AN EXPLAINABLE CACHE-AUGMENTED LLM FOR REGULATION-BASED QUESTION ANSWERING IN TURKISH FIRE SAFETY

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

BY

ESRA ŞEKERCI

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

NOVEMBER 2025

Approval of the thesis:

AN EXPLAINABLE CACHE-AUGMENTED LLM FOR REGULATION-BASED QUESTION ANSWERING IN TURKISH FIRE SAFETY

submitted by **ESRA ŞEKERCI** 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. İnci Batmaz
Supervisor, **Statistics, METU** _____

Examining Committee Members:

Prof. Dr. Ceylan Yozgatlıgil
Statistics, METU _____

Prof. Dr. İnci Batmaz
Statistics, METU _____

Prof. Dr. İhsan Tolga Medeni
Management Information Systems, Ankara Yıldırım Beyazıt University _____

Date:21.11.2025

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: Esra Şekerci

Signature :

ABSTRACT

AN EXPLAINABLE CACHE-AUGMENTED LLM FOR REGULATION-BASED QUESTION ANSWERING IN TURKISH FIRE SAFETY

Şekerci, Esra
M.S., Department of Statistics
Supervisor: Prof. Dr. İnci Batmaz

NOVEMBER 2025, 78 pages

Turkish fire-safety regulation constitutes a safety-critical domain where low-resource language challenges demand strict grounding of model outputs in authoritative sources. Standard large language models are prone to parametric hallucinations, particularly with respect to numeric safety thresholds and technical citations, which undermines their reliability for regulatory decision support. To address this problem, this thesis proposes an explainable, end-to-end architecture that integrates a structured section-level regulatory corpus, hybrid information retrieval, and cache-augmented generation using a parameter-efficiently adapted instruction-tuned backbone. The system is evaluated on a curated regulation-based question–answering dataset, where the Fusion+Rerank retrieval configuration achieves a Recall@10 of 0.93, ensuring robust and relevant context selection. Conditioned on this retrieved and cached context, the proposed pipeline attains a BERTScore-F1 of 0.93 and a BLEU score of 0.34, indicating high fidelity to authoritative regulatory provisions. From a theoretical perspective, word-level explainability analysis demonstrates a functional decoupling between system components: the retrieval module isolates semantically relevant regulatory topics, while the generative component accurately extracts quantitative constraints from

the cached context. Overall, the findings provide rigorous empirical evidence on the capabilities and limitations of cache-augmented, explainable large language model architectures for regulation-based question answering and inform the design of related systems in specialised regulatory domains and other low-resource language settings.

Keywords: Cache-Augmented Generation, Explainable AI, Legal Decision Support, Regulation-based Question Answering, Turkish Building Fire-Safety Regulation

ÖZ

TÜRKÇE YANGIN GÜVENLİĞİ YÖNETMELİĞİNE DAYALI SORU–CEVAP SİSTEMİ İÇİN AÇIKLANABİLİR ÖNBELLEK ARTIRIMLI BÜYÜK DİL MODELİ

Şekerci, Esra
Yüksek Lisans, İstatistik Bölümü
Tez Yöneticisi: Prof. Dr. İnci Batmaz

Kasım 2025, 78 sayfa

Türkçe yangın güvenliği yönetmeliği; düşük kaynaklı dil zorluklarının, model çıktılarında yetkili kaynaklara sıkı bir temellendirmeyi zorunlu kıldığı, güvenlik açısından kritik bir alan teşkil etmektedir. Ancak standart büyük dil modelleri, özellikle sayısal güvenlik eşikleri ve teknik atıflar konusunda parametrik halüsinasyonlara yatkındır ve bu durum, yönetmelik tabanlı karar destek süreçlerinde güvenilirliği zedelemektedir. Bu sorunu çözmek amacıyla tezde; bölüm düzeyinde yapılandırılmış yönetmelik derlemi, hibrit bilgi erişimi ve parametre-verimli uyarlanmış yönerge ayarlı (instruction-tuned) bir temel model içeren, açıklanabilir ve uçtan uca bir mimari önerilmektedir. Önbellek artırımlı üretim (cache-augmented generation) yaklaşımını kullanan bu sistem, oluşturulan yönetmelik tabanlı soru–cevap veri kümesi üzerinde değerlendirilmiştir. Deneylerde, Füzyon+Yeniden Sıralama (Fusion+Rerank) bilgi erişim konfigürasyonu @10'da 0.93 Anımsama (Recall) değerine ulaşarak kararlı bir bağlam seçimi sağlamıştır. Bu bağlam üzerinden çalışan önerilen mimari ise 0.93 BERTScore-F1 ve 0.34 BLEU skoru elde etmiş, böylece yetkili yönetmelik hüküm-

lerine yüksek düzeyde metne bağlılık (fidelity) sergilemektedir. Teorik açıdan sözcük düzeyinde açıklanabilirlik analizi, sistem bileşenleri arasında işlevsel bir ayrışma olduğunu doğrulamaktadır: Bilgi erişim modülü anlamsal konuları ayrıştırırken, üretici bileşen önbellekteki bağlamdan sayısal kısıtları hassas biçimde çıkarmaktadır. Genel olarak sonuçlar, yönetmelik tabanlı soru–cevap için önbellek artırımlı, açıklanabilir büyük dil modeli mimarilerinin yetkinlikleri ve sınırlarına ilişkin deneysel kanıtlar ortaya koymakta ve düşük kaynaklı dillerdeki benzer sistemlerin tasarımına yönelik ilkelerin geliştirilmesine katkıda bulunmaktadır.

Anahtar Kelimeler: Açıklanabilir Yapay Zekâ, Binaların Yangından Korunması Hakkında Yönetmelik, Hukuki Karar Destek Sistemi, Önbellek Artırımlı Üretim, Yönetmelik Tabanlı Soru–Cevap

To those who contributed to this thesis study,

ACKNOWLEDGMENTS

I wish to express my sincere gratitude to my thesis supervisor, Prof. Dr. İnci Batmaz, for the trust she has placed in me and for her patience, and encouragement throughout this study.

I am also grateful to Prof. Dr. Özlem İlk Dağ and Prof. Dr. Ceylan Yozgatlıgil for their inspiring teaching and exemplary academic stance, and I thank Prof. Dr. Yozgatlıgil and Prof. Dr. İhsan Tolga Medeni, as members of my thesis examining committee, for their valuable suggestions and constructive feedback, which improved the quality of this thesis.

I owe much to İbrahim Gülay for his constant support, understanding, and presence throughout this journey.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xiv

LIST OF FIGURES xv

LIST OF ABBREVIATIONS xvii

CHAPTERS

1 INTRODUCTION 1

2 LITERATURE REVIEW 5

2.1 Foundations of Large Language Models 5

2.1.1 Probability Modeling and Sequential Prediction 5

2.1.2 Transformer Architecture 7

2.1.3 Decoder-Only LLMs, Scaling, and Efficiency 9

2.2 Tokenization and Language Adaptation for Turkish 10

2.2.1 Morphological Complexity in Turkish 10

2.2.2 Subword and Neural Tokenization Approaches 10

2.2.3 Domain-Specific Challenges and Evaluation 11

2.3 Explainable Artificial Intelligence in Language Models 12
2.3.1 Evolution of Explainability in High-Capacity Models 12
2.3.2 Model-Agnostic and Model-Specific Techniques 12
2.3.3 Explainability for LLMs and Safety-Critical/Legal Contexts . . 13
2.4 LLM-Powered Decision Support in Legal and Safety-Critical Domains 13
2.4.1 LLMs in Disaster and Risk Management 13
2.4.2 Legal and Regulatory Decision-Support 14
2.5 Research Gaps and Theoretical Motivation 15
3 METHODOLOGY 17
3.1 Overview 17
3.2 Theoretical Foundations and Model Architecture 17
3.2.1 Preliminaries and Scaling Guidance 17
3.2.2 Model Candidates and Uniform Screening Setup 19
3.3 Parameter-Efficient Fine-Tuning 21
3.3.1 Low Rank Adaptation 21
3.3.2 Quantization 22
3.4 Cache-Augmented Generation 23
3.4.1 Retrieval Configuration 26
3.5 Evaluation Metrics and Explainability Techniques 28
3.5.1 Text Generation Quality Metrics 28
3.5.2 Retrieval Quality Metrics 29
3.5.3 Explainability Techniques 31
3.6 Environment Settings 31

4 EMPIRICAL ANALYSIS 33
4.1 Experimental Setup 33
4.2 Design of the Datasets 33
4.2.1 Regulation Dataset 34
4.2.1.1 Preprocessing and Exploratory Data Analysis 34
4.2.1.2 Question–Answer Generation 40
4.2.2 Court Decision Dataset 44
4.3 Model Evaluation 45
4.3.1 Dataset Splits 45
4.3.2 Zero-Shot LLM Baselines 45
4.3.3 Fine-Tuned Model 50
4.4 CAG Evaluation 52
4.4.1 Regulation-Only CAG with the Fine-Tuned Model 52
4.4.2 Retrieval Metrics Results 54
4.4.2.1 Retrieval Results on Regulation QA 54
4.4.2.2 Retrieval Results on Court–to–Regulation Linking . . . 55
4.4.3 Explainability Analysis 56
4.4.3.1 Retrieval Reasoning 56
4.4.3.2 Generation Reasoning 59
4.5 Efficiency Analysis 63
4.6 Discussion 64
5 CONCLUSION 67
REFERENCES 71

LIST OF TABLES

TABLES

Table 4.1 Corpus statistics under Gemma-2 tokenization 36

Table 4.2 Word-level statistics for the regulation corpus 38

Table 4.3 Descriptive statistics for the court decision dataset 44

Table 4.4 Descriptive statistics of the QA dataset splits 45

Table 4.5 Zero-shot performance of instruction-tuned LLMs 46

Table 4.6 Zero-shot answers for Example 1 (idx=26) 47

Table 4.7 Zero-shot answers for Example 2 (idx=58) 48

Table 4.8 Zero-shot answers for Example 3 (idx=167) 49

Table 4.9 Legal QA performance across inference regimes 52

Table 4.10 Representative CAG answers on the legal QA test set 53

Table 4.11 Retrieval effectiveness for different configurations 54

Table 4.12 Retrieval performance for linking court decisions to the regulation . 55

Table 4.13 Word-level generation metrics for the predicted answer (idx=26) . . 59

Table 4.14 Word-level generation metrics for the predicted answer (idx=58) . . 61

Table 4.15 Word-level generation metrics for the predicted answer (idx=167) . . 62

Table 4.16 Efficiency of QA inference experiments 63

LIST OF FIGURES

FIGURES

Figure 2.1 The Transformer model architecture (adapted from [13]). 7

Figure 3.1 Comparison of RAG and CAG workflows 24

Figure 4.1 Top-25 content terms without stopword and numeral 37

Figure 4.2 Section length by regulatory category 39

Figure 4.3 Excerpt from the generated regulation–QA dataset 43

Figure 4.4 Loss across QLoRA sweep configurations 51

Figure 4.5 Word-level SHAP values for question (idx=26) 56

Figure 4.6 Word-level SHAP values for question (idx=58) 57

Figure 4.7 Word-level SHAP values for question (idx=167) 58

LIST OF ABBREVIATIONS

ABBREVIATIONS

AYM	Anayasa Mahkemesi (Constitutional Court)
BF16	Brain Floating Point 16
BLEU	Bilingual Evaluation Understudy
BM25	Okapi BM25 Ranking Function
BPE	Byte-Pair Encoding
CAG	Cache-Augmented Generation
DSS	Decision Support System
EM	Exact Match
GDPR	General Data Protection Regulation
GPU	Graphics Processing Unit
IDF	Inverse Document Frequency
IQR	Interquartile Range
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
KV	Key-Value (in transformer cache)
LCS	Longest Common Subsequence
LIME	Local Interpretable Model-agnostic Explanations
LLM	Large Language Model
LoRA	Low-Rank Adaptation
MLE	Maximum Likelihood Estimation
MLP	Multilayer Perceptron
MRR	Mean Reciprocal Rank
nDCG	Normalized Discounted Cumulative Gain
NFPA	National Fire Protection Association
NLP	Natural Language Processing

OCR	Optical Character Recognition
PEFT	Parameter-Efficient Fine-Tuning
QA	Question Answering
QLoRA	Quantized Low-Rank Adaptation
RAG	Retrieval-Augmented Generation
RoPE	Rotary Positional Embedding
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SHAP	SHapley Additive exPlanations
SimCSE	Simple Contrastive Learning of Sentence Embeddings
STPA	System-Theoretic Process Analysis
STTR	Standardized Type-Token Ratio
TF	Term Frequency
TF–IDF	Term Frequency–Inverse Document Frequency
TS	Turkish Standards
TTR	Type-Token Ratio
VRAM	Video Random Access Memory
XAI	Explainable Artificial Intelligence

CHAPTER 1

INTRODUCTION

Large language models (LLMs) now achieve strong performance on a wide range of general-purpose tasks, yet their deployment in specialized, high-stakes domains remains challenging. When model outputs may influence safety-critical decisions, systems must not only be accurate, but also explicitly grounded in authoritative sources and amenable to scrutiny. These requirements are particularly stringent in legal and regulatory contexts, where small changes in thresholds, conditions, or exceptions can have disproportionate consequences. The difficulty is amplified for linguistically complex, low-resource languages such as Turkish, where rich morphology and limited supervision complicate both retrieval and generation.

This thesis studies these issues in the context of Turkish fire-safety regulation, focusing on the *Binaların Yangından Korunması Hakkında Yönetmelik* (Building Fire-Safety Regulation) and its official implementation guide. The central objective is to design and empirically evaluate a regulation-grounded question answering (QA) system in which an instruction-tuned LLM answers questions about fire safety using only information from these texts, while remaining traceable to the underlying provisions. Concretely, the work seeks to (i) retrieve relevant sections for short questions and long, narrative inputs, (ii) generate answers that stay close to the regulatory language, and (iii) expose which parts of the queries and retrieved sections drive the system's decisions. A curated set of high-court decisions is used as an additional testbed to examine whether the same retrieval and cache mechanisms can operate on long, noisy judicial narratives and still recover the governing regulatory sections.

To achieve these objectives, this thesis systematically addresses five primary research questions anchored in the theoretical challenges of legal text processing. First, it in-

vestigates the comparative efficacy of sparse lexical versus dense semantic retrieval representations for agglutinative languages, specifically asking whether morphology-aware normalization offers superior alignment with technical regulatory nomenclature than embedding-based methods (RQ1). This addresses the theoretical tension between surface-form precision and semantic generalization in low-resource domains. Second, the study examines the capability of instruction-tuned models to adhere to strict normative constraints, analyzing the extent to which domain-specific fine-tuning is required to suppress parametric hallucinations—such as fabricated citations or inexact numeric thresholds—in favor of source-grounded faithfulness (RQ2). Third, the research evaluates the theoretical and practical advantages of cache-augmented generation (CAG) over standard retrieval-augmented baselines, specifically determining whether establishing a lossless, pre-encoded context window mitigates the information bottleneck and selection errors inherent in top-k retrieval (RQ3). Fourth, the thesis assesses the auditability of model outputs at the granular level, asking whether word-level Shapley Additive exPlanations (SHAP) values and log-probability visualizations can reliably distinguish between evidence-based generation and parametric hallucination (RQ4). Finally, the study examines the cross-domain robustness of the proposed architecture, testing its ability to generalize from structured statutory texts to the noisy, narrative structure of judicial decisions without additional task-specific adaptation (RQ5).

In addressing these questions, this thesis makes several distinct scientific contributions to the field of natural language processing (NLP) and reliable model deployment. The primary contribution is the development of a structured Turkish Regulatory Corpus, which introduces a novel, semantically segmented dataset of the Turkish Fire-Safety Regulation paired with a curated QA testbed, filling a foundational gap for low-resource legal research. Secondly, the thesis provides a rigorous comparative analysis of retrieval strategies, demonstrating that for technical, morphologically rich text, hybrid lexical approaches significantly outperform dense retrievers that suffer from vocabulary mismatch. Thirdly, it validates an end-to-end regulation-grounded CAG architecture that utilizes parameter-efficient adaptation via quantized low-rank adaptation (QLoRA) to enforce strict adherence to statutory language. This architecture is shown to effectively minimize specific failure modes, including the halluci-

nation of numeric safety thresholds. Fourthly, the work offers a detailed qualitative taxonomy of errors, systematically categorizing how tokenization artifacts in agglutinative languages contribute to downstream generation failures. Finally, the thesis advances the domain of explainable artificial intelligence (XAI) by adapting token-level attribution tools to regulatory tasks, establishing a method for visually auditing the alignment between generated answers and their retrieved legal context.

