekleşemez.
Medeni Kanun'da evlilik şartları nelerdir?	GPT-3.5 + RAG	Medeni Kanun'a göre evlilik için gerekli şartlar: - Tarafların reşit olması (kadın 18, erkek 18 yaşını doldurmuş olmalı) - Evlenme ehliyetine sahip olmaları (kısıtlı olmamaları) - Evlenme iradelerini serbestçe açıklamaları - Evlenme akdinin kıyılması için gerekli olan rıza - Evlenme akdinin kıyılması için gerekli olan niyet - Evlenme akdinin kıyılması için gerekli olan meşru sebep bulunmaması

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun'da evlilik şartları nelerdir?	GPT-4 + RAG	Türk Medeni Kanunu'na göre evlenmenin geçerli olabilmesi için birtakım şartlar öngörülmüştür. Bunlar olumlu şartlar (evlenebilmek için bulunması gereken koşullar) ve olumsuz şartlar (evlenmeye engel durumlar) şeklinde incelenir: 1. Olumlu Şartlar 1-Yaş Şartı: Erkek ve kadın için evlenme yaşı 17'dir. Haklı sebeplerle hâkim izniyle 16 yaşında da evlenme mümkündür. 2-Ayırt Etme Gücü: Evlenme sırasında kişilerin fiil ehliyetine sahip olması gerekir; ayırt etme gücü olmayanların evliliği geçersizdir. 3-Rıza: Evlenme serbest iradeye dayanmalıdır; baskı, tehdit veya hileyle yapılan evlenmeler iptal edilebilir. 2. Olumsuz Şartlar (Evliliğe Engel Haller) 1-Hısımlık Yasağı: Üstsoy-altsoy, kardeşler ve bazı evlilik bağıyla oluşan hısımlıklar arasında evlenme yasaktır. 2-Mevcut Evlilik: Kişi ancak tek bir evlilik yapabilir; önceki evlilik sona ermeden yeni bir evlilik yapılamaz. 3-Bazı Hastalıklar: Bulaşıcı ağır hastalıklar evlenmeye engel olabilir. 4-Bekleme Süresi: Boşanmış kadınlar için üç yüz günlük bekleme süresi vardır; doğumla veya mahkeme kararıyla kaldırılabilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun'da evlilik şartları nelerdir?	Our System	721 sayılı Türk Medeni Kanunu'na göre evliliğin geçerli olabilmesi için bazı maddi ve şekli şartlar bulunur: Maddi Şartlar (Evlenme Ehliyeti): Yaş: Kişinin evlenebilmesi için belirli bir yaşa ulaşmış olması gerekir (Bir sonraki soruda detaylandırılmıştır). Ayırt Etme Gücü: Evlenecek kişilerin, evliliğin anlam ve sonuçlarını kavrayabilecek ayırt etme gücüne (temyiz kudretine) sahip olmaları gerekir. Akıl hastaları gibi bu güce sahip olmayanlar evlenemez. Evlenme Engellerinin Bulunmaması: Yakın Hısımlık: Üstsoy ile altsoy arasında (anne/baba ile çocuk/torun); kardeşler arasında; amca, dayı, hala ve teyze ile yeğenleri arasında evlilik yasaktır. Mevcut Evlilik: Bir kimse, önceki evliliği sona ermeden yeniden evlenemez (Tek eşlilik ilkesi). Şekli Şartlar (Evlenme Töreni): Evliliğin, yetkili bir evlendirme memuru (belediye başkanı veya görevlendirdiği memur, muhtar vb.) önünde, iki tanığın katılımıyla ve aleni (herkese açık) olarak yapılması zorunludur.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Türk Ceza Kanunu temel ilkeleri nelerdir?	GPT-3.5	1. Adalet Ilkesi: Ceza Kanunu kapsamında suç işleyen kişilerin cezalandırılmalarında adalet ve eşitlik ilkesi ön planda tutulur. 2. İnsan Haklarına Saygı İlkesi: Ceza Kanunu'nun uygulanmasında insan haklarına saygı temel bir ilkedir ve cezalandırma sürecinde insan haklarına zarar verilmemeye özen gösterilir. 3. Kanunilik İlkesi: Ceza Kanunu'nda suç ve cezaların sınırları, somut bir şekilde belirtilir ve suçu işleyen kişiler bu belirlenen sınırlar içinde yargılanır. 4. Hukuk Devleti İlkesi: Ceza Kanunu'nun uygulanmasında hukuk devleti ilkesine uyulur ve herkesin hukuk önünde eşit olduğu ilkesi gözetilir. 5. İnsan Onuruna Saygı İlkesi: Ceza Kanunu'nun amacı suç işleyen kişileri cezalandırmak değil, toplumu korumak ve suçun tekrarlanmasını önlemektir. Bu nedenle insan onuruna saygı ön planda tutulur.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Türk Ceza Kanunu temel ilkeleri nelerdir?	GPT-4	1. Yasallık ilkesi: Hiçbir eylem ceza kapsamına giremez ve hiçbir kişiye ceza verilemez yasa ile önceden belirlenmemişse. 2. Suçluluk ilkesi: Bir kişi, yalnızca suçunun kanıtlanmış olması durumunda cezalandırılabilir. 3. Kişiye özgülük ilkesi: Suç işleyen kişiye ceza verilir, başkalarına değil. 4. Humanizm ilkesi: Cezalar, insan onuruna ve temel insan haklarına aykırı olmamalıdır. 5. İkili ceza ilkesi: Bir kişi, aynı suç için iki kere cezalandırılamaz. 6. Müktesep haklara saygı ilkesi: Bir ceza, kişinin temel haklarını sınırlayamaz veya yok edemez. 7. Cezanın bireyselleştirilmesi ilkesi: Cezalar, her bir suçlunun kişisel durumuna uygun olarak belirlenmelidir.
Türk Ceza Kanunu temel ilkeleri nelerdir?	GPT-3.5 + RAG	Türk Ceza Kanunu'nun temel ilkeleri şunlardır: - Yasama yoluyla belirlenmiş olması - Suç ve ceza ilişkisinin belirlenmesi - Ceza hukukunun genel ilkelerine uygun olması - Hukuka aykırı eylemlere karşı caydırıcılık ilkesine dayanması - İnsan haklarına saygı gösterilmesi - Hukuk devleti ilkesine uygun olması - Ceza kanunlarının geriye yürümezliği ilkesine uygun olması