The remainder of this thesis is organized as follows. Chapter 2 reviews background and related work on LLMs, tokenization and representation for Turkish, XAI, and legal and regulatory decision-support systems. Chapter 3 presents the methodology: corpus construction, retrieval configurations, parameter-efficient fine-tuning, CAG, and the evaluation and explainability toolkit. Chapter 4 reports experimental results on regulation QA and court–to–regulation linking, covering retrieval quality, answer faithfulness, efficiency, and explanation analyses. Chapter 5 concludes with a summary of findings, a discussion of limitations, and directions for future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Foundations of Large Language Models

2.1.1 Probability Modeling and Sequential Prediction

The probabilistic view of language modeling originates in information theory, where next-symbol prediction and linguistic predictability were first formalized [1]. Early approaches relied on count-based n-gram models with Markov assumptions, smoothing, and back-off to estimate conditional word probabilities, but these suffered from data sparsity and short-context limitations [2, 3]. Neural language models addressed these issues by learning distributed representations and continuous conditional probability functions over word sequences [4, 5]. Subsequent advances in representation learning and sequence modeling, including distributed word embeddings and deep autoregressive architectures, contributed to the large-scale pretraining of generative language models that underpin contemporary LLMs [6, 7].

Formally, let $x_{0:m} = (x_0, \ldots, x_m)$ denote a token sequence of length $m+1$ with $x_t \in \mathcal{V}$ for vocabulary $\mathcal{V}$, and define the prefix shorthand $x_{<t} = (x_0, \ldots, x_{t-1})$. An autoregressive language model with parameters θ defines a joint distribution via the chain rule:

$$p_\theta(x_{0:m}) = p_\theta(x_0) \prod_{t=1}^{m} p_\theta(x_t \mid x_{<t}). \tag{2.1}$$

In modern LLMs, $\mathcal{V}$ is typically a finite, learned subword inventory supporting open-vocabulary modeling. After extensive pretraining on large corpora, such models capture rich syntactic and semantic regularities in natural language [8].

At inference, greedy next-token generation selects the vocabulary element maximizing the model's conditional probability given the preceding context:

$$\hat{x}_t \;=\; \arg\max_{v \in \mathcal{V}} \; p_\theta(x_t = v \mid x_{<t})\,, \tag{2.2}$$

or alternatively samples from $p_\theta(\cdot \mid x_{<t})$ when diversity is required. Discrete tokens are mapped to dense vector representations grounded in the distributional hypothesis [9, 10], enabling generalization in a continuous space.

In modern implementations, $p_\theta(x_t \mid x_{<t})$ is a categorical distribution over $\mathcal{V}$ obtained via a softmax over network logits. Training proceeds by maximum likelihood; given a corpus $\mathcal{D} = \{x^{(i)}_{0:m_i}\}_{i=1}^{N}$ of variable-length sequences, the negative log-likelihood objective is:

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = -\sum_{i=1}^{N}\Big[\log p_\theta\Big(x_0^{(i)}\Big) + \sum_{t=1}^{m_i} \log p_\theta\Big(x_t^{(i)} \mid x_{<t}^{(i)}\Big)\Big], \tag{2.3}$$

which instantiates Eq. (2.1) and is optimized using teacher forcing (conditioning on the true prefix $x_{<t}$ during training). From an information-theoretic perspective, minimizing $\mathcal{L}_{\mathrm{NLL}}$ corresponds to minimizing the empirical cross-entropy between the data distribution and the model; in expectation:

$$\mathbb{E}_{x_{0:m}\sim p_{\text{data}}}[-\log p_\theta(x_{0:m})] = H(p_{\text{data}}) + D_{\mathrm{KL}}(p_{\text{data}} \,\|\, p_\theta)\,, \tag{2.4}$$

where $H(p) = \mathbb{E}_{x\sim p}[-\log p(x)]$ is the Shannon entropy. Teacher forcing estimates gradients of the maximum-likelihood objective under the empirical data distribution, whereas at inference the model conditions on its own generated history; this train–test mismatch may compound errors during long-horizon decoding [11].

The vocabulary $\mathcal{V}$ is typically constructed using subword tokenization to support open-vocabulary modeling and mitigate sparsity relative to purely word-level vocabularies, which is particularly relevant in morphologically rich languages. In practice, reported performance improvements for next-token prediction and downstream tasks depend not only on model capacity and training resources, but also on data composition, optimization choices, and, critically, the evaluation protocol and reporting conventions [12].

2.1.2 Transformer Architecture

The Transformer architecture, originally proposed as an encoder–decoder framework [13], converts input token sequences into deeply contextualized representations within the encoder and generates coherent output sequences through the decoder. By leveraging self-attention, Transformers avoid strictly sequential computation during training, enabling direct modeling of long-range dependencies and parallelization across sequence positions; together with residual connections and normalization, these design choices also improve gradient flow in deep networks and enable stable optimization at scale.

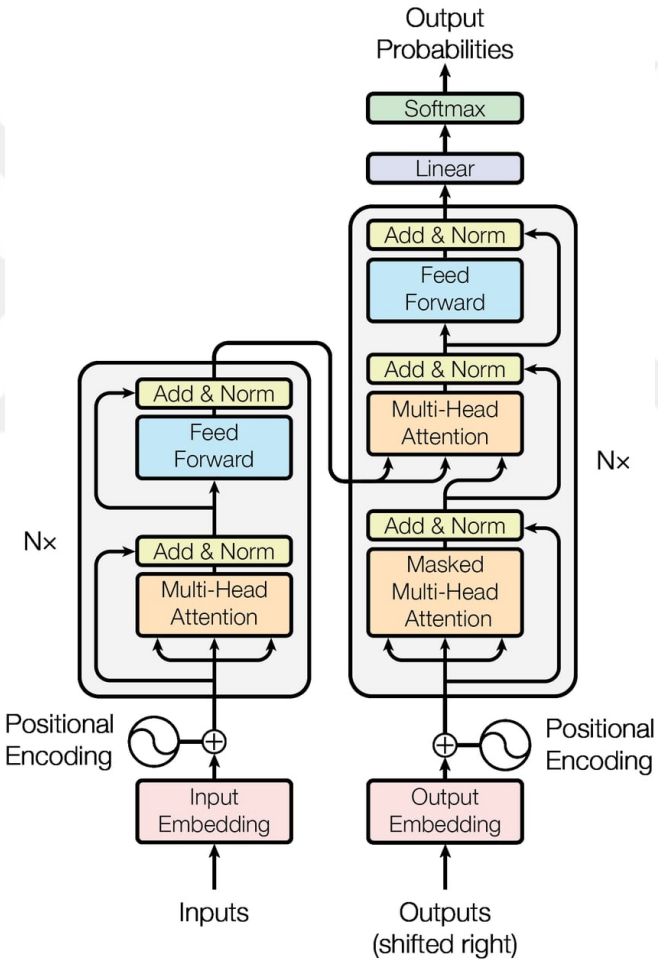


Figure 2.1: The Transformer model architecture (adapted from [13]).

An architectural overview is provided in Fig. 2.1, which situates self-attention within the encoder/decoder stacks before the attention computation is formalized in Eq. (2.5).

Self-attention maps token embeddings to query (Q), key (K), and value (V) vectors; attention scores are scaled dot products of Q and K and converted to weights via softmax, which then aggregate V to yield context-dependent representations:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V. \tag{2.5}$$

Here $Q, K \in \mathbb{R}^{n\times d_k}$ and $V \in \mathbb{R}^{n\times d_v}$ for sequence length n. Multi-head attention uses h parallel heads; with model width d_{model}, each head typically uses $d_k = d_v = d_{\text{model}}/h$, so that $h\,d_k = h\,d_v = d_{\text{model}}$. Learned projections map model states to per-head subspaces, $W_Q, W_K \in \mathbb{R}^{d_{\text{model}}\times(hd_k)}$ and $W_V \in \mathbb{R}^{d_{\text{model}}\times(hd_v)}$, and the resulting tensors are reshaped into h heads and concatenated after attention, enabling parallel, heterogeneous dependency modeling. This operation provides content-addressable conditioning: each query token selectively aggregates information from its keys/values, supporting the conditional factors in Eq. (2.1) with differentiable, long-range context. The attention mechanism acts as a dynamic weighting function, allowing the model to assign non-uniform importance to different parts of the conditioning history rather than assuming a fixed-window Markov property. The scaling by $\sqrt{d_k}$ controls the variance of $QK^\top$, preventing softmax saturation and stabilizing optimization at depth.

Since self-attention is permutation-invariant, positional information is injected by adding positional embeddings to token embeddings:

$$e_i = w_i + PE(i), \tag{2.6}$$

where $w_i \in \mathbb{R}^{d_{\text{model}}}$ denotes the token embedding (lookup of the discrete token $x_i \in \mathcal{V}$) and $PE(i) \in \mathbb{R}^{d_{\text{model}}}$ is the positional embedding. To generalize better beyond training sequence lengths, relative positional encodings introduce distance-dependent biases into attention [14], and rotary positional embeddings (RoPE) implement position as a rotation in embedding space, improving extrapolation and long-range generalization [15]. Absolute encodings tie representations to fixed indices, whereas relative and rotary schemes parameterize distances (or phases), preserving translation-invariant inductive biases for dependencies and supporting out-of-length generalization. Stacking multi-head self-attention with position-wise feed-forward layers, residual connections, and normalization yields a scalable block that can be arranged purely in a causal

(masked) configuration; this causalization enables decoder-only architectures to implement the autoregressive objective.

Concretely, the causal mask prevents each position t from attending to future tokens, ensuring that the representation at t depends only on the prefix $x_{<t}$, consistent with Eq. (2.1); during training this preserves parallel computation across positions, while at inference it enables key–value caching to amortize attention over time steps by reusing prior states, improving decoding efficiency for long prefixes.

2.1.3 Decoder-Only LLMs, Scaling, and Efficiency

A standard Transformer block couples multi-head self-attention with a per-token feed-forward network, residual connections, and normalization [16]. Stacking these blocks yields encoder and decoder modules: encoders apply self-attention over the full input, whereas decoders apply masked self-attention for autoregression and, in sequence-to-sequence settings, cross-attention to encoder states. Contemporary LLMs specialize this architecture into stacked decoder-only Transformers that autoregressively predict the next token given all prior tokens, are trained to minimize cross-entropy on very large text corpora, and exhibit strong performance on diverse open-ended tasks such as completion, summarization, and question answering [17]. Causal masking enforces the conditional independence pattern of Eq. (2.1), aligning the model's receptive field with the autoregressive factorization, while the softmax output layer provides normalized token likelihoods that support both greedy next-token selection (Eq. (2.2)) and sampling-based decoding policies. Naïvely, (masked) self-attention incurs $O(n^2)$ time and memory in the sequence length per layer, with additional linear dependence on the representation size; practical implementations therefore rely on optimized attention kernels and caching of key–value states for attention from previous time steps to make long-context generation feasible at inference time. While training can be parallelized across sequence positions, autoregressive decoding remains sequential across time steps, with parallel computation occurring within each decoding step.

2.2 Tokenization and Language Adaptation for Turkish

2.2.1 Morphological Complexity in Turkish

Turkish is an agglutinative language with highly productive suffixation, yielding long, information-dense word forms and substantial type–token variation. This typology creates data sparsity and complicates disambiguation for word-level models; robust processing therefore requires exposing sublexical structure and handling rich morphosyntactic features [18]. Two-level finite-state analyses and downstream parsing studies consistently show that stems and concatenative affixes encode much of the grammatical information, and that modeling these units explicitly improves generalization in syntactic and semantic tasks [19]. In multilingual pretraining, shared subword vocabularies disproportionately allocate capacity to high-resource languages; for Turkish this often results in over-segmentation (excess byte- or character-like fragments) and longer sequences for equivalent content, with potential implications for efficiency and downstream performance [20].

2.2.2 Subword and Neural Tokenization Approaches

Data-driven subword tokenization enables open-vocabulary language modeling by learning compact sets of frequent character n-grams. Byte-Pair Encoding (BPE) iteratively merges the most frequent symbol pairs, reducing out-of-vocabulary problems in neural machine translation while learning data-driven subword units that may correlate with recurring morphological patterns [21]. WordPiece, introduced for Japanese and Korean voice search, constructs subword vocabularies by greedily adding units that maximize the likelihood of the training corpus under a language model [22]. SentencePiece provides a language-independent tokenizer that operates directly on raw text, supports both BPE and unigram models, and is widely used in large-scale pretraining pipelines for morphologically rich languages [23].

Purely statistical segmenters, however, are morphology-agnostic: in agglutinative languages they may ignore affix boundaries, fragment semantically coherent morphemes, and lengthen sequences. Empirical evaluations on morphologically rich lan-

guages show that segmentation strategies which better respect morphological structure can improve lexical consistency and downstream neural machine translation performance compared to unconstrained BPE-style baselines [24]. For Turkish specifically, tokenizer choice has been shown to materially affect sequence length, perplexity, and language modeling quality [25]. Effective Turkish adaptation often benefits from coordinated choices about tokenizer design (e.g., vocabulary size, normalization, reserved symbols) and from continued or from-scratch pretraining on Turkish corpora [26].

2.2.3 Domain-Specific Challenges and Evaluation

Legal and regulatory corpora contain dense terminology, formal citation patterns (e.g., article and paragraph identifiers, docket numbers), and highly formulaic phrasing. When applied to such data, general-purpose tokenizers tend to over-segment key legal expressions and identifiers, inflating sequence length and fragmenting semantically coherent units. Evidence from legal-domain transformer models such as LEGAL-BERT indicates that continued pretraining on large-scale legal corpora substantially improves performance on tasks such as statute classification and legal entailment, highlighting the importance of aligning model pretraining—and, where possible, tokenization—with the structure of legal text [27].

Recent analyses of numerical reasoning in language models show that even state-of-the-art systems exhibit systematic weaknesses across representation, number sense, manipulation, and complex reasoning, and may rely on dataset artefacts rather than robust numerical inference [28]. For safety-critical regulatory domains such as building fire standards, these limitations intersect with tokenization challenges around numerical and physical constraints: subword tokenizers can split decimals, thresholds, and units into disjoint pieces, which may weaken the coupling between magnitudes and their measurement scales. In bounded regulatory corpora, reliability further depends on verifiable grounding: answers should be traceable to the governing provision, and numeric constraints should be reproduced without distortion. These requirements make retrieval quality and evidence usage central to the practical safety of regulation-grounded decision support.

2.3 Explainable Artificial Intelligence in Language Models

2.3.1 Evolution of Explainability in High-Capacity Models

As language models have increased in scale and application scope, predictive performance has often improved at the cost of transparency. This has motivated an interpretability agenda that formulates desiderata such as fidelity, stability, and usefulness, and proposes corresponding evaluation protocols [29]. Complementary overviews systematize families of interpretability methods, highlight their assumptions and limitations, and emphasize the need to align explanation tools with concrete use cases and stakeholder needs [30].

In parallel, legal and policy analyses stress that transparency is central to accountability in automated decision-making. Detailed examination of the General Data Protection Regulation (GDPR) argues that it does not guarantee a broad, unconditional "right to explanation," while still imposing obligations that push systems toward forms of auditability and intelligibility in practice [31]. Recent work on trust and transparency further suggests that explanations must be calibrated: insufficient detail undermines oversight, whereas excessive or poorly structured detail can overwhelm users and erode meaningful trust [32]. Within this landscape, XAI is positioned as a practical mechanism for scrutiny and audit in high-stakes contexts.

2.3.2 Model-Agnostic and Model-Specific Techniques

Explainability methods for neural models are commonly divided into model-agnostic and model-specific approaches. Model-agnostic techniques, such as perturbation-based analyses and Shapley-value attributions, approximate local decision behavior without requiring access to internal parameters. Shapley values provide an axiomatic framework for distributing the prediction "payout" among feature coalitions, yielding a principled allocation of marginal contributions. Model-specific techniques, including gradient-based saliency and influence-function analysis, exploit architectural details and internal representations to trace how inputs influence outputs.

In high-stakes applications, concerns have been raised that post-hoc explanations of

opaque models may be unreliable or misleading, particularly when end-users treat them as faithful accounts of underlying reasoning. A contrasting perspective argues that, wherever feasible, inherently interpretable models with transparent structure and human-understandable decision rules should replace black-box predictors in sensitive domains, motivating evaluation regimes that consider not only explanation overlays but also the interpretability of the underlying decision process [33].

2.3.3 Explainability for LLMs and Safety-Critical/Legal Contexts

LLMs introduce additional challenges for explainability: predictions may depend on long-range context, subtle prompt variations, and internal mechanisms that are difficult to localize. Recent surveys organize techniques such as token-level attribution, attention-pattern analysis, counterfactual probing, and explanations of tool use, while highlighting open problems in faithfulness, robustness, and user-centered evaluation [34, 35, 36].

Empirical studies caution that raw attention weights, taken in isolation, are not reliable explanations of model reasoning, and advocate causal and perturbation-based tests to assess whether purported explanations track the computations that produce predictions [37, 38]. This reliability gap is exacerbated in agglutinative contexts where subword tokenizers fragment semantic units (Section 2.2.3), dispersing attention across sub-tokens that may not correspond to meaningful entities. In legal and safety-critical decision support, explainability therefore must connect outputs to relevant textual evidence in a way that enables external verification. Early work on predicting court-case outcomes from textual descriptions illustrates both the promise of such systems and the need for explanations aligned with established reasoning patterns [39].

2.4 LLM-Powered Decision Support in Legal and Safety-Critical Domains

2.4.1 LLMs in Disaster and Risk Management

LLMs are increasingly explored for crisis decision support, where they summarize heterogeneous inputs (situation reports, social media, sensor-derived narratives) and

propose candidate actions for human planners. One line of work uses in-context learning and domain guidelines to generate structured plans of action for a disaster scenario; human operators then refine these options, reducing the time needed to arrive at an operationally acceptable plan [40]. Complementary platforms integrate LLMs into end-to-end crisis-management workflows, triaging citizen reports and social-media posts, classifying messages by urgency and topic, and producing consolidated situation overviews for emergency managers [41].

LLM-based systems are also combined with retrieval and multi-agent coordination for natural hazard decision support. RAG-driven architectures over hazard models, weather projections, and technical documentation aim to ground recommendations in retrieved domain evidence, improving verifiability when the cited context is relevant and correctly used [42]. Surveys emphasize that such systems should be treated as components in a broader socio-technical pipeline, with explicit attention to reliability, robustness, and transparent interfaces rather than stand-alone "black-box" oracles [43]. In parallel, safety-engineering work argues for integrating hazard analysis into the development lifecycle of ML-powered systems, and shows how LLMs can partially automate methods such as System-Theoretic Process Analysis (STPA) while keeping humans in the loop at critical steps; this framing treats hazard identification and control design as a proactive workflow, generating pre-deployment "requirements for explanation" [44].

2.4.2 Legal and Regulatory Decision-Support

In legal and regulatory settings, decision-support systems must not only predict outcomes but also articulate why a recommendation follows from authoritative sources. Explainable tools for case-based legal reasoning show that model outputs can be accompanied by structured justifications aligned with case facts and legal argument patterns, helping practitioners inspect which textual elements support particular conclusions and making the reasoning process auditable [45]. For lay users, augmented-intelligence tools based on interactive decision pathways encode statutory and procedural rules into question–answer flows; users are guided through legally grounded questions whose branches correspond to conditions in the underlying rules, and rec-

ommendations are tied back to relevant provisions in an accessible form [46].

In the contract and compliance domain, expert-annotated datasets for legal contract review provide clause-level labels and span annotations for risk-relevant phenomena (e.g., limitation of liability, unilateral termination, non-standard confidentiality); these resources enable models that not only flag potentially problematic clauses but also highlight the specific evidence triggering the alert [47]. Together, such systems illustrate that in high-stakes legal environments, predictive accuracy is insufficient on its own; effective decision support requires traceable, document-grounded justification that supports expert audit.

2.5 Research Gaps and Theoretical Motivation

While LLMs have demonstrated significant capability in general-purpose tasks, their application to safety-critical, low-resource legal domains remains constrained by specific theoretical and empirical limitations. First, retrieval for agglutinative languages like Turkish presents a distinct challenge: surface-form variation induced by productive suffixation can complicate both lexical matching and semantic similarity estimation when retrieving tightly scoped regulatory provisions. In regulatory contexts where suffix variation can alter the normative status of a provision, there is limited empirical evidence comparing the efficacy of dense retrieval against sparse, morphology-aware lexical methods for retrieving specific, numerically constrained statute provisions.

Second, standard RAG relies on a top-k selection step, which introduces a probabilistic bottleneck: if the governing provision is not retrieved within the cutoff, the generation process is fundamentally compromised. For bounded, authoritative corpora such as fire safety regulations, the theoretical potential of CAG, which can preload relevant context into the model's inference context to mitigate selection error and reduce inference latency, has been less commonly studied in the specific setting of bounded, authoritative regulatory corpora used for decision support.

Third, explainability in legal AI often emphasizes the plausibility of generated rationales rather than their faithfulness to the underlying source text. Existing frameworks rarely provide granular, word-level attribution that enables auditors to mechanically

verify whether a model's output is parametrically hallucinated or explicitly driven by the retrieved regulatory tokens.

This thesis investigates these gaps through a scoped empirical study. To address the retrieval challenges associated with Turkish morphology, it evaluates a hybrid retrieval module that explicitly contrasts surface and morphologically normalized signals with dense retrieval representations. To examine the efficiency–faithfulness trade-off, this module is integrated into a CAG pipeline, measuring how pre-encoded context affects both latency and answer stability compared to standard prompting. Finally, to improve auditability, the study applies word-level explanation techniques to trace model decisions back to specific terms in the regulation, offering a reproducible approach for verifying evidence-grounded legal advice.

Overall, these foundations motivate the thesis design choices: retrieval strategies tailored to Turkish regulatory language, an efficient cache-augmented generation pipeline to improve stability and latency, and word-level attribution to support auditability of evidence-grounded answers.

CHAPTER 3

METHODOLOGY

3.1 Overview

This chapter formalizes the study's methodology by specifying the modeling assumptions, the theoretical basis of the CAG procedure, and the experimental environment, ensuring that every result in Chapter 4 is anchored to a precisely defined object. In particular, we consider four candidate generative models — Llama-3-8B-Instruct, Qwen2-7B-Instruct, Mistral-7B-Instruct-v0.3 and Gemma-2-9B-IT— and evaluate them on our QA tasks in a zero-shot setting. The model yielding the highest baseline accuracy is then selected for the fine-tuning and CAG phases. By focusing our comparisons on this selection stage, we avoid extensive model-versus-model evaluation in the main experiments. All subsequent experiments (fine-tuning, CAG, and interpretability analysis) use the chosen model as the backbone.

3.2 Theoretical Foundations and Model Architecture

3.2.1 Preliminaries and Scaling Guidance

The methodology is guided by empirical scaling analyses that jointly consider model size, fine-tuning data volume, and compute budget. Joint scaling results for LLM fine-tuning show that performance can be well-approximated by a multiplicative law over fine-tuning data size and other scaling factors, and that, under a fixed compute envelope, increasing backbone size yields systematically larger gains than either enlarging the pretraining corpus or scaling the number of parameter–efficient adaptation parameters; moreover, scaling LoRA-style heads beyond moderate rank brings only

limited additional benefit while remaining relatively stable among PEFT methods [48]. In this thesis, these findings justify concentrating compute on a single mid-scale instruction-tuned decoder-only backbone in the 7–9B range, loaded in quantized form and equipped with relatively low-rank LoRA adapters, rather than distributing resources across many smaller backbones or heavily overparameterized adapters.

In data-limited settings, rectified scaling analyses model fine-tuning behaviour as a two-phase trajectory, with an initial "pre-power" region followed by a more predictable power-law regime, and introduce an effective pre-learned data size that can be estimated from small-budget runs to compare models without fully fine-tuning every candidate at maximal scale [49]. Complementary multilingual experiments on LLM fine-tuning for in-house translation show that very small supervised corpora on the order of 10^3 examples can degrade performance relative to strong zero-shot baselines, whereas translation quality improves and stabilizes once moderately sized, well-mixed domain corpora are used [50]. Consistent with this evidence, the present study (i) restricts supervised adaptation to a carefully constructed corpus of 1,431 regulation-based question–answer pairs (951 train, the remainder split across validation and test), and (ii) employs a compute-aware model-selection step: four instruction-tuned backbones are first compared under a uniform zero-shot protocol on Turkish fire-safety QA, and only the strongest candidate is carried forward as the sole backbone for supervised fine-tuning and cache-augmented generation.

Transfer-scaling analyses for machine translation further show that downstream performance depends not only on scale but also on the distributional alignment between pretraining data and downstream supervision: when pretraining and task data are well-aligned, both downstream loss and task metrics improve monotonically and admit simple log- or power-law approximations, whereas moderate misalignment can cause task metrics to fluctuate or even degrade despite monotonically improving loss [51]. This motivates restricting supervised signals in this thesis to question–answer pairs derived from the official Turkish fire-safety regulation, its implementation guide, and a curated set of Turkish high-court decisions (Yargıtay, Danıştay, AYM) concerning regulation-compliant and non-compliant scenarios, rather than mixing heterogeneous general-purpose Turkish text.

The design of the retrieval and cache layers in the CAG pipeline is likewise informed by empirical results on context sensitivity: conditioning LLM predictions on retrieved passages has been shown to substantially alter factual outputs in cloze-style and QA tasks, with improvements when context is semantically aligned with the query and degradation when irrelevant or noisy text is injected [52]. In this thesis, retrieval is therefore treated as a separate module with task-appropriate tokenization, embeddings, and reranking that are decoupled from the generator's tokenizer. This allows the system to optimize context selection over the full regulatory corpus and extended court-decision collection—both of which exceed the context window of any single model invocation—while keeping the generative backbone fixed and accessing only those segments that are most informative for each query.

Finally, alignment studies indicate that the bulk of an instruction-tuned model's knowledge is acquired during pretraining and that high-quality behavioral alignment on complex tasks can be achieved with a relatively small number of carefully curated prompt–response pairs [53]. This supports the choice to prioritize legal fidelity, coverage of numerically constrained clauses, and diversity of regulatory scenarios in constructing the 1,431-item QA corpus, rather than attempting to collect or synthesize orders of magnitude more supervision. Combined with parameter-efficient adaptation on a strong instruction-tuned backbone, these scaling and alignment considerations provide a principled justification for (i) selecting a single mid-scale generator via zero-shot screening, (ii) operating in a moderate but carefully curated data regime, and (iii) decoupling retrieval from generation within the CAG framework used in the remainder of the methodology.

3.2.2 Model Candidates and Uniform Screening Setup

Four instruction-tuned decoder-only language models in the 7–9B parameter range are evaluated under a uniform zero-shot protocol: *Llama-3-8B-Instruct* [54], *Qwen2-7B-Instruct* [55], *Mistral-7B-Instruct-v0.3* [56], and *Gemma-2-9B-IT* [57]. All candidates are loaded in a quantized configuration to ensure comparable memory footprints, with deterministic computation enabled and a shared random seed applied across the full pipeline. Tokenization settings are standardized by applying left padding

and, when necessary, assigning the end-of-sequence symbol as the padding token. Each model's maximum context length is taken directly from its configuration to ensure that generation remains within its native limit.

A task-specific Turkish instruction prompt is used across all models. This prompt explicitly states that the question is based on the *Binaların Yangından Korunması Hakkında Yönetmelik*, requires answers in clear and correct Turkish, and mandates fidelity to the regulation's technical and numerical details. The raw question is appended to this prefix and introduced with the marker "Soru:". When a model provides a chat template, the combined instruction and question are embedded into it; otherwise, they are used as plain text. This ensures a consistent legal grounding while respecting each model's expected formatting style.

To avoid truncation and maintain comparability, each model receives an answer-length budget derived from the 99th percentile of reference answer lengths in the training and validation splits. This value is scaled and clipped to a fixed range, then enforced dynamically according to the remaining available context window after encoding the input prompt. Decoding is deterministic across all candidates, using the same set of simple constraints that discourage repetitive loops while preserving domain-relevant phrasing. After generation, all outputs are cleaned to remove special markers and template residues, with non-substantive or empty outputs mapped to a dedicated placeholder to ensure consistent downstream handling.

Because the four models differ in memory demands, screening is carried out with an adaptive strategy that determines the largest feasible batch size for each candidate and automatically reduces resource usage only when necessary. Each model is evaluated on the same held-out test set, and complete per-model artefacts are recorded, including generated answers and run configurations. The model achieving the strongest overall zero-shot performance under the evaluation metrics defined in Section 3.5 is selected as the backbone for all subsequent stages—supervised fine-tuning, cache-augmented generation, and explainability.

As supplementary context, external assessments of ethical behaviour in Turkish indicate that Gemma and Qwen families exhibit some of the strongest safety, fairness, and robustness characteristics among recent open-source models [58]. Including *Gemma-*

2-9B-IT and *Qwen2-7B-Instruct* in the candidate pool therefore aligns both with task-specific needs in Turkish legal question answering and with broader evidence of responsible model behaviour.

3.3 Parameter-Efficient Fine-Tuning

Fine-tuning was conducted using parameter-efficient approaches rather than full-model updates, motivated by computational and memory constraints. We conceptualize Parameter-Efficient Fine-Tuning (PEFT) as a family of approaches that freeze the pretrained backbone while updating only lightweight, task-specific modules. Typical designs include adapter layers inserted within attention and feed-forward blocks or low-rank decompositions that approximate weight updates. This strategy reduces the number of trainable parameters and optimizer states by orders of magnitude, enhances stability under data-constrained conditions, and facilitates modular "swap-in" specialization across tasks [59].

3.3.1 Low Rank Adaptation

In a transformer, many computations are linear projections: for an input row-vector $x \in \mathbb{R}^{1\times d_{\text{in}}}$ and a weight matrix $W \in \mathbb{R}^{d_{\text{in}}\times d_{\text{out}}}$, the layer output is

$$y = xW.$$

These projections appear in attention (q, k, v, o) and in the MLP (gate, up, down). Instead of changing W (which is large), we freeze W and learn only a small correction ΔW that is low rank. "Low rank" means ΔW is factored through a tiny bottleneck of dimension $r \ll \min(d_{\text{in}}, d_{\text{out}})$:

$$\Delta W = s\, BA, \quad B \in \mathbb{R}^{d_{\text{in}}\times r}, \quad A \in \mathbb{R}^{r\times d_{\text{out}}}, \quad s = \frac{\alpha}{r}.$$

- B compresses the input from d_{in} to r.
- A expands it back from r to d_{out}.
- $s = \alpha/r$ is a scaling that keeps the update magnitude roughly stable as you change the rank r (larger $r \Rightarrow$ smaller per-direction scale).

The forward map becomes the frozen base output plus a learned, low-rank residual:

$$y = x(W + \Delta W) = xW + \frac{\alpha}{r}\, xBA \tag{3.1}$$

So the model behaves like the original pretrained network, plus a small trainable add-on that is routed through the r-dimensional bottleneck.

Empirically, LoRA matches or exceeds full fine-tuning on large models while cutting memory by orders of magnitude, motivating its use under tight VRAM budgets. We attach LoRA adapters to both attention projections and MLP projections. Distributing adapters across all linear sublayers improves fidelity to full fine-tuning at a negligible memory cost, consistent with findings that performance hinges more on *where* adapters are placed than on modest changes to rank r [60] .

3.3.2 Quantization

We apply QLoRA-style adapter fine-tuning over a *frozen 8-bit* quantized backbone to fit larger models on a single GPU while keeping matrix multiplications in BF16. Each weight tensor W is stored block-wise in $\mathrm{Int8}$ with per-block affine parameters (s, z); at run time we dequantize to $\mathrm{BF16}$ for compute, while LoRA adapters and optimizer states remain in $\mathrm{BF16}$. Paged optimizers are used to prevent out-of-memory spikes for long sequences, following the QLoRA memory-management design [61].

For a linear layer with a single LoRA adapter, the forward pass combines the dequantized base and the low-rank update:

$$Y_{\mathrm{BF16}} \;=\; X_{\mathrm{BF16}}\, \mathrm{dequant}\big(W_{\mathrm{Int8}}; s, z\big) \;+\; X_{\mathrm{BF16}}\, L_1 L_2, \tag{3.2}$$

where $L_1 \in \mathbb{R}^{h \times r}$ and $L_2 \in \mathbb{R}^{r \times o}$ are the only trainable parameters (the backbone is frozen). The dequantization operator is the standard affine mapping applied per block:

$$\mathrm{dequant}\big(W_{\mathrm{Int8}}; s, z\big) \;=\; s \odot \big(W_{\mathrm{Int8}} - z\big), \tag{3.3}$$

with s the block scale, z the (optional) block zero-point, and $\odot$ denoting elementwise multiplication. This 8-bit instantiation preserves the QLoRA principle of learning

small low-rank adapters on top of a quantized base while offering a straightforward calibration path and stable BF16 computation.

3.4 Cache-Augmented Generation

CAG is our primary inference regime for Turkish legal decision support. In the pure CAG limit, the system pre-encodes a fixed corpus prefix into a reusable key–value (KV) cache, eliminating per-query retrieval latency and selection error only when the full evidence set fits within the model's context budget. We hold the generator M_θ and its tokenizer fixed; performance is controlled via corpus formatting and cache-building choices. We evaluate effects on latency, answer quality, and faithfulness (evidence–attribution overlap). Because the full regulation corpus exceeds the model context budget in our setting, our experiments primarily use a hybrid variant (CAG+retrieval) in which only a retrieved evidence subset is cached per query. For completeness, we first formalize the pure CAG limit and then specify the hybrid instantiation used in this thesis.

Pure CAG (limiting case) Let a long-context decoder-only LLM M_θ with maximum context length $L_{\max}$ and tokenizer $\tau(\cdot)$ be given. Let $D = \{d_1, \ldots, d_N\}$ denote a fixed bounded knowledge corpus. Define the offline preloading (cache build) as concatenating the corpus into a prefix

$$P = \mathrm{format}(d_1, \ldots, d_N) \quad \text{with} \quad |\tau(P)| \leq L_{\max}, \tag{3.4}$$

and computing the per-layer key–value tensors (KV-cache) for that prefix:

$$C_{\mathrm{KV}} = \mathrm{KV\text{-}Encode}_\theta(P). \tag{3.5}$$

During inference for a query q, the model conditions on C_{KV} without re-encoding P:

$$\hat{y} = \arg\max_y p_\theta(y \mid q; C_{\mathrm{KV}}), \qquad p_\theta(y_t \mid y_{<t}, q; C_{\mathrm{KV}}) = \mathrm{softmax}(W_o h_t), \tag{3.6}$$

where the attention states at each layer use cached keys/values via concatenation.