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Türk Ceza Kanunu temel ilkeleri nelerdir?	GPT-4 + RAG	Türk Ceza Kanunu (TCK), suç ve cezaların adil, ölçülü ve hukuk devleti ilkelerine uygun şekilde uygulanabilmesi için birtakım temel ilkeler ortaya koyar. Bunlar şunlardır: Kanunilik İlkesi: Suç ve cezaların kanunla belirlenmesi gerekir. Kanunsuz suç ve ceza olmaz (nullum crimen, nulla poena sine lege). Kusur İlkesi: Kişi ancak kusurlu olduğu fiilden sorumlu tutulabilir; kusur olmadan ceza verilmez. Şahsilik İlkesi: Ceza yalnızca suçu işleyen kişiye uygulanır, başkasına sirayet etmez. Eşitlik İlkesi: Kanun karşısında herkes eşittir; ceza hukukunda ayrımcılık yapılamaz. Adalet ve Ölçülülük İlkesi: Ceza ve güvenlik tedbirleri işlenen fiilin ağırlığıyla orantılı olmalı; insan onuru zedelenmemelidir. Geriye Yürümezlik: Kural olarak ceza kanunları geçmişe uygulanmaz; ancak failin lehine olan hükümler geçmişe de uygulanabilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Türk Ceza Kanunu temel ilkeleri nelerdir?	Our System	5237 sayılı Türk Ceza Kanunu'nun (TCK) başlangıç maddeleri, ceza hukukunun evrensel ilkelerini yansıtır ve kanunun ruhunu oluşturur. Bu ilkeler şunlardır: Kanunilik İlkesi (Madde 2): Kanunun açıkça suç saymadığı bir fiilden dolayı kimseye ceza verilemez ve güvenlik tedbiri uygulanamaz. Adalet ve Kanun Önünde Eşitlik İlkesi (Madde 3): Suç işleyen kişi hakkında, işlenen fiilin ağırlığıyla orantılı ceza ve güvenlik tedbirine hükmolunur. Kanun uygulamasında kişiler arasında ayrım yapılamaz. Ceza Sorumluluğunun Şahsiliği (Madde 20): Ceza sorumluluğu şahsîdir; kimse başkasının fiilinden dolayı sorumlu tutulamaz. Kusur İlkesi: TCK, suçların manevi unsuru olarak "kast" ve "taksir"i düzenleyerek kusursuz ceza olmayacağını benimsemiştir. Bu ilkeler, genel ceza hukuku ilkelerinin TCK metnine yansımış halidir.
Medeni Kanun evlilik yaşı kaçtır?	GPT-3.5	Medeni Kanun'a göre Türkiye'de kadın ve erkekler için evlilik yaşı 18'dir. Ancak; evliliğin 17 yaşından küçük kişiler arasında yapılması halinde, mahkeme izniyle evlilik gerçekleştirilebilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun evlilik yaşı kaçtır?	GPT-4	1. Türk Medeni Kanunu'na göre evlilik yaşı 18'dir. 2. Olağanüstü durumlarda ve hakim kararıyla 17 yaşında evlenilebilir. 3. Çok özel koşullarda, mahkeme kararıyla 16 yaşında evlenilebilir, fakat cinsel rüşt yaşı 18 olarak kabul edilir. Bu durumda, 16 yaşında evlenebilmek için hakimin, çiftin evlenmelerine izin vermesi ve bu evliliğin çiftlerin üstün yararına olacağını tespit etmesi gerekmektedir.
Medeni Kanun evlilik yaşı kaçtır?	GPT-3.5 + RAG	Türk Medeni Kanunu'na göre evlenme yaşı 17'dir; hâkim kararıyla 16 yaşında da evlenilebilir.
Medeni Kanun evlilik yaşı kaçtır?	GPT-4 + RAG	Türk Medeni Kanunu'na göre kadın ve erkek için evlenme yaşı 17'dir. Ancak, istisnai durumlarda hâkim izniyle 16 yaşını doldurmuş olanlar da evlenebilir; bu sınırın altındaki evlilikler geçersizdir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Medeni Kanun evlilik yaşı kaçtır?	Our System	Türk Medeni Kanunu'nun 124. maddesine göre evlilik yaşı şu şekilde düzenlenmiştir: Olağan Evlenme Yaşı: Erkek veya kadın, 17 yaşını doldurmadıkça evlenemez. Olağanüstü Evlenme Yaşı (Hakim İzniyle): Ancak, hâkim olağanüstü durumlarda ve pek önemli bir sebep varsa, 16 yaşını doldurmuş olan erkek veya kadının evlenmesine izin verebilir. Bu durumda mahkeme kararından önce anne ve babanın veya vasinin dinlenmesi zorunludur. Özetle, normal şartlarda evlilik için 17 yaşın doldurulması gerekirken, istisnai durumlarda ve mahkeme kararıyla bu yaş 16'ya inebilmektedir.
6102 sayılı Türk Ticaret Kanunu 4. maddesi neyi düzenler?	GPT-3.5	6102 sayılı Türk Ticaret Kanunu'nun 4. maddesi şu konuları düzenler: 1. Şirketin yerleşim yeri, 2. Şirketin tescil edilme zorunluluğu, 3. Ticaret siciline tescil edilmesi gereken belgeler, 4. Tescilin sicil müdürlüğünce yapılması ve sicil kayıtlarının kesin hükmünde olmasının sağlanması.
6102 sayılı Türk Ticaret Kanunu 4. maddesi neyi düzenler?	GPT-4	6102 sayılı Türk Ticaret Kanunu 4. maddesi, "Kanunla sınırlama veya yasak getirilmeyen herkesin ticaret yapma hakkı"nı düzenler.
6102 sayılı Türk Ticaret Kanunu 4. maddesi neyi düzenler?	GPT-3.5 + RAG	6102 sayılı Türk Ticaret Kanunu'nun 4. maddesi, "Ticaret unvanı"nı düzenler.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
6102 sayılı Türk Ticaret Kanunu 4. maddesi neyi düzenler?	GPT-4 + RAG	Türk Ticaret Kanunu'nun (TTK) 4. maddesi, hangi davaların ticari dava sayılacağını düzenler. Buna göre: 1-Mutlak Ticari Davalar: Kanunda açıkça ticari sayılan hususlardan doğan davalar (örneğin, ticari işletme ile ilgili bazı sözleşmeler). 2-Nispi Ticari Davalar: Her iki tarafın da tacir olduğu ve uyuşmazlığın ticari işletmeleriyle ilgili bulunduğu davalar. 3-Ayrıca, kanunda düzenlenmemiş olsa bile tarafların ticari iş karinesi çerçevesinde ortaya çıkan uyuşmazlıkları da ticari dava kapsamında değerlendirilir. Bu madde, ticaret mahkemelerinin görev alanını belirlemesi bakımından önemlidir. Çünkü bir uyuşmazlığın "ticari dava" sayılıp sayılmayacağı, davanın hangi mahkemede görüleceğini (asliye ticaret mahkemesi mi, yoksa genel mahkeme mi) doğrudan etkiler.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
6102 sayılı Türk Ticaret Kanunu 4. maddesi neyi düzenler?	Our System	6102 sayılı Türk Ticaret Kanunu'nun (TTK) 4. maddesi, "Ticari İşler" kavramının kapsamını belirler ve hangi işlerin ticari sayılacağını düzenler. Bu maddenin temel amacı, bir davanın ticari dava olup olmadığını ve dolayısıyla hangi mahkemenin (Asliye Ticaret Mahkemesi mi, Asliye Hukuk Mahkemesi mi) görevli olduğunu saptamaktır. Madde özetle şunu belirtir: Bir tacirin (ticari işletme sahibinin) borçlarının ticari olması esastır. Ancak, bir tacirin yaptığı işlemin ticari sayılmaması için iki durumdan biri mevcut olmalıdır: Gerçek kişi olan tacirin, işlemi yaptığı anda bunun ticari işletmesiyle ilgili olmadığını karşı tarafa açıkça bildirmesi. İşlemin niteliği gereği ticari sayılamayacak olması (örneğin bir tacirin ailesi için ev kiralaması gibi). Taraflardan sadece biri için ticari nitelikte olan sözleşmeler, kanunda aksine hüküm bulunmadıkça, diğeri için de ticari sayılır. Bu madde, ticari hayatın gerektirdiği hızlı ve güvenli işleyişi sağlamak adına, bir tacirin yaptığı hemen her işin ticari işletmesiyle ilgili olduğu varsayımını (karinesini) getirir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
4857 sayılı İş Kanunu fazla mesai ücreti hesaplamasıyla ilgili bilgi ver.	GPT-3.5	1. İş Kanunu'na göre, haftalık çalışma süresi 45 saattir. Bu süreyi aşan çalışmalarda fazla mesai ücreti ödenmesi gerekmektedir. 2. Fazla mesai ücreti, normal çalışma saatinin yüzde 50'si üzerinden hesaplanır. 3. Örneğin, normal çalışma saatinin saatlik ücreti 10 TL ise, fazla mesai saati için ödenecek ücret saatlik 15 TL olacaktır. 4. Fazla mesai ücretleri çalışanların resmi tatil günlerinde yapılan çalışmalarda yüzde 100 artarak uygulanır. 5. İşveren fazla mesai ücretini düzenli olarak ödemekle yükümlüdür. Aksi takdirde işveren yasal yaptırımlarla karşılaşabilir. 6. İş Kanunu'nda belirtilen kurallara göre çalışanlar, fazla mesai yaptıklarında hak ettikleri ücreti almalıdır.
4857 sayılı İş Kanunu fazla mesai ücreti hesaplamasıyla ilgili bilgi ver.	GPT-4	1. İş Kanunu'na göre, normal çalışma saatleri dışında kalan süreler fazla mesai olarak kabul edilir. 2. Buna göre, haftalık çalışma süresi 45 saatten fazla olanlar fazla mesai yapan olarak kabul edilir. 3. Fazla mesai yapıldığında, her saat için normal ücretin %50 fazlası ödenir. 4. Fazla mesai saatleri bir yılda toplam 270 saati geçemez. 5. Fazla mesai karşılığı alınacak ücret, fazla mesainin yapıldığı döneme ait ücret ödeme dönemi içinde ödenir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
4857 sayılı İş Kanunu fazla mesai ücreti hesaplamasıyla ilgili bilgi ver.	GPT-3.5 + RAG	1. Fazla mesai ücreti, haftalık çalışma süresini aşan çalışma saatleri için ödenir. 2. Fazla mesai ücreti, normal çalışma saati ücretinin en az %50 fazlası olarak hesaplanır. 3. Fazla mesai ücreti, resmi tatil günlerinde çalışılan saatler için normal çalışma saati ücretinin en az %100 fazlası olarak ödenir. 4. Fazla mesai ücreti, gece çalışılan saatler için normal çalışma saati ücretinin en az %50 fazlası olarak hesaplanır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
4857 sayılı İş Kanunu fazla mesai ücreti hesaplamasıyla ilgili bilgi ver.	GPT-4.5 + RAG	4857 sayılı İş Kanunu'na göre haftalık normal çalışma süresi 45 saattir. Bu sürenin üzerindeki çalışmalar fazla çalışma (fazla mesai) olarak kabul edilir. Fazla mesai ücreti şu şekilde hesaplanır: 1-Normal Saatlik Ücret: Öncelikle işçinin aylık brüt ücreti üzerinden saatlik ücreti bulunur. Saatlik ücret = Aylık brüt ücret ÷ (30 gün × günlük 7,5 saat) 2-Fazla Mesai Ücreti: Her bir fazla mesai saati için, normal saatlik ücretin %50 zamlı tutarı ödenir.Fazla mesai ücreti = Saatlik ücret × 1,5 3-Hafta Tatili, Ulusal Bayram ve Genel Tatillerde Çalışma: Bu günlerde yapılan çalışmalar ayrıca düzenlenmiştir. Örneğin, ulusal bayram ve genel tatil günlerinde çalışma halinde bir günlük ücrete ek olarak ayrıca bir günlük ücret ödenir. 4-Fazla Mesai Sınırı: İş Kanunu'na göre fazla mesai süresi yılda toplam 270 saati geçemez.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
4857 sayılı İş Kanunu fazla mesai ücreti hesaplamasıyla ilgili bilgi ver.	Our System	4857 sayılı İş Kanunu'nun 41. maddesi uyarınca fazla mesai (fazla çalışma), kanunda belirtilen haftalık normal çalışma süresi olan 45 saati aşan çalışmalardır. Fazla mesai ücreti şu şekilde hesaplanır: Ücret Oranı: Fazla çalışılan her bir saat için verilecek ücret, normal çalışma ücretinin saat başına düşen miktarının yüzde elli (%50) yükseltilmesiyle ödenir. Yani, fazla mesai yapılan her saat için 1,5 kat ücret ödenir. Hesaplama Adımları: Aylık Brüt Ücret: Çalışanın aylık brüt maaşı belirlenir. Aylık Çalışma Saati: İş Kanunu'na göre bir ay, 225 çalışma saati olarak kabul edilir. Saatlik Normal Ücret: Aylık Brüt Ücret / 225 = Saatlik Normal Ücret Saatlik Fazla Mesai Ücreti: Saatlik Normal Ücret x 1,5 = Saatlik Fazla Mesai Ücreti Toplam Fazla Mesai Alacağı: Saatlik Fazla Mesai Ücreti x Toplam Fazla Mesai Saati = Aylık Fazla Mesai Alacağı Örnek: Aylık brüt maaşı 45.000 TL olan bir işçinin saatlik normal ücreti 45.000 / 225 = 200 TL'dir. Bu işçi bir saat fazla mesai yaptığında alacağı ücret 200 x 1,5 = 300 TL olacaktır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
5237 sayılı TCK 765. madde hakkında bilgi ver.	GPT-3.5	5237 sayılı Türk Ceza Kanunu'nun 765. maddesi "İntihara Teşebbüs" başlığını taşır ve intihar girişimini suç olarak tanımlar. Bu maddeye göre: 1. Bir kişinin intihar etmeye teşebbüs etmesi suçtur. 2. İntihara teşebbüs halinde, kişiye 1 yıldan 3 yıla kadar hapis cezası verilebilir. 3. İntihara teşebbüs eden kişiye hapis cezası verilmesi halinde, kişinin tedavi edilmesi için gerekli önlemlerin de alınması gereklidir. 4. İntihara teşebbüs, kişinin canını kendi eliyle alma eylemi olarak değerlendirilir ve toplumda önleyici etki yaratılması amaçlanır. 5. İntihara teşebbüs davalarında genellikle kişinin psikolojik durumu ve sebep olduğu etkiler de dikkate alınır. Maddeye konu olan intihara teşebbüs, kişinin hayatına son verme niyeti taşımasından kaynaklanır ve ciddi bir suç olarak değerlendirilir. Bu nedenle intihar düşüncesi olan kişilerin yardım alması ve destek görmesi önemlidir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
5237 sayılı TCK 765. madde hakkında bilgi ver	GPT-4	Üzgünüm ancak istediğiniz TCK 765. maddenin içeriği mevcut değil. Türk Ceza Kanunu içerisinde bu madde numarası geçmemektedir. Kanun, 5237 numarası altında 1'den 194'e kadar maddeleri içermektedir. Kontrol etmek istediğiniz kanunun numarasını veya içeriğini tekrardan gözden geçirmenizi öneririm.
5237 sayılı TCK 765. madde hakkında bilgi ver.	GPT-3.5 + RAG	5237 sayılı Türk Ceza Kanunu'nun 765. maddesi, "Suç işlemek amacıyla örgüt kurma" suçunu düzenlemektedir. Bu maddeye göre, suç işlemek amacıyla örgüt kuran kişi veya kişiler hakkında cezai yaptırımlar öngörülmektedir.
5237 sayılı TCK 765. madde hakkında bilgi ver.	GPT-4 + RAG	5237 sayılı Türk Ceza Kanunu (TCK), 1 Haziran 2005 tarihinde yürürlüğe girmiştir ve 1926 tarihli 765 sayılı eski TCK'nın yerine geçmiştir. Dolayısıyla, 5237 sayılı TCK'da 765. madde yoktur çünkü yeni kanun toplam 345 maddeden oluşmaktadır. "765. madde" denildiğinde, aslında yürürlükten kaldırılan 765 sayılı eski Türk Ceza Kanunu'nun kendi içindeki maddelerinden biri kastediliyor olabilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
5237 sayılı TCK 765. madde hakkında bilgi ver.	Our System	Bu soruda bir karışıklık bulunmaktadır. 5237 sayılı Türk Ceza Kanunu (TCK), 1 Haziran 2005 tarihinde yürürlüğe giren ve halen yürürlükte olan kanundur. Bu kanun toplam 345 maddeden oluşmaktadır ve dolayısıyla 765. maddesi yoktur. Muhtemelen kastettiğiniz kanun, 2005 yılında yürürlükten kaldırılan eski 765 sayılı Türk Ceza Kanunu'dur. O kanunda maddeler çok daha yüksek sayılara ulaşıyordu. Dolayısıyla, mevcut ve geçerli olan 5237 sayılı TCK'da 765 numaralı bir madde bulunmamaktadır. Hukuki bir işlem veya araştırma yaparken güncel kanun olan 5237 sayılı TCK'yı esas almak gerekir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Yargıtay İş Dairesi kararları hakkında bilgi ver.	GPT-3.5	1. Yargıtay İş Dairesi, Türkiye Cumhuriyeti'nin en yüksek yargı mercii olan Yargıtay'a bağlı olarak faaliyet gösteren bir dairedir. 2. İş Dairesi, iş hukuku alanında çıkan davalara ilişkin temyiz başvurularını görüşür ve karara bağlar. 3. Temyiz incelemesi sonucunda Yargıtay İş Dairesi, hükümlerin hukuka uygunluğunu denetler ve gerekli gördüğü hallerde karar verir. 4. Yargıtay İş Dairesi kararları, yargılama süreci sonunda kesinleşerek içtihat oluşturur ve diğer mahkemeler için bağlayıcı bir nitelik taşır. 5. İş Dairesi'nin verdiği kararlar, iş hukuku alanında uygulamada referans olarak kullanılır ve kararlarında hukuki prensipleri belirleyici bir rol oynar.
Yargıtay İş Dairesi kararları hakkında bilgi ver.	GPT-4	1. Yargıtay İş Dairesi, iş davalarına bakar. 2. Davalar genellikle iş sözleşmelerinin feshi, tazminat, işe iade gibi konuları içerir. 3. İş Dairesi kararları, tüm Türkiye'de iş mahkemeleri için emsal teşkil eder. 4. Yargıtay İş Dairesi'nin kararlarını değiştirebilecek bir yargı mercii yoktur. 5. Daire, 1967'den itibaren hizmet verir. 6. Kararlar, Yargıtay Başkanlığı'nın resmi internet sitesinde yayınlanır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Yargıtay İş Dairesi kararları hakkında bilgi ver.	GPT-3.5 + RAG	1. Yargıtay İş Dairesi kararları, 6. Daire tarafından verilmiştir. 2. İstanbul İli, Kartal İlçesi'nde bir taşınmaza ilişkin imar planlarına yapılan itirazların zımnen reddedilmesi üzerine açılan davalara ilişkin kararlar içermektedir. 3. İlgili kararlar, 13.06.2018 ve 29.06.2018 tarihlerinde açılan davalara ilişkin olup, 6. Daire tarafından verilmiştir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Yargıtay İş Dairesi kararları hakkında bilgi ver.	GPT-4 + RAG	Yargıtay'ın daireleri arasında yer alan İş Daireleri, iş hukuku alanındaki uyuşmazlıkların temyiz incelemesini yapan dairelerdir. Bu kararlar, özellikle 4857 sayılı İş Kanunu, 6098 sayılı Türk Borçlar Kanunu'nun iş ilişkilerine dair hükümleri, 1475 sayılı mülga İş Kanunu'nun kıdem tazminatına dair 14. maddesi, toplu iş sözleşmeleri ve sendikal haklar gibi konularda içtihat oluşturur. 1. İnceleme Konuları 1-Kıdem ve ihbar tazminatı davaları 2-Fazla mesai, hafta tatili, ulusal bayram ve genel tatil ücretleri 3-İş kazaları ve meslek hastalıkları nedeniyle tazminat talepleri 4-Sendikal tazminat ve işe iade davaları 2. Kararların Niteliği 1-Yargıtay İş Dairesi kararları, iş mahkemelerinden gelen dosyaları hukuka uygunluk açısından inceler. 2-Bu kararlar, hem uygulamayı yönlendiren emsal nitelik taşır hem de iş hukukunun dinamik yapısına uygun olarak sürekli gelişir. 3-Çoğu zaman işçi-işveren ilişkilerinde belirsizlikleri gideren ve alt mahkemeler için yol gösterici olan içtihat birliği sağlar.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Yargıtay İş Dairesi kararları hakkında bilgi ver.	Our System	Yargıtay, Türkiye'deki adli yargı alanında en yüksek temyiz (kanun yolu) merciidir. İş mahkemeleri tarafından verilen kararların son incelemesi Yargıtay'ın ilgili İş Daireleri tarafından yapılır (Örn: Yargıtay 9. Hukuk Dairesi, 22. Hukuk Dairesi vb.). Yargıtay İş Dairelerinin kararları şu açılardan önemlidir: İçtihat Oluşturma: Kanun metinlerinin somut olaylara nasıl uygulanacağını gösterirler. Belirsiz veya yoruma açık konularda verdikleri kararlar, alt derecedeki iş mahkemeleri için yol gösterici niteliktedir. Bu kararlara içtihat denir. Hukuk Birliği Sağlama: Ülke genelindeki farklı iş mahkemelerinin benzer konularda farklı kararlar vermesini önleyerek hukuk uygulamasında birliği sağlarlar. Kanun Yolu İncelemesi: İş mahkemesi kararının usul ve yasaya uygun olup olmadığını denetlerler. Kararı hukuka uygun bulurlarsa onama, hukuka aykırı bulurlarsa bozma kararı verirler.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Danıştay kamulaştırma kararlarından bahset.	GPT-3.5	1. Danıştay, kamu idarelerinin yaptığı kamulaştırma işlemlerinin yasalara uygunluğunu denetleyen bir yargı kurumudur. 2. Kamulaştırma kararları genellikle ilgili idarenin kararı üzerine yapılır ve kamulaştırma bedeli belirlenir. 3. Kamulaştırma kararlarına karşı idari yargıda dava açılabilir ve bu davalarda Danıştay yetkilidir. 4. Danıştay, kamulaştırma kararlarını inceleyerek idarenin hukuka uygun hareket edip etmediğini değerlendirir ve kararına göre sonuçlandırır. 5. Kamulaştırma kararlarına karşı kişilerin yasal haklarını korumak için Danıştay'a başvurma hakkı bulunmaktadır. Her iki taraf da Danıştay kararlarına uymak zorundadır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Danıştay kamulaştırma kararlarından bahset.	GPT-4	1. Danıştay, kamulaştırma kararları konusunda son derece önemli bir role sahiptir. 2. Kamulaştırma, devletin veya diğer kamu tüzel kişilerinin, özel mülkiyet hakları üzerinde kamunun yararı amacıyla yapılan bir müdahaledir. 3. Kamulaştırma kararları, Danıştay'ın İdari Dava Daireleri tarafından verilir. 4. Danıştay'ın verdiği kararlar, idare hukuku açısından emsal teşkil eder. 5. Kamulaştırma kararlarına itiraz edildiği durumlarda, Danıştay bu itirazları değerlendirir ve son kararı verir. 6. Danıştay, kamulaştırma kararlarındaki haksızlıkları, hukuka aykırılıkları ve usul hatalarını ele alır. 7. Kamulaştırma kararlarına itiraz süreci, kamulaştırmanın hukuka uygunluğunu değerlendirmek için kullanılır. 8. Kamulaştırma kararının hukuka uygun olup olmadığını belirleyen Danıştay'ın kararı kesindir ve bağlayıcıdır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Danıştay kamulaştırma kararlarından bahset.	GPT-3.5 + RAG	Danıştay'ın kamulaştırma kararları, idarenin kamu yararı gerekçesiyle özel mülkiyete müdahalesini denetler. Bu kararlarda genellikle şu konular öne çıkar: Kamu yararı kararının hukuka uygunluğu Bedel tespitinin adil olup olmadığı Usul hataları (tebligat, bilirkişi incelemesi, mahkeme kararı olmadan el koyma) Mülkiyet hakkının Anayasa ve AİHS çerçevesinde korunması