Let T_{build} be the one-time cost of building $C_{\mathrm{KV}} = \mathrm{KV\text{-}Encode}_\theta(P)$ (Eq. (3.5)). Let $T_{\mathrm{gen}}(q)$ denote the decoding cost for query q with cache reuse. For a session with

queries $\{q_j\}_{j=1}^{m}$,

$$T_{\mathrm{CAG}}(\{q_j\}) = T_{\mathrm{build}} + \sum_{j=1}^{m} T_{\mathrm{gen}}(q_j), \qquad \bar{T}_{\mathrm{CAG}} = \frac{T_{\mathrm{build}}}{m} + \mathbb{E}[T_{\mathrm{gen}}(q)]. \tag{3.7}$$

Thus, the amortized per-query cost decreases with session length m. In contrast, retrieval-augmented generation adds a per-query retrieval term $T_{\mathrm{ret}}(q)$; pure CAG asymptotically removes this overhead.

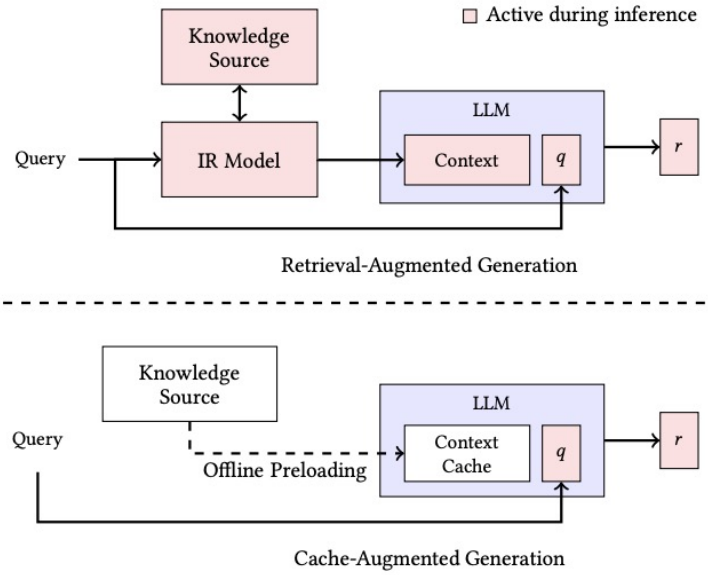


Figure 3.1: Comparison of RAG and CAG workflows

In the pure CAG limit, if the relevant evidence set $\mathcal{R}(q) \subseteq D$ fits in the cached prefix P (i.e., $|\tau(P)| \leq L_{\max}$ and $\mathcal{R}(q) \subseteq P$), then CAG achieves perfect retrieval recall by construction (no selection error), and generation proceeds with holistic context.

Hybrid CAG (CAG+retrieval; used in this thesis) The perfect-recall property above applies only to the pure CAG limit. In our setting, the full regulation corpus does not fit within the model context budget. Therefore, for each query q we first retrieve a bounded, query-dependent evidence subset

$$R_k(q) = \mathrm{Retrieve}_k(q; D), \tag{3.8}$$

where $R_k(q) \subseteq D$ denotes the top-k retrieved items (e.g., section-level regulation

snippets). We then cache only the corresponding prefix

$$P^*(q) = \text{format}(R_k(q)), \qquad C_{\text{KV}}(q) = \text{KV-Encode}_\theta(P^*(q)). \tag{3.9}$$

Selection error occurs whenever the true relevant evidence set $\mathcal{R}(q)$ is not fully covered by the retrieved set $R_k(q)$, i.e., when $\mathcal{R}(q) \nsubseteq R_k(q)$. Consequently, hybrid CAG preserves retrieval as the mechanism for evidence selection and inherits its recall limitations, while cache augmentation materializes the retrieved evidence as an explicit KV prefix and reuses it throughout decoding.

To unify notation for cache maintenance, let P_{cache} denote the cached prefix: $P_{\text{cache}} = P$ in the pure CAG limit and $P_{\text{cache}} = P^*(q)$ in the hybrid setting. Practical operation requires (i) a bounded, query-dependent evidence subset (e.g., top-k retrieved sections), (ii) stable formatting to ensure deterministic KV builds, and (iii) cache reset by truncation after each answer to maintain a clean prefix:

$$C_{\text{KV}}(q) \leftarrow \text{truncate}(C_{\text{KV}}(q),\ |\tau(P_{\text{cache}})|)\,. \tag{3.10}$$

In summary, pure CAG corresponds to the mapping $(D, q) \mapsto \hat{y}$ with

$$\hat{y} = M_\theta(q \mid \text{KV-Encode}_\theta(\text{format}(D)))\,, \tag{3.11}$$

whereas the hybrid instantiation used in this thesis is

$$\hat{y} = M_\theta(q \mid \text{KV-Encode}_\theta(\text{format}(R_k(q))))\,, \tag{3.12}$$

eliminating repeated re-encoding of the retrieved evidence during decoding while preserving explicit retrieval as the mechanism for evidence selection.

In the main implementation, CAG is instantiated with a long-context Gemma–2–9B–IT model equipped with a QLoRA adapter trained on regulation-derived question–answer pairs. For each query, the retrieved section-aligned regulation snippets are formatted into $P^*(q)$, tokenized with the model's native tokenizer, and encoded once into a KV cache (HybridCache). The model then answers the query by reusing this cached evidence while only encoding the query and generated tokens; decoding hyperparameters and post-processing are held fixed across experiments, and only the retrieved evidence set and cache configuration vary.

We also study a second setting in which the cached evidence is extended to include regulation text together with a curated subset of court decisions (subject to the same token-budget constraints). Because this mixed evidence departs from the distribution used for QLoRA fine-tuning, we disable the adapter and use the base Gemma–2–9B–IT weights in this setting, keeping the cache-augmented mechanism (prefix construction, KV encoding, cache reuse, truncation) unchanged so that performance differences can be attributed primarily to the composition of the cached evidence rather than to changes in model parameters.

Whenever the concatenated legal evidence would exceed the context window, CAG is combined with retrieval by caching only a bounded top-k subset per query. Concretely, we implement and evaluate multiple retrieval variants over the same section-level corpus: (i) lexical BM25, (ii) a morphology-aware BM25 variant that operates on stem-like tokens, (iii) a supervised SimCSE-style dense retriever trained on question–context pairs, (iv) a learned hybrid fusion of lexical and dense signals, and (v) an optional fusion-plus-reranking variant with a cross-encoder. On a held-out test split, these methods are compared using standard information-retrieval metrics (MRR, Recall@k, nDCG@k); the best-performing variant is selected according to Recall@10 (with MRR as a tie-breaker) and then frozen as the default retriever in all CAG+retrieval experiments. Thus, pure CAG is a limiting case where the full corpus fits in the cache; in our setting, we apply hybrid CAG, caching the retrieved subset per query, and report retrieval metrics separately to quantify selection quality.

3.4.1 Retrieval Configuration

When the total length of the legal corpus (regulation only, or regulation plus court decisions) exceeds the model's context budget, cache-augmented generation is preceded by an explicit retrieval step over section-level segments. We compare five configurations. (i) A lexical BM25 retriever on surface-level Turkish tokens (BM25) using the Okapi scoring function [63]. (ii) A morph-aware variant (BM25+lemma), where a light, rule-based suffix stripping normalizes common Turkish inflectional endings so that inflected forms share a stem. (iii) A supervised dense retriever (SimCSE) in the spirit of SimCSE [64], initialised from cased BERTurk [65] and trained with a

triplet-loss objective on regulation QA pairs, using the gold section as positive and hard negatives mined from the lexical index. (iv) A hybrid linear Fusion model that z-score–normalizes BM25 surface scores, BM25 morph scores, and TF–IDF cosine similarities, optionally augmented with SimCSE similarities, and learns sparse fusion weights on the validation set via logistic regression, following hybrid lexical–neural retrieval practice [66]. (v) A Fusion+Rerank configuration that takes top candidates from Fusion and reranks them with a BERTurk-based cross-encoder scoring (question, segment) pairs [67].

Each retrieval configuration is evaluated offline, independently of the generator. For the regulation–QA dataset, every test question is paired with its gold regulation section(s); the system ranks all sections by the configuration's similarity scores and we compute MRR, Recall@k, and nDCG@k as defined in Section 3.5.2. Separately, on the court–to–regulation linking dataset, each full court decision is treated as a query and the manually annotated `gold_reg_sections` form the relevant set; the same ranking procedure is applied over the regulation corpus and Hit@k, Recall@k, and MRR are reported (Section 4.4.2.2). For the purpose of model selection, we use only the regulation–QA test split and report Recall@$\{1, 3, 5, 10\}$ and nDCG@$\{1, 3, 5, 10\}$, together with nonparametric bootstrap confidence intervals over questions. The primary selection criterion is mean Recall@10, with MRR used as a tie-breaker. The best-performing configuration under this criterion is then fixed as the default retriever inside the CAG pipeline: for each new query, the retriever selects a small set of top-ranked sections, these sections are concatenated into the prefix that is pre-encoded into the KV cache, and cache-augmented decoding proceeds as in Section 3.4. In the regulation-only CAG experiments, the generator is the fine-tuned Gemma–2–9B–IT model, whereas in the court decision experiments the same retrieval stack is reused but paired with the untuned base checkpoint, so that retrieval quality is held constant and any differences in behaviour can be attributed to the domain shift in inputs rather than to retrieval tuning.

3.5 Evaluation Metrics and Explainability Techniques

3.5.1 Text Generation Quality Metrics

BERT-F1 (BERTScore) Given reference tokens $X = (x_1, \ldots, x_m)$ and candidate tokens $Y = (y_1, \ldots, y_n)$, let $e(\cdot)$ be contextual embeddings from a pretrained encoder and define the cosine-similarity matrix $S_{ij} = \cos\big(e(x_i), e(y_j)\big)$. The soft precision/recall are

$$P_{\text{BERT}} = \frac{1}{n}\sum_{j=1}^{n} \max_{1\le i\le m} S_{ij}, \qquad R_{\text{BERT}} = \frac{1}{m}\sum_{i=1}^{m} \max_{1\le j\le n} S_{ij}, \tag{3.13}$$

and the F1 is

$$\text{BERT-F1} \;=\; \frac{2\,P_{\text{BERT}} R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}. \tag{3.14}$$

BERT-F1 aligns each token to its most semantically similar counterpart via cosine similarity and thus captures paraphrastic equivalence beyond exact n-gram overlap [68].

ROUGE_L (Longest Common Subsequence) Let $\text{LCS}(X, Y)$ be the length of the longest common subsequence between reference X and candidate Y. Define

$$R_{\text{LCS}} = \frac{\text{LCS}(X,Y)}{|X|}, \qquad P_{\text{LCS}} = \frac{\text{LCS}(X,Y)}{|Y|}, \tag{3.15}$$

and the (possibly recall-weighted) F-measure

$$\text{ROUGE_L} \;=\; \frac{(1+\beta^2)\,R_{\text{LCS}}\,P_{\text{LCS}}}{R_{\text{LCS}} + \beta^2 P_{\text{LCS}}}. \tag{3.16}$$

ROUGE_L rewards long in-sequence matches without requiring contiguity; in practice $\beta \gg 1$ makes it primarily recall-oriented [69].

ChrF^{++} ChrF computes an F-score over character n-gram matches; ChrF^{++} augments this with word-bigram overlap. With clipped overlap counts for character n-grams ($n = 1\!:\!N$) and word bigrams, form character-level precision/recall

$$P_{\text{chr}} = \frac{\sum_{n=1}^{N} w_n \,\text{match}_n}{\sum_{n=1}^{N} w_n \sum_{g\in G_n} c_Y(g)}, \quad R_{\text{chr}} = \frac{\sum_{n=1}^{N} w_n \,\text{match}_n}{\sum_{n=1}^{N} w_n \sum_{g\in G_n} c_X(g)}, \tag{3.17}$$

mix with word-bigram precision/recall by $P' = \alpha P_{\text{chr}} + (1-\alpha)P_{w2}$, $R' = \alpha R_{\text{chr}} + (1-\alpha)R_{w2}$, and report

$$\text{ChrF}^{++} \;=\; \frac{(1+\beta^2)\,P'\,R'}{R' + \beta^2 P'}. \tag{3.18}$$

character n-grams capture subword morphology (useful in agglutinative languages), while word bigrams add word-order sensitivity [70, 71].

BLEU For $n \in \{1, \ldots, N\}$ let p_n be the modified (clipped) n-gram precision. With reference length r and candidate length c, the brevity penalty is $\text{BP} = \exp\big(\min(0, 1 - r/c)\big)$. The BLEU score is

$$\text{BLEU} \;=\; \text{BP} \cdot \exp\left(\frac{1}{N}\sum_{n=1}^{N} \log p_n\right). \tag{3.19}$$

emphasizes exact n-gram matches and word order; the brevity penalty discourages overly short hypotheses [72].

Exact Match (EM) For Q examples with predictions $\hat{y}_i$ and normalized gold sets $\mathcal{A}_i$,

$$\text{EM} \;=\; \frac{1}{Q}\sum_{i=1}^{Q} \mathbb{I}[\hat{y}_i \in \mathcal{A}_i]\,. \tag{3.20}$$

a strict string-identity accuracy (after standard normalization), widely used in extractive QA [73].

Jaccard Similarity For token sets A and B,

$$J(A, B) \;=\; \frac{|A \cap B|}{|A \cup B|}. \tag{3.21}$$

an order-agnostic overlap coefficient in $[0, 1]$; 1 indicates identical token sets [74].

3.5.2 Retrieval Quality Metrics

Mean Reciprocal Rank (MRR) For a query set of size Q, let r_i denote the rank of the first relevant item for query i, and set $1/r_i = 0$ when no relevant item is retrieved. The mean reciprocal rank is then

$$\text{MRR} \;=\; \frac{1}{Q}\sum_{i=1}^{Q} \frac{1}{r_i}, \tag{3.22}$$

which measures how early, on average, the system returns at least one relevant item, with rank-1 hits receiving the highest credit [75].

Recall@k For query i, let R_i be the set of relevant items and $R_{i,k}$ the set of the top-k retrieved items. Recall@k is defined as

$$\text{Recall@}k \;=\; \frac{|R_i \cap R_{i,k}|}{|R_i|}, \tag{3.23}$$

capturing the proportion of relevant items that appear within the top-k results, irrespective of their internal ordering.

Hit@k For some analyses (notably the court–to–regulation linking experiment), we also report the simpler Hit@k indicator, which only checks whether *any* relevant item appears in the top-k:

$$\text{Hit@}k(i) \;=\; \begin{cases} 1, & \text{if } R_i \cap R_{i,k} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases} \tag{3.24}$$

Hit@k is the binary counterpart of Recall@k and answers the question "does the system retrieve at least one correct section within the top-k candidates?"; we again report macro-averaged values over queries.

nDCG@k Given graded relevance scores rel_i at rank i, the (discounted) cumulative gain at cutoff k is

$$\text{DCG@}k \;=\; \sum_{i=1}^{k} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)}. \tag{3.25}$$

The normalized discounted cumulative gain is obtained by dividing by the maximum achievable DCG at the same cutoff,

$$\text{nDCG@}k \;=\; \frac{\text{DCG@}k}{\text{IDCG@}k}, \tag{3.26}$$

where $\text{IDCG@}k$ is DCG computed on the ideal (sorted) ranking. nDCG@k rewards rankings that place highly relevant items early while normalizing scores across queries with different relevance profiles [76].

3.5.3 Explainability Techniques

SHAP We explain a single prediction of f using an additive, locally faithful surrogate over binary presence features $z' \in \{0,1\}^M$ [77]:

$$g(z') \;=\; \phi_0 + \sum_{i=1}^{M} \phi_i \, z'_i. \tag{3.27}$$

Imposing local accuracy, missingness, and consistency uniquely yields Shapley attributions $\{\phi_i\}$:

$$f(x) \;=\; \phi_0 + \sum_{i=1}^{M} \phi_i. \tag{3.28}$$

With value function $v(S) = \mathbb{E}[f(Z) \mid Z_S = x_S]$, the exact attribution is

$$\phi_i \;=\; \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!\,(M-|S|-1)!}{M!} \Big(v(S \cup \{i\}) - v(S) \Big), \qquad \phi_0 = \mathbb{E}[f(Z)]. \tag{3.29}$$

In our experiments the explained scalar is the answer log-probability,

$$f(x) \;=\; \log p_\theta(\hat{y} \mid x, c), \tag{3.30}$$

so positive ϕ_i increase the predicted answer's log-likelihood under context c, while negative ϕ_i decrease it.

3.6 Environment Settings

All experiments were executed on Google Colab Pro+ equipped with a single NVIDIA A100-SXM4 GPU featuring 40 GB of VRAM. The computational environment was based on Python 3.11.13 running on a Linux kernel (6.1.123+), with CUDA support enabled. The deep learning stack comprised PyTorch 2.6.0+cu124, Hugging Face Transformers 4.55.0, PEFT 0.17.0, Accelerate 1.9.0, Evaluate 0.4.5, and BitsAndBytes 0.47.0.