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Danıştay kamulaştırma kararlarından bahset.	GPT-4 + RAG	Danıştay'ın kamulaştırma kararları, idarenin kamu yararı gerekçesiyle özel mülkiyete müdahalesini denetleyen en önemli içtihatlardandır. Anayasa'nın 46. maddesi ve 2942 sayılı Kamulaştırma Kanunu çerçevesinde verilen bu kararlar üç ana eksende toplanır: 1-Kamu Yararı Denetimi: Kamulaştırma kararlarının geçerli olabilmesi için kamu yararının somut ve makul gerekçelere dayanması gerekir. Danıştay, şekli bir "kamu yararı" kararını yeterli görmez. 2-Mülkiyet Hakkının Korunması: Kamulaştırma, mülkiyet hakkına ciddi bir müdahale olduğundan ölçülülük ilkesi gereği sıkı denetime tabidir. 3-Bedel Tespiti ve Usul: Kamulaştırma bedelinin gerçek ve güncel olması, maliklere adil şekilde ödenmesi ve tüm sürecin tebligat, bilirkişi incelemesi gibi usul kurallarına uygun yürütülmesi zorunludur. Danıştay'ın özellikle "fiili el atma" davalarındaki kararları, idarenin mahkeme kararı olmadan taşınmaza el koymasını hukuka aykırı bulmakta ve tazminat ödenmesine hükmetmektedir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Danıştay kamulaştırma kararlarından bahset.	Our System	Danıştay, Türkiye'deki idari yargının en yüksek merciidir. Kamulaştırma, idarenin (devletin) kamu yararı amacıyla özel mülkiyetteki bir taşınmaza bedelini peşin ödeyerek el koyması işlemidir ve bu bir idari işlemdir. Danıştay'ın kamulaştırma ile ilgili rolü şöyledir: İdari İşlemin Hukukiliğini Denetleme: Danıştay, kamulaştırma işleminin kendisine karşı açılan davalarda, işlemin hukuka uygunluğunu denetler. İncelediği temel unsurlar şunlardır: Kamulaştırmanın gerçekten geçerli bir kamu yararı amacına dayanıp dayanmadığı. İdarenin kamulaştırma sürecinde kanunda öngörülen adımları (örneğin yeterli araştırma yapma, kamu yararı kararı alma gibi) doğru izleyip izlemediği. Kamulaştırmayı yapan idarenin yetkili olup olmadığı ve işlemin hukuka uygun bir sebebe dayanıp dayanmadığı. Önemli bir ayrım olarak, kamulaştırma bedelinin tespiti ve tescili ile ilgili davalar adli yargıda, yani Asliye Hukuk Mahkemelerinde görülür. Danıştay, bedelin miktarına değil, kamulaştırma kararının bir idari işlem olarak hukuka uygun olup olmadığına bakar. Eğer Danıştay, bir kamulaştırma işlemini hukuka aykırı bulursa iptal kararı verir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
AYM temel hak ihlali kararlarından bahset.	GPT-3.5	1. Anayasa Mahkemesi, temel hak ihlali kararı verme yetkisine sahiptir. 2. Temel hak ihlali kararı, bireylerin veya kurumların temel hak ve özgürlüklerinin ihlal edildiğini tespit ettiği durumlarda verilir. 3. Temel hak ihlali kararı, ilgili kanun veya idari işlemler ile ilgili olabilir. 4. Anayasa Mahkemesi, temel hak ihlali kararı verdiğinde, ihlalin giderilmesi için gereken düzenlemeleri yapma yetkisine sahiptir. 5. Temel hak ihlali kararları, genel olarak kamuoyunda büyük yankı uyandırır ve hükümetin politikalarını da etkileyebilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
AYM temel hak ihlali kararlarından bahset.	GPT-4	1. AYM, Anayasa Mahkemesi, Türkiye'de temel hak ve özgürlüklerin korunması için son başvuru mekanizmasıdır. 2. İhlal kararları, AYM'nin bir kişi veya tüzel kişilik hakkındaki bir durumu değerlendirip, anayasaya uygun olmadığına karar verdiği durumlardır. 3. Bu kararlar genellikle, kişinin hak ve özgürlüklerinin ihlal edildiği durumlarla alakalıdır. 4. Temel hak ihlalleri, kişinin yaşam hakkı, işkence yasağı, kişi hürriyeti ve güvenliği, mahkemeye başvurma hakkı gibi haklarını kapsar. 5. AYM, ihlal kararı aldığında, hükümetin veya başka bir organın bu kararı dikkate alarak hareket etmesi ve durumu düzeltmesi beklenir. 6. Kararlar genellikle, başvuruda bulunan kişiye tazminat ödenmesi, başvurucunun haklarının iadesi veya ihlalin bir daha tekrarlanmaması şeklinde sonuçlar doğurur. 7. AYM'nin bu kararları doğrultusunda, ilgili kurum veya kuruluşların gereken düzeltmeleri yapmaları, ülkenin hukuk devleti ilkesine uygun bir şekilde hareket etmesi önemlidir.
AYM temel hak ihlali kararlarından bahset.	GPT-3.5 + RAG	Anayasa Mahkemesi'nin temel hak ihlali kararlarında genellikle adil yargılanma hakkı, mülkiyet hakkı, ifade ve basın özgürlüğü, kişi özgürlüğü ve güvenliği ile özel hayatın gizliliği gibi konularda ihlaller tespit edilmektedir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
AYM temel hak ihlali kararlarından bahset.	GPT-4 + RAG	Anayasa Mahkemesi (AYM), bireysel başvurular üzerine verdiği temel hak ihlali kararlarında özellikle adil yargılanma hakkının ihlali, mülkiyet hakkının korunmaması, ifade ve basın özgürlüğüne yönelik kısıtlamalar, hukuka aykırı gözaltı ve tutuklamalar ile özel hayatın gizliliğinin ihlali gibi konularda önemli içtihatlar geliştirmektedir; bu kararlar hem bireylerin haklarının korunmasına katkı sağlamakta hem de kamu otoritelerine temel haklara saygı yükümlülüğü getirmektedir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
AYM temel hak ihlali kararlarından bahset.	Our System	Anayasa Mahkemesi (AYM), bireysel başvuru yoluyla, kamu gücü tarafından temel hak ve özgürlükleri ihlal edilen kişilerin başvurularını inceleyen yargı organıdır. Kişiler, bir hak ihlaline uğradıklarında öncelikle diğer tüm idari ve hukuki yolları (mahkeme süreçlerini) tüketmek zorundadır. Bu yollar tüketildikten sonra hala haklarının ihlal edildiğini düşünüyorlarsa 30 gün içinde AYM'ye bireysel başvuruda bulunabilirler. AYM, Anayasa'da ve Avrupa İnsan Hakları Sözleşmesi'nde güvence altına alınan temel hakların ihlal edilip edilmediğini denetler. En sık karşılaşılan başvuru konuları şunlardır: Yargılamanın makul sürede bitmemesi, mahkemeye erişim hakkının engellenmesi, gerekçesiz karar verilmesi gibi ihlaller. Düşünce ve kanaatleri açıklama hürriyetine yönelik müdahaleler. Kamulaştırmasız el atma, mülkün kullanımının haksız engellenmesi gibi durumlar. Hukuka aykırı tutuklamalar. AYM bir hak ihlali tespit ederse, ihlalin sonuçlarının ortadan kaldırılması için ne yapılması gerektiğini belirtir. Bu, yeniden yargılama yapılması, tazminat ödenmesi veya ihlali giderecek başka bir tedbir alınması şeklinde olabilir. Kararları kesindir ve tüm devlet organlarını bağlar.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş kazası tazminatı nasıl hesaplanır?	GPT-3.5	İş kazası tazminatı hesaplanırken genellikle aşağıdaki unsurlar dikkate alınır: 1.Maddi Zararlar: Tedavi masrafları.Geçici iş göremezlik süresince alınacak gelir kaybı.Kalıcı iş göremezlik durumunda alınacak gelir kaybı. Maluliyet durumunda oluşacak ekstra masraflar 2-Manevi Zararlar: Acı ve ıstırap tazminatı. Geçmiş ve gelecek yaşam kalitesinde oluşabilecek düşüş nedeniyle tazminat 3-Diğer Unsurlar: İşverenin ihmali veya kusuru nedeniyle ödenecek cezai tazminat. Maddi ve manevi zararlar için hesaplanan faiz 4-Hesaplama Süreci:Tazminat hesaplaması genellikle uzman bir avukat veya hukuk firması aracılığıyla yapılır. Çalışanın kazaya maruz kalma durumu, meydana gelen zararın boyutu ve işverenin yaptığı hatalar göz önünde bulundurulur. Hesaplama yapılırken, yasal düzenlemeler ve içtihatlar da dikkate alınır. 5-Davalar: Eğer işverenle anlaşmaya varılamazsa, dava yoluyla iş kazası tazminatı talep edilebilir. Mahkeme kararı ile tazminat miktarı belirlenebilir. Son olarak, iş kazası tazminatı hesaplaması oldukça karmaşık bir süreç olabilir. Bu nedenle, bir hukuk danışmanına başvurmanız ve konuyla ilgili profesyonel destek almanız önemlidir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş kazası tazminatı nasıl hesaplanır?	GPT-4	1. İlk olarak iş kazasına uğrayan kişinin geliri tespit edilir. 2. İkinci aşamada gelir, genel sağlık sigortası aylık tavan sınırına çıkarılır. 3. Ardından iş göremezlik oranı belirlenir. 4. En son aşamada ise bu değerler birleştirilir ve iş kazası tazminatı miktarı hesaplanır. Bu hesaplama genel bir bilgidir, davanın türüne ve durumuna göre çeşitlilik gösterebilir. Kesin ve net bir hesaplama için bir hukuk danışmanıyla görüşme yapılabilir.