We evaluated four instruction-tuned LLMs for question answering task: *Llama-3-8B-Instruct*, *Qwen2-7B-Instruct*, *Mistral-7B-Instruct-v0.3*, and *Gemma-2-9B-IT*. Across all experiments, base weights were loaded in 8-bit via BitsAndBytes (`LLM.int8()`).

This kept memory usage within the 40 GB budget while maintaining throughput and numerically stable forward passes. For adaptation, we attached LoRA adapters via PEFT to attention and feed-forward modules while freezing the 8-bit base weights; optimizer states were maintained in standard precision appropriate to the mixed-precision setting.

Training and evaluation were orchestrated with Hugging Face Accelerate for hardware-aware dispatch, mixed precision, and gradient accumulation under constrained VRAM.

CHAPTER 4

EMPIRICAL ANALYSIS

4.1 Experimental Setup

This chapter details the empirical framework used to evaluate the proposed methodology across successive stages of model selection, adaptation, and inference. The central question is whether (i) adapting a LLM to Turkish, domain-specific fire-safety content and (ii) augmenting inference with CAG improve both answer quality and explainability in regulation-oriented question–answering.

The exposition follows the end-to-end pipeline: corpus construction and preprocessing, zero-shot screening of multiple instruction-tuned LLM candidates, parameter-efficient fine-tuning of the selected backbone, and integration of CAG with regulation and court-decision retrieval. Each stage is anchored in the theoretical considerations introduced in Chapter 3 and is accompanied by its main implementation details and evaluation design. The chapter concludes with a comparative assessment of model performance along reliability, retrieval robustness, and interpretability dimensions, thereby linking the methodological contributions of the thesis to their empirical outcomes in Turkish legal fire-safety decision support.

4.2 Design of the Datasets

The empirical analysis relies on two complementary corpora with distinct roles in the pipeline: (i) a regulation-centric question–answer dataset derived from the official Turkish fire-safety regulation and its guide, and (ii) a corpus of Turkish court decisions involving fire incidents and related liability judgments. The first dataset supervises and evaluates the core question–answering behaviour of the models; the

second provides an auxiliary knowledge source for retrieval within the CAG framework, allowing the system to ground answers both in statutory text and in case-based reasoning patterns

4.2.1 Regulation Dataset

The primary data source is the *Binaların Yangından Korunması Hakkında Yönetmelik Kılavuzu* (Guide to the Regulation on Fire Protection of Buildings) [78], published by the Ministry of Environment, Urbanization and Climate Change, General Directorate of Vocational Services, in December 2024. This guide complements the *Binaların Yangından Korunması Hakkında Yönetmelik*, originally promulgated in the Official Gazette on 19 December 2007 (No. 26735) and subsequently amended to reflect evolving safety requirements, and serves as an authoritative interpretative document for implementing its binding provisions.

The guide permits citation and quotation with proper attribution; all other rights remain with the Ministry. In this thesis, it is used exclusively for academic, non-commercial research. Apart from technical preprocessing for computational analysis, no substantive modifications are introduced, derived datasets are restricted to research use, and attribution to the Ministry is preserved in line with licensing terms and scholarly norms.

Methodologically, this source is well suited to the present study: the regulation is the central legal instrument governing fire safety in Turkey, its formally structured legal language (definitions, classifications, annexes, procedural requirements) provides a rigorous testbed for hierarchical and technical text, and its official, publicly available status supports reproducibility and transparency.

4.2.1.1 Preprocessing and Exploratory Data Analysis

The preprocessing pipeline follows a structure-first design and yields the section-level corpus reused throughout the thesis. Text is extracted directly from the PDF's embedded text layer using a Python-based parser, thereby avoiding optical character recognition and preserving the original character encoding and layout cues. For

each page, word-level coordinates are obtained and grouped into "visual lines" by bucketing tokens according to their vertical position (top coordinate) within a fixed tolerance and then ordering them from left to right; concatenating the tokens in each bucket reconstructs a reading order that remains faithful to the human layout even in the presence of indentation, lists, and multi-column formatting.

The resulting line stream is normalized through a sequence of deterministic transformations. Repeated headers, footers, and isolated page numbers are removed via regular expressions; hyphenation across line breaks is healed; and redundant whitespace is standardized. Short lines containing only a bullet, number, or letter (e.g. "(a)" or "1)") are merged with the subsequent line to reconstruct full list items. Finally, multiple blank lines are collapsed and the cleaned lines are concatenated into a single flattened string, which serves as the canonical input for segmentation and auditing.

A second script reinstates the legal hierarchy and constructs the section-level dataset. Using curated lists of all *Kısım*, *Bölüm*, *Madde*, and *Ek* headings, the script scans the flattened text, records the character offsets of each marker, and uses consecutive offsets to slice the text into coherent spans. Each span is assigned a canonical label that concatenates the full hierarchical path (e.g. "Sekizinci Kısım: Tehlikeli Maddelerin Depolanması ve Kullanılması – Üçüncü Bölüm: Parlayıcı ve Patlayıcı Gazlar – LPG tüplerinin kullanılması – Madde 109"). The resulting corpus consists of section–text pairs, which become the authoritative units for question–answer generation and retrieval.

After preprocessing, the structured dataset contains 205 sections, each represented as a pair `(section, text)`. Of these, 177 entries correspond to main-body regulatory articles (Madde 1–171), while the remaining 28 are annex or auxiliary sections (*Ek*-) that predominantly encode tabular constraints and special-case provisions. All section identifiers are unique and no empty texts remain, indicating that the segmentation and cleaning stages yield a one-to-one mapping between legal units and textual spans.

Table 4.1: Corpus statistics under Gemma-2 tokenization

Metric	**Value**
Characters	319,318
Characters (no spaces)	276,717
Tokens (Gemma-2 9B tokenizer)	95,658
Unique token IDs	3,720
Type–Token Ratio (TTR)	0.0389
Standardized TTR (STTR@1K, token IDs)[1]	0.3878
Sentences	2,803
Tokens per sentence — mean	34.02
Tokens per sentence — median	29.00
Tokens per sentence — Q1	2.00
Tokens per sentence — Q3	49.00
Tokens per sentence — p90	71.80
IQR (Q3–Q1)	47.00
Quartile skewness (Bowley)	-0.1490
Tail ratio (p90 / median)	2.4760

As summarised in Table 4.1, the regulation corpus comprises 95,658 Gemma subword tokens distributed over 2,803 sentences. The median sentence length is 29 tokens with an interquartile range of 47 (Q1 = 2, Q3 = 49), indicating substantial variation in syntactic span. The near-zero Bowley skewness (-0.149) suggests a roughly symmetric central mass, whereas the 90th percentile (71.8 tokens) is about 2.5 times the median, evidencing a pronounced upper tail driven by multi-level enumerations, cross-referenced conditions, and annex-style prescriptions. These characteristics support segmenting the corpus along legally meaningful boundaries (articles, subsections, annex elements) rather than fixed-length token windows, so that extended definitions and conditional clauses remain intact for both retrieval and generation.

[1] STTR@1K is the mean type–token ratio over 1,000-token sliding windows (stride 500) computed on Gemma subword IDs.

Unigram frequencies are computed after removing numerals and a curated Turkish stoplist, with no stemming and only display-level inflectional unification. Figure 4.1 visualizes the top 25 content terms. The head of the distribution is dominated by domain-specific nouns: the most frequent term, `yangın`, appears 635 times ($\approx 2.1\%$ of all tokens), while the next four terms (`bina`, `kaçış`, `yapı`, `gerekir`) contribute a further $\approx 4.8\%$. Within the top-25 list, individual token shares range from roughly 0.4% to 2.1%, with a median around 0.6%. This concentration in a small, semantically coherent vocabulary is consistent with the highly topical and technical nature of the regulation, where obligations, physical structures, and safety systems recur across sections.

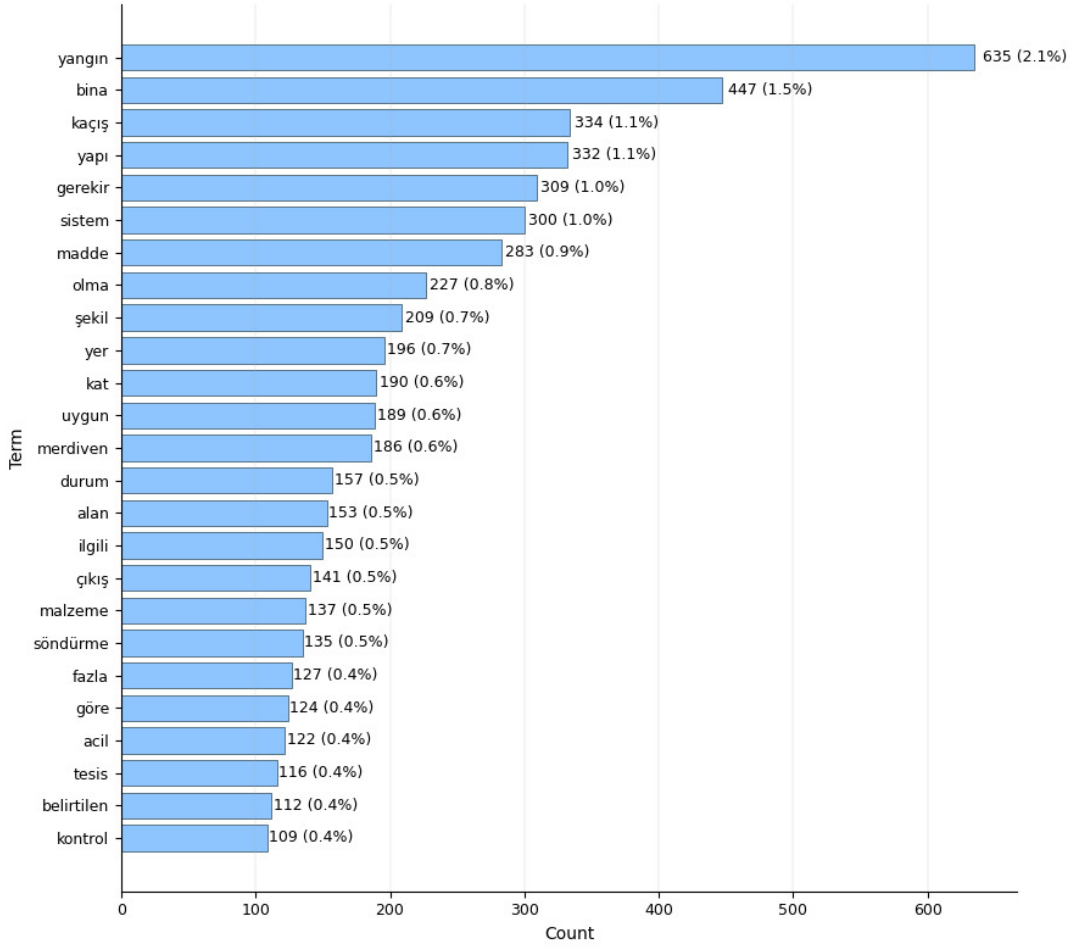


Figure 4.1: Top-25 content terms without stopword and numeral

Table 4.2: Word-level statistics for the regulation corpus

Metric	**Value**
Vocabulary size (unique words)	6,511
Lexical density (% content words)	85.14%
Average word length (chars)	6.38
Stopwords — percentage	14.86%
Hapax legomena — count[1]	3,376
Hapax legomena — % of types	49.01%
Dis legomena — count[2]	1,121
Dis legomena — % of types	16.27%
Zipf rank exponent s (|slope|)[3]	1.06
Power-law α (MLE, $k_{\min} = 2$)[4]	1.842
Numerals — token share	6.97%
Numeric+unit — token share	1.82%

The corpus exhibits a content-dense, technical vocabulary shaped by agglutinative morphology. The 6,511 unique word types and high lexical density (85.14%) indicate that semantic information is carried predominantly by domain-specific terms rather than function words, which is further reflected in the relatively low stopword share (14.86%). The average word length of 6.38 characters is characteristic of Turkish legal and regulatory discourse, where complex suffix chains encode modality, case, and aspect.

Distributional analysis reveals substantial sparsity at the type level: nearly half of the vocabulary (49.01%) consists of hapax legomena and a further 16.27% appears exactly twice. This pattern is consistent with productive affixation and compound technical expressions, yielding many low-frequency surface forms. Nevertheless, the rank–frequency curve closely follows a Zipfian relation with slope ≈ 1.06, while the

[1]Hapax legomenon: a type observed exactly once in the corpus.

[2]Dis legomenon: a type observed exactly twice in the corpus.

[3]Zipf rank exponent s: rank–frequency follows $f(r) \propto r^{-s}$; larger s implies a faster drop and greater dominance of high-rank words.

[4]Power-law α (MLE): the tail of type-frequency counts follows $P(k) \propto k^{-\alpha}$ for $k \geq 2$; larger α indicates a thinner tail with fewer ultra-rare types.

fitted power-law exponent $\alpha \approx 1.842$ for $k_{\min} = 2$ confirms a heavy-tailed regime in which a small set of high-frequency tokens coexists with a long tail of rare types. Methodologically, these properties argue against aggressive stemming or lemmatization that would collapse morphologically meaningful distinctions. Instead, they motivate tokenisation and smoothing strategies that respect Turkish morphology and prioritise content-bearing terms in retrieval and QA alignment.

Numeric tokens form a non-trivial part of the corpus: 6.97% of tokens are numerals and an additional 1.82% are numeral–unit composites. These typically encode thresholds, dimensions, and performance criteria (e.g. distances, capacities, pressure levels) that are crucial for the operational interpretation of fire-safety rules. Their explicit preservation is therefore essential both for verifiable retrieval and for evaluating model outputs against quantitatively specified constraints.

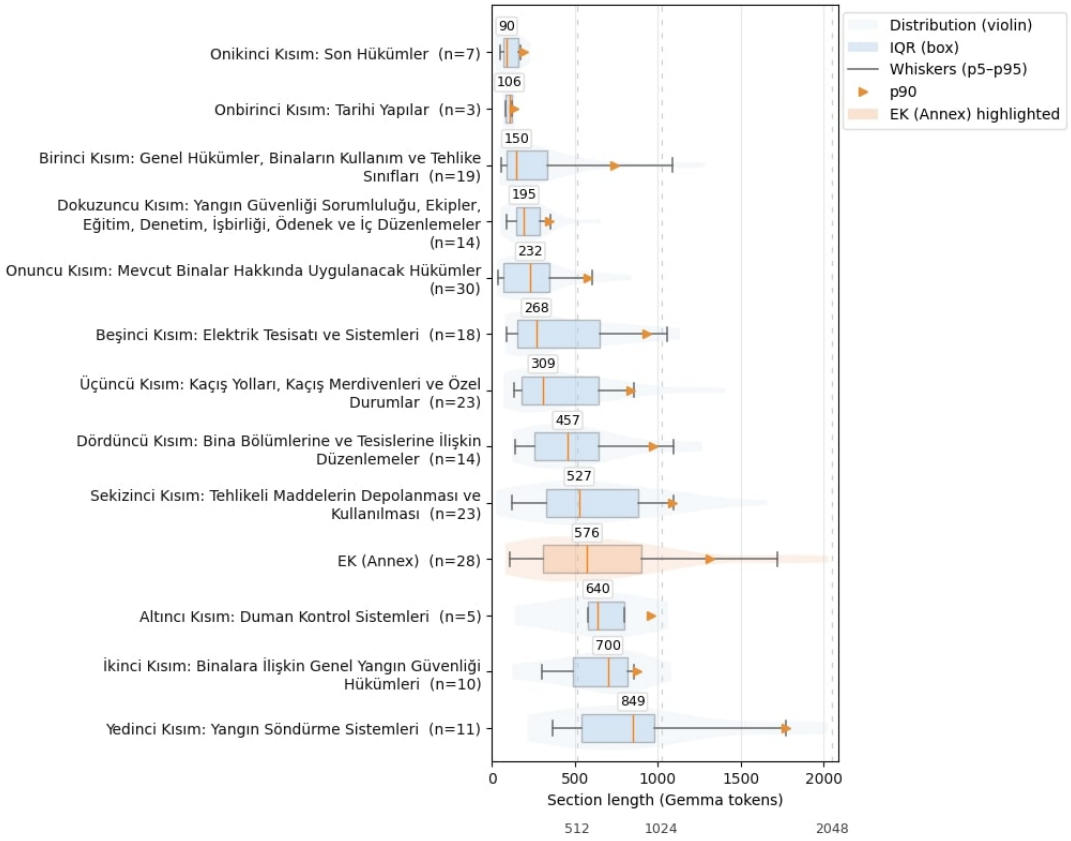


Figure 4.2: Section length by regulatory category

Section-length distributions, summarised in Figure 4.2, exhibit pronounced heterogeneity across regulatory categories, reflecting both functional scope and drafting

conventions. The shortest median lengths arise in the concluding provisions ($\sim$ 93 tokens) and historical clauses ($\sim$ 106 tokens), which largely consist of formulaic statements. General provisions and coordination rules occupy the lower mid-range ($\sim$ 195–232 tokens), while electrical-systems and egress-related sections show intermediate typical lengths ($\sim$ 268–309 tokens). Provisions governing building elements extend further ($\sim$ 457 tokens), whereas hazardous-material regulations ($\sim$ 527; $n = 23$) and annexes ($\sim$ 576; $n = 28$) display both higher medians and broader dispersion.