İş kazası tazminatı nasıl hesaplanır?	GPT-3.5 + RAG	İş kazası tazminatı, iş kazasına uğrayan işçiye veya işçinin yakınlarına, iş kazasından kaynaklanan maddi ve manevi zararların karşılanması amacıyla ödenen bir tazminattır. İş kazası tazminatı hesaplanırken genel olarak aşağıdaki unsurlar dikkate alınır: 1. İş kazasının neden olduğu maddi zararlar (tedavi masrafları, gelir kaybı vb.) 2. İş kazasının neden olduğu manevi zararlar (acı, ıstırap, moral bozukluğu vb.) 3. İş kazasının meydana geldiği tarih 4. İşçinin aldığı günlük ücret 5. İşçinin kusur oranı 6. İşverenin kusur oranı 7. İşçinin mesleki durumu 8. İşçinin yaşam standartları Bu unsurlar göz önünde bulundurularak iş kazası tazminatı hesaplanır ve ödenir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş kazası tazminatı nasıl hesaplanır?	GPT-4 + RAG	İş kazası tazminatı, işçinin uğradığı zararın niteliğine göre farklı yöntemlerle hesaplanır. Genel olarak işçinin yaşı, kazadan önceki gelir düzeyi, meslekte kazanma gücü kaybı oranı ve çalışma süresi dikkate alınır. Eğer işçi hayatını kaybetmişse, destekten yoksun kalan yakınlarının yaşları ve muhtemel yaşam süreleri de hesaba katılır. Hesaplama, genellikle ilgili tablolar ve Yargıtay içtihatları çerçevesinde yapılır; işçinin aylık net geliri belirlenir, bu gelirden sosyal güvenlik ödemeleri düşülür ve kalan miktar üzerinden tazminat miktarı hesaplanır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İş kazası tazminatı nasıl hesaplanır?	Our System	İş kazası sonucu doğan tazminatların hesaplanması karmaşık bir süreçtir ve genellikle mahkemeler tarafından aktüer (hesap uzmanı) bilirkişiler aracılığıyla yapılır. Tazminat, temel olarak ikiye ayrılır: 1. Maddi Tazminat: Kazazedenin veya vefatı halinde hak sahiplerinin uğradığı somut ekonomik kayıpları karşılar. Başlıca kalemleri şunlardır:Tedavi gierleri, geçici İş Göremezlik Tazminatı Sürekli İş Göremezlik (Maluliyet) Tazminatı, Destekten Yoksun Kalma Tazminatı 2. Manevi Tazminat: Bu tazminatın net bir hesaplama formülü yoktur. Amacı, iş kazası nedeniyle duyulan fiziksel acı, elem, keder ve yaşama sevincindeki azalmayı bir nebze de olsa telafi etmektir. Hâkim; olayın ağırlığını, tarafların sosyal ve ekonomik durumunu, maluliyet oranını ve kusur durumunu göz önünde bulundurarak hakkaniyete uygun bir miktar takdir eder. Hesaplamalar, birçok teknik detay ve yasal parametre içerdiğinden mutlaka bir hukuk profesyonelinden destek alınmasını gerektirir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Kamulaştırma bedeli belirleme kriterleri nelerdir?	GPT-3.5	Kamulaştırma bedeli belirleme kriterleri şunlardır: 1. Taşınmazın bulunduğu bölgenin konumu ve değeri. 2. Taşınmazın yasal durumu ve mülkiyeti. 3. Taşınmazın yüzölçümü ve imar durumu. 4. Taşınmazın kullanım amacı ve potansiyeli. 5. Kamulaştırmanın yapılma nedeni ve aciliyet derecesi. 6. Taşınmazın ekonomik değeri ve çevresel etkileri. 7. Uzmanların raporları ve değerlendirmeleri. 8. Hukuki mevzuat ve yönetmeliklerde belirtilen kriterler.
Kamulaştırma bedeli belirleme kriterleri nelerdir?	GPT-4	1. Taşınmazın bulunduğu lokasyon 2. Taşınmazın m^2 büyüklüğü 3. Taşınmazın üzerindeki yapı veya tesisler 4. Taşınmazın kullanım amacı 5. Taşınmaza talebin yoğunluk durumu 6. Taşınmazın bir tarih öncesinde değeri 7. Taşınmazı etkileyen hukuki ve fiili durumlar 8. Taşınmazın kamulaştırma amacı 9. Taşınmazın piyasa değeri 10. Uzman bir değerleme kuruluşu tarafından düzenlenen değerleme raporu.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Kamulaştırma bedeli belirleme kriterleri nelerdir?	GPT-3.5 + RAG	1. Kamulaştırma bedeli belirleme kriterleri, kamulaştırma yapılacak taşınmazın değerini tespit etmek amacıyla kullanılan yöntemlerdir. 2. Kamulaştırma bedeli belirleme kriterleri arasında taşınmazın bulunduğu bölgedeki emlak piyasası değerleri, taşınmazın niteliği, büyüklüğü, ulaşım imkanları gibi faktörler önemli rol oynar. 3. Kamulaştırma bedeli belirleme kriterleri genellikle yasal düzenlemelerde belirtilir ve uzmanlar tarafından bu kriterlere göre değerleme yapılır.
Kamulaştırma bedeli belirleme kriterleri nelerdir?	GPT-4 + RAG	Kamulaştırma bedeli, Anayasa'nın 46. maddesi ve 2942 sayılı Kamulaştırma Kanunu uyarınca kamu yararı amacıyla taşınmazın gerçek değeri üzerinden belirlenir. Bu bedelin tespitinde taşınmazın niteliği ve kullanım amacı, bölgedeki emsal satış değerleri, tarım arazileri için ürün verimi ve yıllık gelir üzerinden hesaplanan kapitalizasyon değeri, taşınmaz üzerindeki bina, tesis veya ağaçların bedeli ile tapu kayıtları, imar planları ve bilirkişi incelemeleri dikkate alınır. Amaç, mülkiyet hakkını koruyacak şekilde malik için adil ve gerçek bir değer belirlemek, böylece kamulaştırmadan doğan zararı tam olarak karşılamaktır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Kamulaştırma bedeli belirleme kriterleri nelerdir?	Our System	Kamulaştırma bedelinin tespiti, bir idari işlem değil, adli bir süreçtir ve Asliye Hukuk Mahkemeleri tarafından yürütülür. Mahkeme, bedeli belirlemesi için bir bilirkişi heyeti atar. Bu heyetin ve mahkemenin, 2942 sayılı Kamulaştırma Kanunu'nun 11. maddesinde belirtilen kriterlere göre değerlendirme yapması zorunludur. Kamulaştırma bedelinin tespitindeki temel kriterler şunlardır: Taşınmazın Cinsi ve Nevi: Arazinin tarım arazisi mi, arsa mı olduğu; üzerindeki yapının konut mu, ticari işletme mi olduğu gibi özellikler. Yüzölçümü: Taşınmazın büyüklüğü. Emlak Vergisi Değeri: Kamulaştırma tarihindeki en son emlak vergisi beyanı. Yapılar İçin Resmi Birim Fiyatları: Çevre, Şehircilik ve İklim Değişikliği Bakanlığı'nın her yıl yayımladığı yapı yaklaşık birim maliyetleri ve bu maliyetten yıpranma payının düşülmesi. Arsalar İçin Emsal Satış Değeri: Kamulaştırılan arsaya benzer nitelikteki arsaların, kamulaştırma tarihinden önceki özel amaç taşımayan (serbest) satış bedelleri karşılaştırılır. Bu, en önemli kriterlerden biridir. Mahkeme, bu kriterlerin tamamını kullanarak ve somut olaya uygulayarak taşınmazın gerçek ve objektif değerini tespit eder.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Boşanma nafaka miktarı belirleme yöntemlerinden bahset.	GPT-3.5	1. Yasal Yöntemler: Türk Medeni Kanunu'na göre boşanma nafakasının belirlenmesinde hakim tarafından tarafların ekonomik durumu, gelir düzeyi, evliliğin süresi, çocukların durumu ve diğer tüm hususlar dikkate alınır. 2-Taraflar Arasındaki Anlaşma:Taraflar arasında anlaşma sağlanması durumunda, boşanma protokolünde nafaka miktarı ve süresi belirlenebilir.Anlaşma, mahkeme kararıyla onaylandığında yürürlüğe girer ve bu şekilde belirlenen nafaka miktarı ve süresi geçerli olur.3-Uzmanlar Tarafından Yapılan Ekonomik Analizler: Mahkeme, tarafların ekonomik durumlarını değerlendirmek amacıyla uzmanlar tarafından yapılan ekonomik analizleri de dikkate alabilir. Bu analizler, tarafların gelir, gider ve varlık durumları gibi faktörleri detaylı bir şekilde inceler ve nafaka miktarının adil bir şekilde belirlenmesine yardımcı olur. 4-Makul ve Adil Nafaka: Boşanma nafakası belirlenirken, tarafların ekonomik durumlarına göre makul ve adil bir miktar tespit edilmeye çalışılır. Nafaka miktarı, tarafların geçimini sağlamaya yetecek düzeyde olmalı ve aynı zamanda hakkaniyete uygun olmalıdır. Unutulmaması gereken önemli bir nokta, boşanma ve nafaka konularında profesyonel bir hukukçudan destek almanın önemidir. Bu konuda uzman avukatlardan veya hukuk danışmanlarından destek alarak daha detaylı bilgi ve yönlendirme alabilirsiniz.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Boşanma nafaka miktarı belirleme yöntemlerinden bahset.	GPT-4	1. Çiftin Evlilik Süresi: Çiftin evlilik süresi uzadıkça nafaka miktarı artabilir. 2. Çiftin Kazanç Durumu: Çiftin her bir üyesinin maddi durumu, nafaka miktarını belirlemede etkilidir. 3. Eşin İhtiyacı: Nafaka alacak olan eşin ihtiyaç durumu ve yaşam standardı dikkate alınır. 4. Çocuk Faktörü: Çocuğun velayetine sahip olan eş daha fazla nafaka alabilir. 5. Eşlerin Yaşı ve Sağlık Durumu: Her iki eşin de yaşları, sağlık durumları ve iş bulabilme olanakları nafaka miktarını belirlemede rol oynar. 6. Eşlerin Eğitim Seviyeleri: Eşlerin eğitim seviyeleri, kişiye ne kadar nafaka verileceği konusunda göz önünde bulundurulur. 7. Eşlerin Mesleki Becerileri: Eşlerin iş deneyimleri ve mesleki becerileri, nafaka miktarını belirlemede önemli bir faktördür.