The longest sections are found in smoke-control and extinguishing-system requirements, as well as in general fire-safety rules for buildings, with medians of approximately 640, 849, and 700 tokens, respectively. In many categories the p_{90} markers lie far to the right of the interquartile boxes, indicating strongly right-skewed distributions driven by enumeration-heavy technical passages. Annexes and system-oriented parts exhibit the widest interquartile ranges, whereas definitional and general-provision sections remain comparatively compact. These patterns confirm that section type is strongly associated with both central tendency and variability in length, reinforcing the choice to treat sections (and annex substructures) as primary retrieval units rather than splitting them into uniform-length slices.

4.2.1.2 Question–Answer Generation

To enable supervised evaluation, the section-level corpus is extended with question–answer (QA) pairs. Programmatically, this transformation is implemented as a Python pipeline that reads a JSON file of sections, constructs an instruction-style prompt for each entry, and queries the instruction-tuned decoder-only model Gemma-2-9B-IT to generate JSON-formatted QA lists.

All random number generators are seeded for reproducibility, and the model is run in mixed precision on a GPU. To avoid context truncation, longer sections are split into coherent chunks that fit within the effective context window once the system prompt and output budget are included. The expected number of QA pairs for a section, n, is estimated from the number of numbered items and the total word count, enforcing a minimum of five pairs but allowing more for longer, more articulated provisions.

Prompt construction is hierarchy-aware. For ordinary (non-annex) sections, the heading is parsed into a sequence of levels (part, chapter, section, article), and a short metadata block describing this hierarchical path, focus subheading, and article number is prepended to the system instruction. This metadata explicitly constrains the model to remain within the focal subheading and to avoid drift into sibling sections. For annexes whose content is essentially tabular ("Ek-" sections), the prompt includes an explanation of how colon, semicolon, and period delimit key–value pairs and rows, instructing the model to derive questions solely from these textualised table entries.

The core system instruction, written entirely in Turkish to match the language of the regulation, is:

```
Aşağıda, Binaların Yangından Korunması Hakkında Yönetmelik Kılavuzu'ndan
alınmış resmi ve bağlayıcı bir mevzuat metni verilecektir.
Yalnızca bu metne dayanarak, doğrudan mevzuatın ilgili hükümlerini esas
alan {n} farklı ve çeşitli soru-cevap çifti oluştur.

- Her madde, alt madde veya tanım için en az bir bilgi verici ve özgün
  soru-cevap üret.
- Sorular ve cevaplar sadece metindeki bilgiyle sınırlı olmalı;
  metinde olmayan çıkarım ya da genelleştirme yapma.
- Soru türlerini çeşitlendir (tanım, kapsam, uygulama, yükümlülük, istisna,
  sorumluluk, sınır/değer, süreç, neden-sonuç, karşılaştırma vb.).
- Şekil/Tablo gibi görsel referanslarını soru veya cevaplarda KULLANMA
  (örn. "Bkz.", "Şekil", "Tablo", "Figure").
- SADECE GEÇERLİ bir JSON LİSTESİ üret; başka açıklama, başlık veya
  ek bilgi ekleme.

Format:
[
  {
    "question": "Soru burada",
    "answer":  "Cevap burada"
  }
]
```

When the output cannot be parsed as valid JSON or yields too few usable pairs, generation is retried up to a fixed limit. The decoded JSON list is then passed through a

hierarchy-aware filter that removes empty or trivially short strings, generic placeholders (e.g. “bu madde”), any question or answer referring to figures or tables (“Şekil”, “Bkz.”, “Tablo”, “Figure”), and pairs whose combined question–answer tokens exhibit insufficient lexical overlap with the source text. For non-annex sections, a sibling-drift guard discards pairs that mention headings absent from the current section, and extractiveness is further enforced by requiring at least one shared bigram between the answer and the source span, except for very short answers.

At the orchestration level, the generator maintains a resume-safe JSON file: previously processed sections are reloaded and skipped, and after each new section the corresponding QA list is appended and written back to disk. GPU usage is kept stable by clearing caches between sections and generating one section at a time.

The final dataset comprises 204 sections and 1,431 QA pairs. The number of QA pairs per section has a median of 5, a mean of about 7.0, and a 90th percentile of 13, with a maximum of 24 questions for the most complex provisions. Section texts average roughly 1,558 characters ($\approx$ 208 words), and the number of QA pairs is strongly correlated with section length (Pearson $r \approx 0.74$), confirming that longer provisions yield more questions. Generated questions are concise but informative (mean 85 characters, median 82), whereas answers are more expansive (mean 125, median 114), producing the intended interrogative–explanatory asymmetry. Quality checks on the final JSON file reveal no empty fields and no exact duplicate QA pairs, indicating that the parsing and filtering pipeline successfully eliminates malformed or redundant items.

Figure 4.3 presents representative samples from the dataset, illustrating how narrative clauses and numerically parameterised annexes are mapped into structured QA items. Questions remain tightly aligned with their source spans and collectively cover definitions, scope conditions, prohibitions, thresholds, and classification rules. Ellipses in the JSON snippet indicate additional text, further QA pairs for each section, and further sections that are omitted for brevity.

Figure 4.3: Excerpt from the generated regulation–QA dataset


```
[
  {
    "section": "İkinci Kısım: Binalara İlişkin Genel Yangın Güvenliği Hükümleri –
Üçüncü Bölüm: Yangın Kompartımanları, Duvarlar, Döşemeler, Cepheler ve Çatılar
– Yangın kompartımanları – Madde 24",
    "text": "... 5. Atriumlu bölümlere, sadece düşük ve orta tehlikeli
sınıfları içeren kullanımlara sahip binalarda müsaade edilir. Atrium
alanının hiçbir noktada 90 m²'den küçük olmaması esastır. Alanı 90 m²'den
küçük olan atrium boşluklarının çevresi her katta en az 45 cm yüksekliğinde
duman perdesi ile çevrelenir ve yağmurlama sistemi ile korunan binalarda
duman perdesinden 15 ila 30 cm uzaklıkta, aralarındaki mesafe en çok
2 m olacak şekilde yağmurlama başlığı yerleştirilir...",
    "qa_pairs": [
      {
        "question": "Atrium alanı 90 m²'den küçük olduğunda yangın
güvenliği için ne tür önlemler alınmalıdır?",
        "answer": "Atrium boşluklarının çevresi her katta en az 45 cm
yüksekliğinde duman perdesi ile çevrilmeli ve yağmurlama sistemi ile
korunmalıdır. Duman perdesi ile korunan binalarda bu perdeler 15 ila
30 cm uzaklıkta yerleştirilmeli ve aralarındaki mesafe en çok 2 m olmalıdır."
      },
      ...
    ]
  },
  {
    "section": "Onuncu Kısım: Mevcut Binalar Hakkında Uygulanacak Hükümler –
Birinci Bölüm: Genel Hükümler – İlave çıkış ve kaçış merdiveni – Madde 141",
    "text": "1. Binada, ilave çıkış gerekliliğini veya kaçış merdivenlerinin
yeniden düzenlenme mecburiyetini gerektiren bir kullanım mevcut ise,
binanın bütünü göz önüne alınarak, bina sahibi veya kat malikleri tarafından,
binanın tamamı için ilave çıkış veya kaçış merdiveni yaptırılması şarttır...",
    "qa_pairs": [
      {
        "question": "Binada ilave çıkış veya kaçış merdiveni yaptırılması hangi
 koşullarda zorunludur?",
        "answer": "Binada, ilave çıkış gerekliliğini veya kaçış merdivenlerinin
yeniden düzenlenme mecburiyetini gerektiren bir kullanım mevcut ise,
binanın bütünü göz önüne alınarak bina sahibi veya kat malikleri tarafından
yaptırılması şarttır."
      },
      ...
    ]
  },
  ...
]
```

4.2.2 Court Decision Dataset

To assess performance on real judicial material, we construct a complementary dataset of Turkish court decisions. Using the subscription database Legalbank Elektronik Hukuk Bankası [79], we retrieve over 500 decisions with fire- and building-related query terms. Each decision is manually screened to verify that it applies or interprets the fire safety regulation. For the CAG–XAI experiments, we use 36 decisions that (i) explicitly cite the fire safety regulation (by title or article number) and (ii) yield clean, extractable text after OCR and preprocessing. Each judgment is treated as long-form input (*case text*) and linked to the regulation sections relied upon in the reasoning. The full case text serves as a query to the best-performing Fusion+Rerank configuration (hybrid BM25 and TF–IDF on surface and morph-normalized tokens plus a cross-encoder reranker), which returns top-k candidate sections. These sections, together with the case text and a Turkish JSON-style instruction prompt, are passed to the base Gemma-2-9B-IT model.

Table 4.3 summarises the length profile of the 36 decisions. On average, a judgment contains 1.6k words and 4.6k Gemma tokens (median 1.5k and 4.1k). At the 90th percentile, decisions reach 3.4k words and 9.1k tokens, and the longest case exceeds 6.6k words and 17.5k tokens (up to 493 sentences). Many decisions already occupy a large fraction of the context window, and some exceed it once retrieved regulation sections are appended, motivating retrieval- and cache-augmented strategies instead of naïve full-document prompting.

Table 4.3: Descriptive statistics for the court decision dataset

	Mean	**Std. Dev.**	**Min**	**Median**	**Max**
Characters (n_{chars})	13,704.4	11,476.8	1,121	12,102.5	54,653.0
Words (n_{words})	1,641.9	1,386.3	124	1,464.0	6,612.0
Sentences (n_{sent})	127.7	106.9	20	95.5	493.0
Paragraphs (n_{para})	3.4	1.9	1	3.0	9.0
Gemma tokens (n_{tokens})	4,610.1	3,705.8	486	4,099.5	17,547.0

4.3 Model Evaluation

4.3.1 Dataset Splits

The QA corpus contains 1,430 question–answer pairs, divided into train (950), validation (139), and test (341) subsets using a fixed random seed. Table 4.4 summarizes surface statistics computed after minimal normalization. Question lengths are stable across splits, averaging about eleven words. Answers are longer and more variable, with mean lengths near sixteen words and maxima between 47 and 76 words.

Table 4.4: Descriptive statistics of the QA dataset splits

Metric	**Train**	**Validation**	**Test**	**Total**
Samples	950	139	341	1430
Mean question length (words)	10.91	10.73	10.80	10.87
Mean answer length (words)	16.36	15.81	16.42	16.32
Min. question length	3	4	3	3
Max. question length	30	24	32	32
Min. answer length	2	1	1	1
Max. answer length	76	47	66	76
Questions with numbers (%)	14.00	19.42	11.44	13.92
Answers with numbers (%)	39.26	44.60	41.06	40.21

Numerical content is well distributed: 11–19% of questions and around 40% of answers contain at least one number. Only four near-duplicate QA pairs appear in the entire dataset, confirming that the splits provide diverse and non-overlapping supervision.

4.3.2 Zero-Shot LLM Baselines

The zero-shot results in Table 4.5 provide an initial benchmark for how well general-purpose instruction-tuned LLMs can answer Turkish fire-safety questions without any domain adaptation. Among the four candidates, Gemma-2-9B-IT attains the

strongest overall performance: it achieves the highest BERTScore-F1 (0.869), the best character-level ChrF^{++} (32.32), and the highest ROUGE-L (0.188), indicating that its predictions are both semantically closest to the gold regulation-based answers and comparatively well aligned with their surface form. Meta-Llama-3-8B-Instruct and Qwen2-7B-Instruct form a second tier: their BERTScore-F1 values (0.858 and 0.857, respectively) are close to Gemma's, but lower ChrF^{++} (28.48 and 25.81) and ROUGE-L (0.138 and 0.116) show that they diverge more from the exact phrasing and structure of the gold answers. Mistral-7B-Instruct-v0.3 is consistently weakest across metrics, with the lowest BERTScore-F1 (0.841), ChrF^{++} (24.25), ROUGE-L (0.094), and Jaccard similarity (0.035).

BLEU remains below 0.01 for all models, and the normalized exact-match rate is 0.00 in every case. This behaviour is expected: the reference answers are short but lexically rigid, often containing enumerations, formal legal expressions, and precise numeric thresholds. Even when the models capture the correct core meaning, they rarely reproduce the full token sequence (including legal boilerplate and measurements) required for exact string identity. In this regime, semantic similarity (BERTScore-F1) and character-level overlap (ChrF^{++}) are therefore more informative than exact-match style metrics.

Table 4.5: Zero-shot performance of instruction-tuned LLMs

Model	**BERTScore-F1**	**ChrF^{++}**	**ROUGE-L**	**BLEU**	**Jaccard**
Gemma-2-9B-IT	0.869	32.32	0.188	0.007	0.096
Meta-Llama-3-8B-Instruct	0.858	28.48	0.138	0.004	0.075
Qwen2-7B-Instruct	0.857	25.81	0.116	0.002	0.052
Mistral-7B-Instruct-v0.3	0.841	24.25	0.094	0.001	0.035

To complement aggregate metrics, we examine qualitative zero-shot examples to evaluate how these models respond to regulatory questions without domain adaptation. Each example compares (i) the true regulation-based reference answer and (ii) the outputs of the four models. This analysis highlights behaviour at the semantic, factual, and stylistic levels—particularly their tendency to hallucinate article numbers, fabricate standards, or generalise instead of providing precise regulatory values.

Table 4.6: Zero-shot answers for Example 1 (idx=26)

Role	Text
Question	Bina çevresinde 10–30 metre mesafe içerisinde ağaçların durumu nasıl olmalıdır?
Reference	Bina çevresinde 10–30 metre mesafe içerisinde varsa ağaç araları 3 metreden az olamaz, tabii zeminden 2 metreye kadar olan dallar temizlenir ve zemin yüzeyi her altı ayda bir kuru nebattan temizlenir.
Gemma-2-9B-IT	Binalarda Yangından Korunma Hakkındaki Yönetmelik'e göre, bina çevresindeki ağaçlar 10–30 metreyi aşmayan alanda kesilmeli veya budanmalıdır. Yönetmeliğin ilgili maddesi bu konuda detaylı bilgi vermektedir.
Qwen2-7B-Instruct	Binaların Yangından Korunması Hakkındaki Yönetmelik'te belirtilen teknik detaylara göre, bina çevresindeki ağaçların durumu şu şekilde olmalıdır: 1. Ağaçlar bina ile diğer yapılar arasında 15 metreyi aşmamalıdır...
Mistral-7B-Instruct-v0.3	Bina çevresine ilişkin koruma koşullarını düzenleyen bir yönetmelikte, 15 metre mesafedeki ağaçların budanması gerektiği belirtilir. Ancak bu mesafe bazı koşullarda 16 metreye kadar uzatılabilir...
Meta-Llama-3-8B-Instruct	Yönetmeliğe göre, "Bina çevresine 10 metreden az mesafe içerisindeki ağaçlar, bina yangını riskini artırmamalıdır." Bu nedenle, bina çevresindeki ağaçların en az 10 metre mesafede olması gerekir...

Table 4.6 illustrates a critical failure mode of zero-shot LLMs when applied to regulation-based tasks: the "illusion of competence" through semantic mimicry. While all four models generate responses that are syntactically coherent and topically relevant, correctly identifying the query as a fire-safety spacing issue, they uniformly fail to re-

trieve the specific quantitative constraints (3 meters spacing, 2 meters pruning height) essential for regulatory compliance. Theoretically, this reflects the limitation of probabilistic next-token prediction in low-resource technical domains; the models rely on high-probability generic safety language rather than retrieving exact, low-frequency numeric tokens found in the specific regulation. The behavior of Mistral-7B-Instruct-v0.3 is particularly notable for its "hallucinated citation" error, where it fabricates a specific clause to lend authority to incorrect values. This divergence underscores that zero-shot instruction tuning aligns models to be helpful and fluent, but does not inherently ground them in specific external knowledge bases, leading to dangerous failures where the model confidently asserts incorrect safety thresholds.

Table 4.7: Zero-shot answers for Example 2 (idx=58)

Role	**Text**
Question	Kullanıcı yükü 25 kişi olan yüksek tehlikeli mekânlarda kaç çıkış bulunması gerekir?
Reference	25 kişinin aşılması halinde yüksek tehlikeli mekânlarda en az 2 çıkış bulunması gerekir.
Gemma-2-9B-IT	Kullanıcı yükünün 20'den fazla olduğu yüksek tehlikeli mekânlar için en az iki adet çıkış zorunludur. Bu bilgi "Bina yangın güvenliği yönetmeliği"nin ilgili maddesinde yer almaktadır.
Qwen2-7B-Instruct	Yönetmelikte belirtilen teknik detaylara göre, 20 kişiye kadar olan yüksek riskli mekânlarda en az iki çıkış bulunmalıdır. 21–50 kişi için en az üç çıkış, 51–100 kişi için en az dört çıkış gerekir...
Mistral-7B-Instruct-v0.3	Bazı yönergelerde yüksek tehlikeli mekânlarda tek çıkışın yeterli olabileceği belirtilmiştir; açık alanlarda ise ek çatılar altında çıkış kapakları bulunmalıdır...
Meta-Llama-3-8B-Instruct	Yönetmeliğe göre, "Kullanıcı yükünün 25 kişiden fazla olması halinde, yüksek tehlikeli mekânda en az 2 çıkış bulunmalıdır." Bu nedenle, 25 kişilik bir mekânda en az iki çıkış bulunmalıdır.

Table 4.7 provides empirical evidence of the "semantic-numeric alignment gap" in

LLMs. The reference regulation stipulates a precise conditional logic: if occupancy exceeds 25 persons in a high-hazard area, a minimum of two exits is required. The results demonstrate that while models like Gemma-2-9B-IT can approximate the outcome ("two exits"), they frequently fail to reproduce the governing condition or citation correctly. Theoretically, this suggests that the models have learned a correlation between "high hazard" and "multiple exits" from their pre-training data, but lack the specific retrieval mechanism to verify the "25 persons" threshold. More concerning is the behavior of Qwen2-7B-Instruct, which hallucinates a complex, tiered scaling rule (e.g., "21-50 people requires 3 exits") that does not exist. This "parametric fabrication" indicates that when a model faces uncertainty about a specific value, it may default to generating plausible-sounding but fictitious technical scaffolding to satisfy the user's prompt structure.

Table 4.8: Zero-shot answers for Example 3 (idx=167)

Role	**Text**
Question	Gazlı yangın söndürme sistemlerinin tasarımında hangi standart esas alınır?
Reference	TS ISO 14520 standardı esas alınır.
Gemma-2-9B-IT	Gazlı yangın söndürme sistemi tasarımlarında TS 10947 standardı esas alınmaktadır.
Qwen2-7B-Instruct	Gazlı söndürme sistemleri tasarımında genellikle "NFPA 13" adlı Amerikan standardı esas alınır...
Mistral-7B-Instruct-v0.3	Gazlı yangın söndürme sistemlerinin tetkikinde NFPA 13 standardı kullanılmaktadır...
Meta-Llama-3-8B-Instruct	Gazlı yangın söndürme sistemlerinin tasarımında TS 2585 standardı esas alınır.

Table 4.8 underscores a pervasive limitation in Large Language Models regarding normative identifier prediction and the boundaries of parametric memory. While the question demands the retrieval of a specific, low-frequency alphanumeric token se-

quence ("TS ISO 14520"), every zero-shot baseline fails to recall this exact standard, instead substituting it with hallucinated or irrelevant codes.

A systematic evaluation of the candidate models reveals a consistent 'illusion of competence': while instruction-tuned baselines demonstrate high fluency in broad, conceptual reasoning, they exhibit a fundamental inability to satisfy the rigid, numerically constrained dependencies characteristic of fire-safety regulation. The experiments uncover pervasive parametric hallucination, where models compensate for the absence of specific retrieval context by fabricating plausible-sounding but factually incorrect article numbers, safety thresholds, and technical standards. Comparative analysis indicates that while Gemma-2-9B-IT achieves the highest semantic alignment with regulatory intent, Qwen and Mistral display significant susceptibility to generative fabrication, and Meta-Llama frequently produces pseudo-citations to simulate authority. These findings empirically confirm that the probabilistic next-token prediction of zero-shot LLMs is insufficient for statutory compliance, necessitating a shift toward retrieval-augmented architectures that ground generation in verifiable external evidence.

4.3.3 Fine-Tuned Model

In this stage, the base model `google/gemma-2-9b-it` is adapted to the Turkish fire-safety domain using 8-bit QLoRA to balance computational efficiency with representational fidelity. Unlike standard fine-tuning approaches that rely solely on validation loss, a metric that minimizes perplexity but does not guarantee semantic faithfulness in open-ended generation, this methodology implements a metric-driven selection protocol based on BERTScore-F1 maximization. The adaptation process employs an extended hyperparameter sweep across four theoretical dimensions: adapter placement strategy, rank-alpha scaling pairs (r, α), learning rate $\eta \in \{2.0 \times 10^{-4}, 2.5 \times 10^{-4}\}$, and LoRA dropout $\delta \in \{0.05, 0.10\}$. Training is executed with `paged_adamw_8bit` optimizers to manage memory spikes, using cosine scheduling with a 10% warmup to stabilize the initial optimization trajectory and an early stopping mechanism with a patience of two epochs to prevent overfitting on the limited regulation corpus ($N = 950$).

The analysis of parameter sensitivity reveals distinct theoretical trade-offs between model capacity and generalization in low-resource legal domains. Increasing the rank r from 16 to 64 theoretically expands the expressive capacity of the trainable update ΔW, yet empirical observations suggest diminishing returns beyond moderate ranks ($r = 32$); in data-constrained regimes, highly expressive adapters risk overfitting the stochastic noise of the training set rather than learning robust legal reasoning patterns. Furthermore, the comparison between adapter placements indicates that the `attn_complete` strategy, which restricts updates to the self-attention mechanism, offers superior stability compared to `attn_mlp`. Theoretically, this implies that the adaptation task is more effectively solved by learning to route attention toward relevant regulatory tokens, optimizing the "retrieval" head of the network, rather than by rewriting the model's internal factual representations in the feed-forward layers, which are more susceptible to catastrophic forgetting. Consequently, the introduction of higher dropout (0.10) becomes essential for high-rank configurations, acting as a necessary regularizer to enforce sparsity and prevent the memorization of surface-level phrasing.

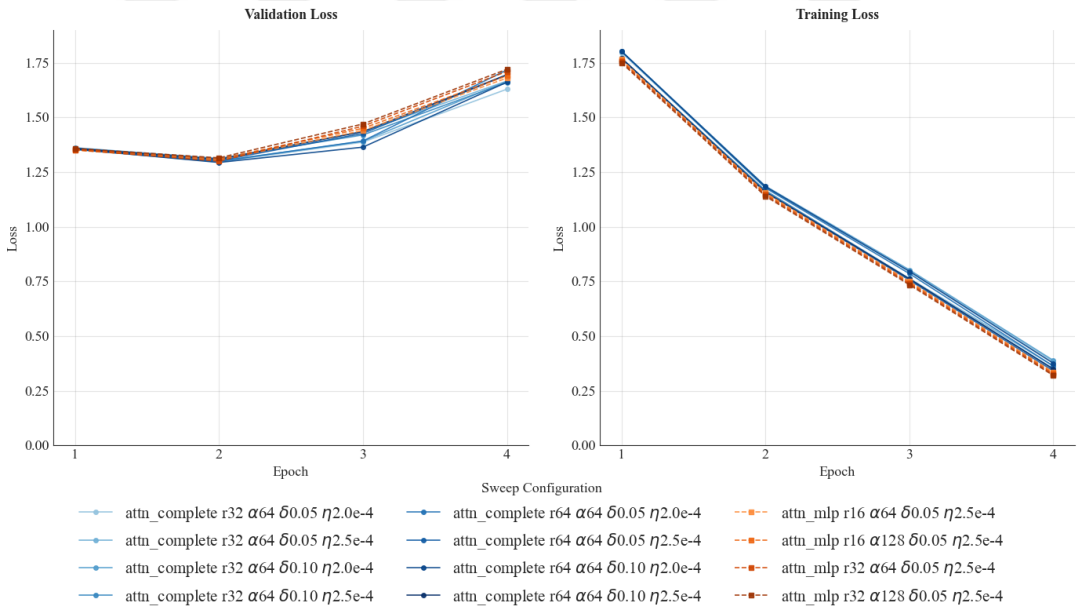


Figure 4.4: Loss across QLoRA sweep configurations

Ultimately, the selection process identifies the configuration (`attn_complete`, $r = 32$, $\alpha = 64$, $lr = 2.5 \times 10^{-4}$, $dropout = 0.05$) as the optimal Pareto frontier, bal-

ancing effective adaptation with robust generalization. While training loss decreases monotonically across all sweep configurations, validation BERTScore follows a concave trajectory, peaking before the loss fully converges. This divergence highlights a critical "alignment gap" where minimizing perplexity (prediction uncertainty) does not necessarily equate to maximizing generation quality (semantic correctness).

4.4 CAG Evaluation

4.4.1 Regulation-Only CAG with the Fine-Tuned Model

Table 4.9 reports answer quality for the regulation-only CAG setting with the fine-tuned Gemma-2-9B-IT adapter. On the held-out legal QA test split, CAG attains BERTScore-F1 $= 0.93$, chrF^{++} $= 58.4$, ROUGE-L $= 0.54$, BLEU $= 0.34$, Jaccard $= 0.42$, and normalized exact match $= 0.062$. Relative to the zero-shot baseline, BERTScore increases by 0.06 absolute, chrF^{++} by roughly 26 points, and ROUGE-L by 0.35, indicating markedly closer agreement with reference answers at both token and character level. BLEU rises from 0.007 to 0.336 (approximately $48\times$), and Jaccard from 0.10 to 0.42, reflecting substantially higher n-gram and set overlap. Compared to the fine-tuned, non-CAG regime, CAG yields additional gains of 0.03 BERTScore, 0.29 ROUGE-L, and more than threefold increases in BLEU and Jaccard. The normalized exact-match rate increases from 0.003 to 0.062, suggesting that combining domain-specific fine-tuning with cache-based access to the full regulation enables the model to reproduce statutory wording more precisely on this QA benchmark.

Table 4.9: Legal QA performance across inference regimes

Regime	**BERTScore-F1**	**ChrF^{++}**	**ROUGE-L**	**BLEU**	**Jaccard**
Zero-shot Gemma-2-9B-IT	0.869	32.32	0.188	0.007	0.096
Fine-tuned	0.897	30.35	0.250	0.026	0.130
CAG	0.930	58.38	0.538	0.336	0.421

Furthermore, the sampled cases show that the system retrieves the correct regulatory

section and generates answers that are either verbatim copies of the reference text or differ only by minor surface-form variations. In all three examples, the model preserves the legally salient information, distance ranges, required number of exits, and the exact standard identifier, indicating a high-fidelity, text-grounded generation behaviour consistent with the strong lexical-overlap metrics reported in Section 4.4.1.

Table 4.10: Representative CAG answers on the legal QA test set

Role	**Text**
Example 1 (test index 26)	
Question	Bina çevresinde 10–30 metre mesafe içerisinde ağaçların durumu nasıl olmalıdır?
Reference	Bina çevresinde 10–30 metre mesafe içerisinde varsa ağaç araları 3 metreden az olamaz, tabii zeminden 2 metreye kadar olan dallar temizlenir ve zemin yüzeyi her altı ayda bir kuru nebattan temizlenir.
CAG	Bina çevresinde 10–30 metre mesafe içerisinde ağaç araları 3 metreden az olamaz, tabii zeminden 2 metreye kadar olan dallar temizlenir ve zemin yüzeyi her altı ayda bir kuru nebattan temizlenir.
Example 2 (test index 58)	
Question	Kullanıcı yükü 25 kişi olan yüksek tehlikeli mekânlarda kaç çıkış bulunması gerekir?
Reference	25 kişinin aşılması halinde yüksek tehlikeli mekânlarda en az 2 çıkış bulunması gerekir.
CAG	Kullanıcı yükü 25 kişi olan yüksek tehlikeli mekânlarda en az 2 çıkış bulunması gerekir.
Example 3 (test index 167)	
Question	Gazlı yangın söndürme sistemlerinin tasarımında hangi standart esas alınır?
Reference	TS ISO 14520 standardı esas alınır.
CAG	Gazlı yangın söndürme sistemlerinin tasarımında TS ISO 14520 standardı esas alınır.

4.4.2 Retrieval Metrics Results

4.4.2.1 Retrieval Results on Regulation QA

For each query, the relevant item is the gold regulation section from which the reference answer was constructed. Table 4.11 summarizes mean retrieval effectiveness across all configurations introduced in Section 3.4.1.

Lexical BM25 and its morph-normalized variant (BM25+lemma) achieve high effectiveness, with MRR around 0.79 and Recall@10 above 0.90, indicating that simple lexical matching already retrieves the correct section within the top–10 candidates for most queries. The Fusion model, which combines BM25, BM25+lemma, and TF–IDF (optionally with dense scores), yields small but consistent gains in Recall@3–10 and nDCG@10 relative to BM25, suggesting that hybrid weighting can modestly refine the ranking beyond purely lexical evidence. The supervised SimCSE dense retriever attains substantially lower MRR and nDCG@10, and lower recall at all cutoffs, indicating that in this setting dense representations trained on the available supervision do not yet match the reliability of lexical methods. The best-performing configuration is Fusion+Rerank, which combines hybrid fusion with a BERTurk-based cross-encoder: it attains the highest MRR (0.81), Recall@10 (0.93), and nDCG@10 (0.84). In this variant, a relevant section appears within the top–3 in over 86% of cases and within the top–10 in nearly 93%, providing a high-coverage retrieval stage for subsequent CAG+retrieval experiments.

Table 4.11: Retrieval effectiveness for different configurations

Configuration	**MRR**	**Recall@1**	**Recall@3**	**Recall@5**	**Recall@10**	**nDCG@10**
BM25	0.7876	0.7126	0.8504	0.8827	0.9062	0.8138
BM25+lemma	0.7856	0.7097	0.8387	0.8768	0.9179	0.8152
SimCSE	0.1690	0.0909	0.1613	0.2170	0.3167	0.1872
Fusion	0.7831	0.7009	0.8534	0.8856	0.9208	0.8142
Fusion+Rerank	0.8111	0.7419	0.8622	0.8944	0.9296	0.8375

4.4.2.2 Retrieval Results on Court–to–Regulation Linking

We evaluate the hybrid retrieval module described in Section 3.4.1 on the *court–to-regulation* linking task using the manually annotated `gold_reg_sections` for all court decisions. For each judgment, the full decision text is used as a query and the system returns a ranked list of candidate regulation sections. We report macro-averaged Hit@k, Recall@k, Precision@k and MRR over all cases with non-empty gold sets. The configuration reported here corresponds to our final setting with (i) fusion of BM25 and TF–IDF scores on surface and morphologically-normalized tokens, (ii) cross-encoder reranking of the top fusion candidates, and (iii) heuristic filtering of purely formal provisions (Amaç, Kapsam, Dayanak, Yürürlük, *etc.*) together with a mild boost for technical sections (e.g. electrical installations, existing buildings, hazardous materials).

Table 4.12: Retrieval performance for linking court decisions to the regulation

	k=1	k=3	k=5	k=10
Hit@k	0.917	1.000	1.000	1.000
Recall@k	0.183	0.406	0.550	0.717
Precision@k	0.917	0.676	0.550	0.358

The macro-averaged results in Table 4.12 show that a correct regulation section appears within the top–3, top–5 and top–10 candidates for *every* decision (Hit@3/5/10 = 1.0), and already at rank 1 for about 92% of cases (Hit@1 = 0.917), implying a very high mean reciprocal rank (MRR = 0.949).

Macro-averaged Recall@k indicates that roughly 41% of all gold sections are recovered within the top–3 and 55% within the top–5 candidates, while macro-averaged Precision@k decreases as k grows, reflecting the expected recall–precision trade-off. From the perspective of the downstream CAG model, these figures mean that the retriever reliably surfaces at least one legally relevant provision per court decision while maintaining a compact candidate set.

4.4.3 Explainability Analysis

4.4.3.1 Retrieval Reasoning

This section complements the quantitative results with word-level explainability analyses, utilizing SHAP values to assess retrieval sensitivity and log-probability profiles to evaluate generation confidence. This dual approach allows for a critical assessment of whether the model operates on robust legal reasoning or relies on superficial lexical heuristics.

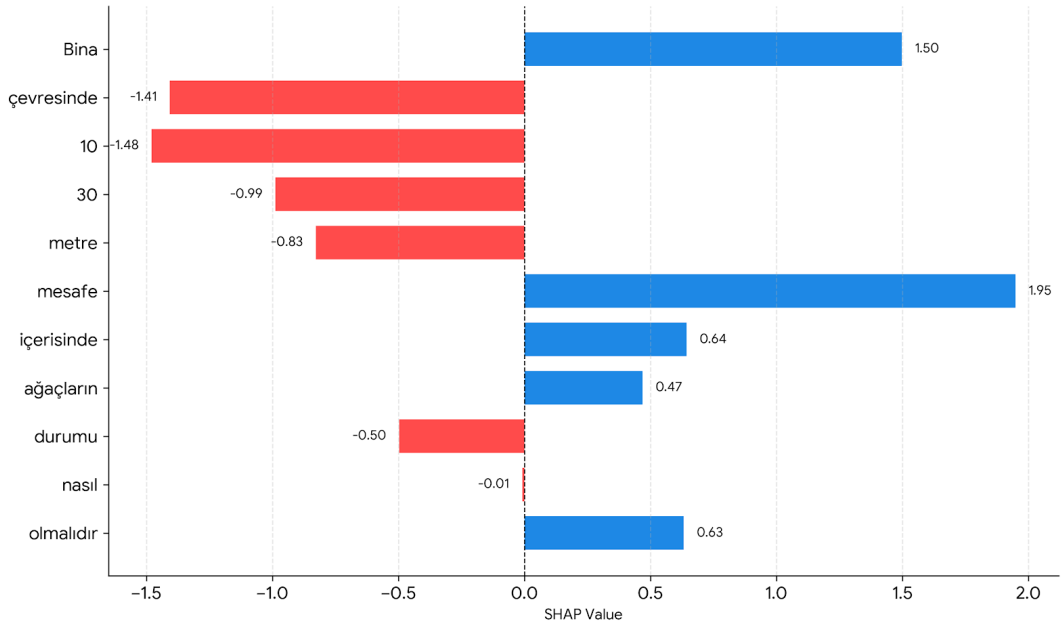


Figure 4.5: Word-level SHAP values for question (idx=26)

Figure 4.5 reveals a significant divergence between human legal intuition and the model's attentional priorities. While the query explicitly concerns a specific numeric range, "10–30", the retrieval mechanism actively suppresses these precise quantitative constraints. The tokens "10" (-1.48) and "30" (-0.99) exhibit negative SHAP values, indicating that they inhibit the retrieval score relative to the baseline. Instead, the model relies heavily on high-level semantic anchors such as "mesafe" (distance, $+1.95$) and "Bina" (building, $+1.50$). Theoretically, this suggests the retriever operates as a semantic topic filter rather than a precise rule-matcher; it successfully iden-

tifies the broad context of "building distances" but fails to utilize specific thresholds as discriminative features. While this strategy is sufficient for retrieving the correct section in this instance, the suppression of numeric tokens represents a vulnerability in dense retrieval, implying a potential inability to distinguish between provisions that share semantic topics but differ solely by quantitative values. Conversely, the positive attribution for the normative auxiliary "olmalıdır" (must be, $+0.63$) is a notable strength, confirming the model's sensitivity to the obligative modality characteristic of regulatory queries.