Boşanma nafaka miktarı belirleme yöntemlerinden bahset	GPT-3.5 + RAG	1. Boşanma nafaka miktarı belirleme yöntemleri Türk Medeni Kanunu'nda düzenlenmiştir. 2. Boşanma nafakası, tarafların ekonomik durumu, nafaka talebinde bulunan tarafın ihtiyacı ve diğer tarafların mali durumu gibi faktörlere göre belirlenir. 3. Nafaka miktarı, mahkeme tarafından tarafların durumuna göre takdir edilir ve karar verilir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Boşanma nafaka miktarı belirleme yöntemlerinden bahset.	GPT-4 + RAG	Boşanma davalarında nafaka miktarı, Türk Medeni Kanunu'nun ilgili hükümleri çerçevesinde hâkim tarafından belirlenir ve bu belirlemede tarafların mali güçleri, yaşam standartları, ihtiyaçları ve sosyal durumları dikkate alınır. Nafaka miktarı belirlenirken, nafaka talep edenin yeme, giyinme, barınma, eğitim ve sağlık gibi zorunlu giderleri ile boşanma sonrası mevcut yaşam düzeyini sürdürebilme imkânı değerlendirilir; diğer yandan, nafaka yükümlüsünün gelir durumu, düzenli kazancı, mal varlığı ve ödeme gücü göz önünde bulundurulur. Ayrıca enflasyon, ekonomik koşullar ve hakkaniyet ilkesi de nafaka miktarının belirlenmesinde rol oynar. Hâkim, tüm bu unsurları birlikte değerlendirerek taraflar arasında dengeyi ve adaleti sağlayacak bir miktara hükmeder.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Boşanma nafaka miktarı belirleme yöntemlerinden bahset.	Our System	Türk Medeni Kanunu'nda nafaka miktarını belirleyen matematiksel bir formül bulunmaz. Hâkim, her somut olayın kendine özgü koşullarını değerlendirerek hakkaniyet ilkesi çerçevesinde bir miktar belirler. Nafakanın türüne göre dikkate alınan kriterler farklılaşır: a) Yoksulluk Nafakası: Boşanma yüzünden yoksulluğa düşecek olan tarafa, kusuru daha ağır olmamak şartıyla ödenir. Miktar belirlenirken şu hususlara bakılır: Tarafların Ekonomik ve Sosyal Durumları. Ödeyecek Olan Eşin Mali Gücü Tarafların Kusur Oranları. b) İştirak Nafakası (Çocuk İçin): Velayeti kendisine verilmeyen eşin, çocuğun bakım ve eğitim giderlerine gücü oranında katılması için ödediği nafakadır. Belirlenirken esas alınanlar: Çocuğun İhtiyaçları.Anne ve Babanın Mali Güçleri: Nafaka yükümlülüğü sadece babaya veya anneye ait değildir. Hâkim, her iki ebeveynin de gelir durumunu değerlendirerek bir denge kurar. c) Tedbir Nafakası: Boşanma davası devam ederken, maddi olarak zor duruma düşecek eş veya çocuklar için geçici olarak hükmedilen nafakadır. Belirlenmesinde de yine tarafların o anki barınma ve geçim ihtiyaçları ile mali güçleri dikkate alınır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Çevre mevzuatındaki güncel düzenlemeler ile ilgili bilgi ver.	GPT-3.5	1. Plastik Poşetler: Plastik poşet kullanımı ve tüketimi sınırlandırılmıştır. Marketlerde ücretli olarak verilen poşetlerin kullanımı azaltılması için çalışmalar yürütülmektedir. 2. Atık Yönetimi: Atık yönetimi ile ilgili düzenlemeler artırılmıştır. Geri dönüşüm ve atık ayrıştırma konularında daha sıkı kurallar getirilmiştir. 3. Hava Kirliliği: Sanayi tesisleri ve araçlardan kaynaklanan hava kirliliğinin azaltılması için yeni düzenlemeler yapılmıştır. Özellikle fosil yakıtların kullanımı sınırlandırılmaya çalışılmaktadır. 4. Su Kirliliği: Su kaynaklarının kirlenmesinin önlenmesi için su arıtma tesislerinin daha etkin kullanımı ve denetlenmesine yönelik düzenlemeler yapılmıştır. Su kirliliğine neden olan faaliyetlere yönelik cezai yaptırımlar artırılmıştır. 5. Yaban Hayatı Koruma: Doğal yaşam alanlarının korunması ve yaban hayatının sürdürülebilirliği için yeni koruma alanları oluşturulmuş ve habitatlar korunmaya çalışılmaktadır. 6. İklim Değişikliği: Paris Anlaşması kapsamında Türkiye de iklim değişikliğiyle mücadele kapsamında taahhütlerini yerine getirmektedir. Yenilenebilir enerji kaynaklarına yatırımlar artırılmıştır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Çevre mevzuatındaki güncel düzenlemeler ile ilgili bilgi ver.	GPT-4	1. Plastik torbaların ücretli hale getirilmesi: 1 Ocak 2019'dan bu yana plastik poşetlerin ücretli olması çevre mevzuatının bir parçası haline gelmiştir. 2. Ambalaj Atıklarının Kontrolü Yönetmeliği: Ambalaj üreten, ithal eden ve ambalajlı ürünler satan firmaların çevre ve atık yönetimi konusunda sorumlulukları düzenlenmiştir. 3. Tek kullanımlık plastiklerin yasaklanması: Türkiye, 2021'den itibaren bazı tek kullanımlık plastik ürünlerin üretimini ve kullanımını yasakladı. 4. Geri dönüşüm süreçlerinin teşvik edilmesi: Son düzenlemeler geri kazanımı ve geri dönüşümü teşvik etmek için tasarlanmıştır. 5. Su kaynaklarının korunması: Yeni yasal düzenlemeler ve uygulamalar, kirliliğin önlenmesi ve su kaynaklarının sürdürülebilir kullanımının sağlanması için tavizsiz bir yaklaşımla yürütülmektedir. 6. Elektrikli ve elektronik eşya atıklarının toplanması ve geri dönüşümü alanında düzenlemeler getirilmiştir. 7. Hava kirliliği ile mücadelede sanayi tesisleri için yeni standartlar belirlenmiştir. 8. Atık yönetimi ve sıfır atık politikaları daha da sıkılaştırılmıştır. Lütfen detaylı, güncel ve resmi bilgileri Çevre ve Şehircilik Bakanlığı'nın resmi web sitesinden takip ediniz. Bu bilgiler genel ve özet niteliğindedir, tüm detayları kapsamamaktadır.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Çevre mevzuatındaki güncel düzenlemeler ile ilgili bilgi ver.	GPT-3.5 + RAG	Türkiye'de çevre mevzuatındaki güncel düzenlemeler şunlardır: Florlu Sera Gazlarına İlişkin Yönetmelik (15.10.2024) Endüstriyel Emisyonların Yönetimi Yönetmeliği (Ocak 2025) Çevre Kanunu'na Göre 2025 Yılı İdari Para Cezaları Tebliği — çevre ihlallerine uygulanacak cezalar güncellendi
Çevre mevzuatındaki güncel düzenlemeler ile ilgili bilgi ver.	GPT-4 + RAG	Türkiye'nin çevre mevzuatı, sürdürülebilirlik ve çevrenin korunması hedefiyle sürekli güncelleniyor. 2024 ve 2025'te öne çıkan düzenlemeler arasında özellikle Florlu Sera Gazlarına İlişkin Yönetmelik (15 Ekim 2024) ve Endüstriyel Emisyonların Yönetimi Yönetmeliği (14 Ocak 2025) yer alıyor; bu düzenlemeler sanayi kaynaklı emisyonları hem çevre hem de insan sağlığı açısından ele alıp entegre kirlilik önlemeye odaklanıyor. Aynı zamanda, 2025 yılı için Çevre Kanunu çerçevesindeki idari para cezaları yeniden değerleme oranına göre güncellendi; böylece çevreye zarar veren ihlallerde caydırıcılık artıyor.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
Çevre mevzuatındaki güncel düzenlemeler ile ilgili bilgi ver.	Our System	Son yıllardaki (2023-2025 dönemi) önemli ve güncel düzenlemeler şunlardır: Depozito İade Sistemi: 2025 yılı itibarıyla ülke genelinde kademeli olarak yaygınlaşması hedeflenen bu sistemle, belirli içecek ambalajlarına (cam, PET, alüminyum) bir depozito bedeli eklenmesi ve bu ambalajların iade noktalarına geri getirilmesiyle bedelin tüketiciye ödenmesi amaçlanmaktadır. Türkiye Ulusal İklim Değişikliği Stratejisi ve Eylem Planı: Türkiye'nin 2053 Net Sıfır Emisyon hedefi doğrultusunda, enerji, sanayi, ulaştırma ve tarım gibi sektörlerde karbon emisyonlarının azaltılmasına yönelik stratejiler güncellenmektedir. Bu kapsamda bir "İklim Kanunu" taslağı üzerindeki çalışmalar devam etmektedir. Kanun yasalaştığında, bir Emisyon Ticaret Sistemi kurulması beklenmektedir. Sıfır Atık Yönetmeliği: "Sıfır Atık" projesinin yasal altyapısını oluşturan bu yönetmelik, atıkların kaynağında ayrı toplanması, geri kazanımının artırılması ve israfın önlenmesi konularında kamu kurumları ve özel sektöre yönelik yükümlülükleri sürekli güncellemektedir.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İşçi hakları mevzuatında son değişiklikler neler?	GPT-3.5	- İş Kanunu'nda yapılan son değişiklikle, ücretsiz izne çıkartılan işçilere kıdem tazminatı ve ihbar tazminatı ödemesi yapılması zorunlu hale getirilmiştir. - İş Sağlığı ve Güvenliği Kanunu'nda yapılan değişiklikle işverenlere iş sağlığı ve güvenliği konusunda daha fazla sorumluluk getirilmiştir. - Pandemi döneminde yapılan düzenlemelerle uzaktan çalışma gibi esnek çalışma modellerine ilişkin düzenlemeler getirilmiştir. - Asgari ücrette belirlenen artışlar ve işçi haklarını korumaya yönelik yapılan düzenlemeler de önemli değişiklikler arasında yer almaktadır.
İşçi hakları mevzuatında son değişiklikler neler?	GPT-4	Üzgünüm, ben bir yapay zekayım ve hali hazırda olan veritabanımı güncellemek gibi bir yeteneğim yok. Bu yüzden en son ve güncel hukuki değişiklikleri size sağlayamam. Bu tür bilgiler için resmi hükümet web siteleri, iş hukuku uzmanları veya güvendiğiniz haber kaynaklarından bilgi edinmenizi öneririm.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İşçi hakları mevzuatında son değişiklikler neler?	GPT-3.5 + RAG	Asgari Ücret Artışı: 2025 için brüt asgari ücret 26.005,50 TL olarak belirlendi. Kıdem Tazminatı Tavanı Yükseltildi: İlk altı ayda brüt 41.828,42 TL iken 46.655,43 TL'ye çıktı. İdari Para Cezalarında Artış: İş Kanunu kapsamındaki cezalar %43,93 oranında güncellendi. KEP ile Bildirim Zorunluluğu: İşverenin yapacağı bildirimler artık kayıtlı elektronik posta (KEP) ile de yapılabilir. Turizm Sektöründe Hafta Tatili Düzenlemesi: Turizm çalışanlarının hafta tatilleriyle ilgili yeni hükümler getirildi. İşyeri Tehlike Sınıflarında Güncelleme ve İş Sağlığı Güvenliği Eğitimleri ile ilgili yeni düzenlemeler hayata geçirildi

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İşçi hakları mevzuatında son değişiklikler neler?	GPT-4 + RAG	2025 yılında işçi haklarına dair mevzuatta göze çarpan önemli güncellemeler hayata geçirildi. Asgari ücret, 1 Ocak itibarıyla brüt 26.005,50 TL'ye yükseltilirken; kıdem tazminatı tavanı, ilk altı ay için brüt 46.655,43 TL olarak güncellendi. Yine aynı dönemde, işçilerin korunması amacıyla İş Kanunu kapsamındaki idari para cezaları yaklaşık %44 oranında artırıldı. İletişime de modern bir adım atıldı; işçiye yapılacak bildirimlerin yazılı veya KEP (kayıtlı elektronik posta) yoluyla yapılabilmesine olanak tanındı. Ayrıca, turizm sektöründe çalışanların hafta tatili düzenlemelerinde ve iş sağlığı ve güvenliği alanlarında da yeni kriterler ve sınıflandırmalar getirildi.

The table continues...

Table 5.1 – *the continuation*

Question	**Model**	**Answer**
İşçi hakları mevzuatında son değişiklikler neler?	Our System	İşçi hakları mevzuatında son yıllarda (2023-2025) yapılan veya etkileri devam eden en önemli düzenlemeler şunlardır: EYT Düzenlemesi: 2023'te yapılan en büyük değişikliklerden biridir. 8 Eylül 1999 öncesi sigorta girişi olanlar için yaş şartını kaldırarak prim günü ve sigortalılık süresini tamamlayanlara emeklilik hakkı tanımıştır.Emekli olup aynı işyerinde çalışmaya devam edenler için "Sosyal Güvenlik Destek Primi" (SGDP) ile çalışma ve kıdem tazminatı haklarının nasıl işleyeceği gibi konular önemini korumaktadır. Asgari Ücret ve Vergi İstisnaları: Son yıllardaki önemli bir gelişme, tüm ücretli çalışanların maaşlarının asgari ücret kadar olan kısmının Gelir Vergisi ve Damga Vergisinden muaf tutulmasıdır. Bu, işçinin net gelirini artıran kalıcı bir düzenleme haline gelmiştir. Uzaktan Çalışma Yönetmeliği: Pandemi sonrası kalıcı hale gelen uzaktan çalışma modelinin hukuki altyapısı "Uzaktan Çalışma Yönetmeliği" ile netleştirilmiştir. Bu yönetmelik; uzaktan çalışanın giderlerinin karşılanması (elektrik, internet vb.), çalışma saatlerinin belirlenmesi, iş sağlığı ve güvenliği önlemleri gibi konularda işverenlere ve işçilere yol göstermektedir.

REFERENCES

[1]A. Gandomi and M. Haider, "Beyond the hype: Big data concepts, methods, and analytics", *International Journal of Information Management*, vol. 35, no. 2, pp. 137–144, 2015.

[2]B. Marr, "How much data do we create every day? the mind-blowing stats everyone should read", *Forbes*, vol. 21, pp. 1–5, 2018.

[3]R. C. Berring, "Legal research and the world of thinkable thoughts", *Journal of Appellate Practice and Process*, vol. 2, p. 305, 2000.

[4]S. Prechal and R. Widdershoven, "Redefining the relationship between 'rewe-effectiveness' and effective judicial protection", *Review of European Administrative Law*, vol. 4, no. 2, pp. 31–50, 2011.

[5]K. D. Ashley, *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. MIT Press, 1988.

[6]B. Hachey and C. Grover, "Extractive summarisation of legal texts", *Artificial Intelligence and Law*, vol. 14, no. 4, pp. 305–345, 2006.

[7]J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "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)*, Association for Computational Linguistics, 2019, pp. 4171–4186.

[8]T. Ansay, "Türk hukuku hakkinda almanca, fransızca, İngilizce kitaplar bibliyografyasi", *Ankara Üniversitesi Hukuk Fakültesi Dergisi*, vol. 26, no. 3, pp. 349–387, 1969.

[9]K. Gözler, "Türk anayasa hukuku", *Ankara Üniversitesi SBF Dergisi*, vol. 55, no. 4, 2000.

[10]M. S. Özbek, "Türkiye'de elektronik devlet uygulaması: Uyap örneği", *Türk İdare Dergisi*, no. 459, 2008.

[11]J. Sawicki, M. Ganzha, and M. Paprzycki, "The state of the art of natural language processing: A systematic automated review of nlp literature using nlp techniques", *Data Intelligence*, vol. 5, no. 3, pp. 707–749, 2023.

[12]D. M. Katz, D. Hartung, L. Gerlach, A. Jana, and M. J. Bommarito II, "Natural language processing in the legal domain", *arXiv preprint arXiv:2302.12039*, 2023.

[13]G. Lewis, *The Turkish Language Reform: A Catastrophic Success*. Oxford University Press, 1999.

[14]K. Oflazer, "Two-level description of turkish morphology", *Literary and Linguistic Computing*, vol. 9, no. 2, pp. 137–148, 1994.

[15]H. Sak, T. Güngör, and M. Saraçlar, "Turkish language resources: Morphological parser, morphological disambiguator and web corpus", in *Advances in Natural Language Processing: 6th International Conference, GoTAL 2008, Gothenburg, Sweden, August 25–27, 2008, Proceedings*, Springer, 2008, pp. 417–427.

[16]D. M. Katz, M. J. Bommarito II, and J. Blackman, "A general approach for predicting the behavior of the supreme court of the united states", *PLOS ONE*, vol. 12, no. 4, e0174698, 2017.

[17]Ş. Ruhi and D. Zeyrek, "Türkçede kiplik belirteçleri ve çekim ekleri ilişkisi üzerine bazı gözlemler", *Dilbilim Araştırmaları Dergisi*, vol. 8, pp. 105–111, 1997.

[18]A. Boer, R. Hoekstra, R. Winkels, T. Van Engers, and F. Willaert, "Metalex: Legislation in xml", in *Legal Knowledge and Information Systems: JURIX 2002, The Fifteenth Annual Conference*, IOS Press, 2002, pp. 1–10.

[19]C. Toraman, E. H. Yilmaz, F. Sahinuc, and O. Ozcelik, "Impact of tokenization on language models: An analysis for turkish", *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 4, pp. 1–21, 2023.

[20]G. Eryiğit, J. Nivre, and K. Oflazer, "Dependency parsing of turkish", *Computational Linguistics*, vol. 34, no. 3, pp. 357–389, 2008.

[21]E. Mumcuoglu, C. E. Ozturk, H. M. Ozaktas, and A. Koc, "Natural language processing in law: Prediction of outcomes in the higher courts of turkey", *Information Processing & Management*, vol. 58, no. 5, p. 102 684, 2021.

[22]Y. B. Kaya and A. C. Tantug, "Effect of tokenization granularity for turkish large language models", *Intelligent Systems with Applications*, vol. 21, p. 200 335, 2024.

[23]F. Zeidi, M. F. Amasyali, and C. Erol, “Legalturk: Optimized bert for multi-label text classification and named entity recognition”, *arXiv preprint arXiv:2407.00648*, 2024.

[24]A. Safaya, E. Kurtulus, A. Goktogan, and D. Yuret, “Mukayese: Turkish nlp strikes back”, *arXiv preprint arXiv:2203.01215*, 2022.

[25]M. A. Bayram, A. A. Fincan, A. S. Gumus, B. Diri, S. Yildirim, and O. Aytas, “Setting standards in turkish nlp: Tr-mmlu for large language model evaluation”, *arXiv preprint arXiv:2501.00593*, 2024.

[26]M. A. A. Albayati and O. Findik, “Turkish legal single-document summarizing”, in *Proceedings of the International Conference on Information Technologies and Their Applications*, Springer, 2024, pp. 32–41.

[27]N. Pipitone and G. H. Alami, “Legalbench-rag: A benchmark for retrieval-augmented generation in the legal domain”, *arXiv preprint arXiv:2408.10343*, 2024.

[28]R. C. Barron, M. E. Eren, O. M. Serafimova, C. Matuszek, and B. S. Alexandrov, “Bridging legal knowledge and ai: Retrieval-augmented generation with vector stores, knowledge graphs, and hierarchical non-negative matrix factorization”, *arXiv preprint arXiv:2502.20364*, 2025.

[29]S. Yao, Q. Ke, Q. Wang, K. Li, and J. Hu, “Lawyergpt: A legal large language model with enhanced domain knowledge and reasoning capabilities”, in *Proceedings of the 3rd International Symposium on Robotics, Artificial Intelligence and Information Engineering*, 2024, pp. 108–112.

[30]J. Li, L. Qian, P. Liu, and T. Liu, “Construction of legal knowledge graph based on knowledge-enhanced large language models”, *Information*, vol. 15, no. 11, p. 666, 2024.

[31]D. Schwarcz, S. Manning, P. Barry, D. R. Cleveland, J. J. Prescott, and B. Rich, “Ai-powered lawyering: Ai reasoning models, retrieval-augmented generation, and the future of legal practice”, *arXiv preprint arXiv:2503.xxxxx*, 2025.

[32]M. R. Kabir, R. M. Sultan, F. Rahman, *et al.*, “Legalrag: A hybrid rag system for multilingual legal information retrieval”, *arXiv preprint arXiv:2504.16121*, 2025.

[33]N. Guha, J. Nyarko, D. Ho, *et al.*, “Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models”, *Advances in Neural Information Processing Systems*, vol. 36, pp. 44 123–44 279, 2023.

[34]Y. Dai, D. Feng, J. Huang, *et al.*, "Laiw: A chinese legal large language models benchmark", *arXiv preprint arXiv:2310.05620*, 2023.

[35]H. Zhong, C. Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun, "How does nlp benefit the legal system: A summary of legal artificial intelligence", *arXiv preprint arXiv:2004.12158*, 2020.

[36]J. Niklaus, V. Matoshi, M. Stürmer, I. Chalkidis, and D. E. Ho, "Multilegalpile: A 689gb multilingual legal corpus", *arXiv preprint arXiv:2306.02069*, 2023.

[37]GlobaLex, NYU, *Turkish public law and legal research*, `https://www.nyulawglobal.org/globalex/Turkey1.html`, Accessed: 2025-07-23, 2025.

[38]F. Ariai and G. Demartini, "Natural language processing for the legal domain: A survey of tasks, datasets, models, and challenges", *arXiv preprint arXiv:2410.21306*, 2024.

[39]D. Trautmann, A. Petrova, and F. Schilder, "Legal prompt engineering for multilingual legal judgement prediction", *arXiv preprint arXiv:2212.02199*, Dec. 2022, Zero-shot LPE enables transfer across legal systems and languages. [Online]. Available: `https://arxiv.org/abs/2212.02199`.

[40]C. Coltekin, "A freely available morphological analyzer for turkish", in *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2010, pp. 19–28.

[41]A. Öztürel, T. Kayadelen, and I. Demirsahin, "A syntactically expressive morphological analyzer for turkish", in *Proceedings of the 14th International Conference on Finite-State Methods and Natural Language Processing*, 2019, pp. 65–75.

[42]H. Sak, T. Güngör, and M. Saraçlar, "Morphological disambiguation of turkish text with perceptron algorithm", in *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics*, Springer, 2007, pp. 107–118.

[43]E. Yildiz, C. Tirkaz, H. Sahin, M. Eren, and O. Sonmez, "A morphology-aware network for morphological disambiguation", in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 30, 2016.

[44]A. A. Akın and M. D. Akın, "Zemberek: An open source nlp framework for turkic languages", *Structure*, vol. 10, pp. 1–5, 2007.

[45]C. Sayallar, "Context-aware lemmatization and morphological tagging method in turkish", *arXiv preprint arXiv:2501.02361*, 2025.

[46]O. Toporkov and R. Agerri, "On the role of morphological information for contextual lemmatization", *Computational Linguistics*, vol. 50, no. 1, pp. 157–191, 2024.

[47]S. Schweter, *Berturk: Bert models for turkish*, version 1.0.0, Apr. 2020. DOI: 10.5281/zenodo.3770924. [Online]. Available: https://doi.org/10.5281/zenodo.3770924.

[48]J. Cui, X. Shen, and S. Wen, "A survey on legal judgment prediction: Datasets, metrics, models and challenges", *IEEE Access*, vol. 11, p. 102 050, 2023.

[49]V. Rodrıguez-Doncel and E. Montiel-Ponsoda, "Lynx: Towards a legal knowledge graph for multilingual europe", *Law in Context: A Socio-Legal Journal*, vol. 37, p. 175, 2020.

[50]C. Çetindağ, B. Yazıcıoğlu, and A. Koç, "Named entity recognition in turkish legal texts", *Natural Language Engineering*, vol. 29, no. 3, pp. 615–642, 2023.

[51]I. Chalkidis, M. Fergadiotis, and I. Androutsopoulos, "Multieurlex: A multilingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer", *arXiv preprint arXiv:2109.00904*, 2021.

[52]D. Bandyopadhyay, A. De, B. Gain, T. Saikh, and A. Ekbal, "A deep transfer learning method for cross-lingual natural language inference", in *Proceedings of the 13th Language Resources and Evaluation Conference (LREC)*, 2022, pp. 3084–3092.

[53]I. J. Unanue, G. Haffari, and M. Piccardi, "T3l: Translate-and-test transfer learning for cross-lingual text classification", *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1147–1161, 2023.

[54]B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo, "Evaluating word embedding models: Methods and experimental results", *APSIPA Transactions on Signal and Information Processing*, vol. 8, e19, 2019.

[55]N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, "Mteb: Massive text embedding benchmark", *arXiv preprint arXiv:2210.07316*, 2022.

[56]T. Schnabel, I. Labutov, D. Mimno, and T. Joachims, "Evaluation methods for unsupervised word embeddings", in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2015, pp. 298–307.

[57]P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”, *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[58]A. Niculescu-Mizil and R. Caruana, “Predicting good probabilities with supervised learning”, in *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, 2005, pp. 625–632.

[59]J. Davis and M. Goadrich, “The relationship between precision-recall and roc curves”, in *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006, pp. 233–240.

[60]I. Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. Androutsopoulos, “Large-scale multi-label text classification on eu legislation”, *arXiv preprint arXiv:1906.02192*, 2019.

[61]V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks”, *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, P10008, 2008.

[62]S. Fortunato, “Community detection in graphs”, *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[63]J. H. Fowler, T. R. Johnson, J. F. Spriggs, S. Jeon, and P. J. Wahlbeck, “Network analysis and the law: Measuring the legal importance of precedents at the us supreme court”, *Political Analysis*, vol. 15, no. 3, pp. 324–346, 2007.