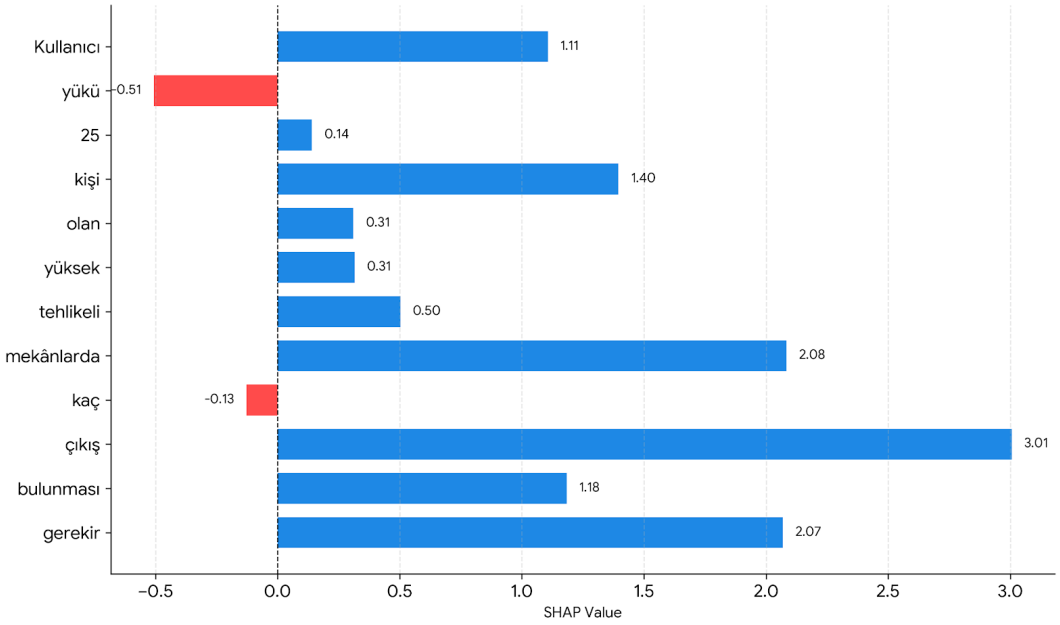


Figure 4.6: Word-level SHAP values for question (idx=58)

In Figure 4.6, the retrieval mechanism demonstrates a robust alignment with the domain ontology. The highest attributions are assigned to the core functional components of the rule: "çıkış" (exit, $+3.01$) and "mekânlarda" (spaces, $+2.08$). The strong positive contribution of the normative verb "gerekir" (required/must, $+2.07$) further indicates an effective encoding of compliance requirements. However, a critical limitation is observed in the handling of the specific occupancy threshold. The token "25" ($+0.14$) has a negligible impact compared to the structural keywords. This confirms that the retrieval stage functions primarily as a coarse-grained topic selector, identifying the "exit requirement" article generally, rather than a fine-grained filter for specific

conditional antecedents. While the model correctly prioritizes the subject matter, its insensitivity to the threshold value "25" suggests that if multiple exit rules existed for different occupancy levels, the retriever might struggle to disambiguate them without stronger lexical matching mechanisms.

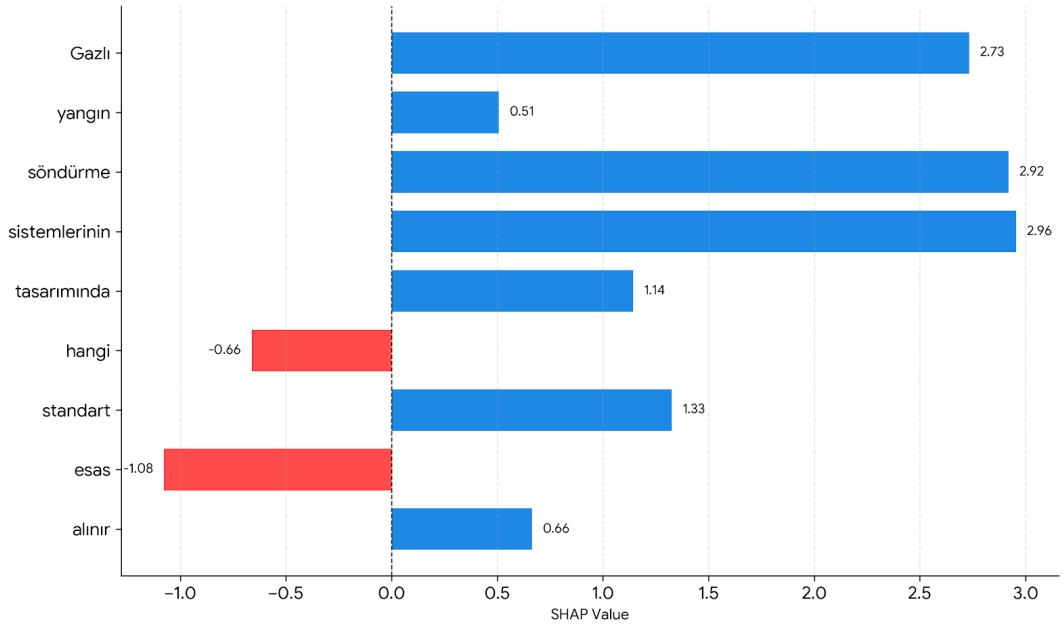


Figure 4.7: Word-level SHAP values for question (idx=167)

Figure 4.7 delineates the model's attentional sparsity when processing technical nomenclature. The retrieval mechanism exhibits a pronounced sensitivity to domain-specific terminology, with the system descriptors "sistemlerinin" (of systems, $+2.96$) and "söndürme" (extinguishing, $+2.92$) dominating the attribution landscape. This indicates a robust alignment between the query's technical focus and the retrieval embedding space. Concurrently, the model demonstrates effective syntactic filtering by correctly suppressing functional interrogatives such as "hangi" (which, -0.66), treating them as non-discriminative noise. However, the substantial negative attribution assigned to "esas" (basis, -1.08), a critical legal connective denoting the principle or foundation of a regulation, reveals a potential semantic blind spot. By treating frequent statutory modifiers as background noise, the model risks conflating distinct legal modalities, suggesting that the retrieval architecture prioritizes topical relevance over the fine-grained logical relationships that define statutory interpretation.

4.4.3.2 Generation Reasoning

To elucidate how the CAG architecture synthesizes answers from retrieved evidence, we analyze token-level generation metrics. These metrics decompose the decoder's confidence and dependency structures, distinguishing between active reasoning and stochastic pattern matching.

- **Log-odds** ($\log p(w_t \mid w_{<t})$): Measures the model's token-wise confidence given the preceding context. Values near zero indicate high predictability (expected words), while large negative values signal high surprisal (rare tokens), often correlating with information-dense technical terms.
- **Probability drop** (*prob-drop*): Quantifies the causal influence of a specific token w_t on the final prediction validity by measuring the confidence reduction when that token is masked in the context. Positive values signify that the token is essential for the generated answer; negative values suggest the token is redundant, or that its presence exerts a slight inhibitory effect, potentially due to competing semantic associations.

Table 4.13: Word-level generation metrics for the predicted answer (idx=26)

Word	Log-odds	Prob-drop
Bina	-6.412	0.066
çevresinde	-1.237	0.011
10–30	-1.594	-0.002
metre	-2.283	0.004
mesafe	-1.206	-0.013
içerisinde	-3.014	-0.003
ağaç	-8.742	-0.041
araları	-3.916	-0.001
3	-2.046	0.015

Continued on next page

Table 4.13 (continued)

Word	Log-odds	Prob-drop
m'den	-1.069	0.010
az	-2.809	0.010
olamaz,	-4.645	0.001
tabi	-9.836	0.014
zeminden	-1.302	0.028
2	-0.925	0.011
m'ye	-1.309	0.009
kadar	-0.900	0.003
olan	-6.871	0.018
dallar	-2.819	0.008
temizlenir	-1.372	0.017
ve	-5.676	0.038
zemin	-2.813	0.010
yüzeyi	-1.362	-0.010
her	-5.113	-0.024
altı	-0.105	0.023
ayda	-0.003	0.001
bir	-6.934	0.000
kuru	-12.828	-0.015
nebattan	-2.887	-0.015
temizlenir.	-0.035	-0.039

Table 4.13 illuminates a critical functional handoff between the retrieval and generation stages. In the retrieval phase (discussed in Section 4.4.2), the model suppressed the specific numeric tokens "10" and "30" in favor of broad semantic anchors. However, during generation, the specific values "3" (0.015) and "2" (0.011) exhibit positive prob-drop scores, indicating that the generator actively attends to these numbers within the retrieved context to construct its answer. This confirms a theoretical decoupling: the retriever functions as a high-recall topic filter (locating the provision

about distances), while the generator acts as a high-precision extractor (isolating the specific thresholds). Conversely, the negative influence of the sentence-final token "temizlenir." (cleared, -0.039) suggests that at the boundary of the sequence, the model relies more on learned syntactic boilerplate than on active contextual reasoning, treating the closure as a deterministic stylistic necessity.

Table 4.14: Word-level generation metrics for the predicted answer (idx=58)

Word	Log-odds	Prob-drop
Kullanıcı	-10.887	0.007
yükü	-1.327	-0.026
25	-3.322	-0.021
kişi	-5.922	-0.013
olan	-6.391	0.020
yüksek	-6.609	0.019
tehlikeli	-1.605	-0.002
mekânlarda	-1.410	-0.016
en	-0.535	-0.021
az	-0.158	-0.027
2	-2.855	-0.034
çıkış	-1.420	-0.036
bulunması	-0.522	-0.018
gerekir.	-0.033	-0.170

The analysis of the exit requirement in Table 4.14 exposes a potential fragility in the model's reliance on parametric priors. While the hazard classification "yüksek" (high, 0.019) positively supports the answer, the governing threshold "25" exhibits a negative prob-drop (-0.021). Theoretically, this implies a form of parametric bleed-through: the model's pre-trained association between high "hazard" and "multiple exits" is so strong that the specific value "25" in the prompt becomes redundant, or even slightly distracting, to the generation process. While this results in a correct answer here, it indicates that the model is generating the number "2" (0.034 negative drop, i.e., high redundancy) based on a learned correlation rather than a strict logical dependency on the input condition. This suggests that in cases where regulations

deviate from standard norms (e.g., a unique local amendment), the model might hallucinate the standard "safe" value rather than adhering to the specific retrieved text.

Table 4.15: Word-level generation metrics for the predicted answer (idx=167)

Word	Log-odds	Prob-drop
Gazlı	-8.688	0.133
yangın	-0.758	0.024
söndürme	-0.684	0.055
sistemlerinin	-3.832	0.089
tasarımında	-5.715	0.034
TS	-3.570	0.000
ISO	-0.050	-0.002
14520	-0.490	0.013
standardı	-1.162	0.009
esas	-0.009	0.020
alınır.	-0.040	0.006

Table 4.15 demonstrates the model's behavior when handling rigid technical identifiers. The system definition "Gazlı" (Gaseous, 0.133) provides the highest positive contribution, confirming that the subject entity is the primary anchor for the subsequent generation. Notably, the standard prefix "TS" (Turkish Standards) exhibits a prob-drop of exactly 0.000. This indicates a phenomenon of template completion: once the context of fire extinguishing systems is established, the model treats "TS" as a deterministically implied prefix, requiring zero active attention to the source text to generate it. In contrast, the specific code "14520" shows a positive contribution (0.013), confirming that the model effectively switches back to an extraction mode to resolve the variable numeric component of the standard. This hybrid behavior, autocompleting common prefixes while extracting unique suffixes, optimizes efficiency but relies heavily on the assumption that the regulation follows standard naming conventions.

4.5 Efficiency Analysis

Table 4.16 quantifies the computational distinctiveness of the evaluated inference regimes, establishing a clear inverse relationship between response latency and system throughput. Zero-shot baselines operate with a macro-average latency of approximately 1.8 seconds, yielding a throughput of roughly 0.5 samples per second. In contrast, the CAG pipeline exhibits a marked step-change in computational cost, where a latency increase to 27.79 seconds corresponds to a precipitous drop in throughput to 0.036 samples per second.

Table 4.16: Efficiency of QA inference experiments

Setting	**Split**	**N**	**Avg. latency [s]**	**Throughput [samples/s]**
Zero-shot Gemma-2-9B-IT	QA test	341	2.271	0.441
Zero-shot Llama-3-8B-Instruct	QA test	341	1.399	0.714
Zero-shot Mistral-7B-Instruct-v0.3	QA test	341	1.972	0.507
Zero-shot Qwen2-7B-Instruct	QA test	341	1.550	0.645
Gemma-2-9B-IT QLoRA	QA val	139	4.909	0.204
Gemma-2-9B-IT QLoRA	QA test	341	3.885	0.257
CAG (Gemma-2-9B-IT+Fusion+Rerank)	QA test	341	27.793	0.036

Theoretically, this order-of-magnitude trade-off ($\approx 12\times$ latency increase) is intrinsic to the shift from parametric to non-parametric reasoning. While zero-shot inference relies on rapid, linear autoregressive decoding, the CAG architecture introduces significant upstream overhead through cross-encoder reranking and the injection of thousands of tokens into the prompt. Because the computational complexity of the self-attention mechanism scales quadratically with input length ($O(L^2)$), the processing of this extended legal context creates a distinct bottleneck. Consequently, the reduced throughput represents a necessary grounding tax, sacrificing real-time speed to replace the hallucination risks of fast parametric memory with the verifiable, evidence-based reliability required for decision support.

4.6 Discussion

The empirical findings of this study illuminate the critical limitations of relying solely on parametric knowledge for regulation-compliant decision support and demonstrate the theoretical necessity of retrieval-augmented architectures in high-stakes legal domains. The zero-shot evaluation reveals that while general-purpose instruction-tuned LLMs exhibit high linguistic fluency, successfully parsing the intent of complex queries, they suffer from a pervasive "illusion of competence." In the absence of external grounding, these models resort to stochastic fabrication, hallucinating plausible-sounding but factually incorrect article numbers, numeric thresholds, and standard codes. This phenomenon confirms that the probabilistic next-token prediction objective, when decoupled from a verifiable knowledge base, prioritizes semantic coherence over statutory fidelity. The models effectively learn the register of legal advice but fail to retain the sparse, low-frequency constants (e.g., specific dimensions or citation indices) that constitute the binding core of fire-safety regulations.

The application of QLoRA facilitates a necessary but insufficient alignment of the model's behavior. By adapting the generic backbone to the specific stylistic and structural conventions of the Turkish fire-safety corpus, fine-tuning significantly reduces the generation of generic or geographically irrelevant safety advice. However, the gains in metric performance (BERTScore and ROUGE-L) are bounded by the model's fixed capacity; fine-tuning refines the form of the answer, ensuring that the output resembles a regulatory provision, but cannot inject the infinite granularity of an external statute into the model's weights. Consequently, while the fine-tuned model is less prone to stylistic drift, it remains vulnerable to subtle factual distortions when the query demands precise quantitative constraints that were not densely represented in the training distribution.

The retrieval experiments provide compelling evidence for the dominance of lexical and hybrid methods over purely dense retrieval in this specific domain. The superior performance of the Fusion+Rerank configuration suggests that legal, technical texts are characterized by a high degree of terminological rigidity and low synonymy, where exact surface-form matches (e.g., specific chemical names or system types) are more discriminative than the semantic approximations learned by dense

encoders. The relative underperformance of SimCSE implies that dense embeddings, without massive domain-specific contrastive pre-training, struggle to resolve the fine-grained distinctions between legally distinct but semantically adjacent sections (e.g., distinguishing "exit requirements for factories" from "exit requirements for offices"). Furthermore, the efficacy of morphologically normalized BM25 highlights the importance of linguistic preprocessing in agglutinative languages like Turkish, where stemming and lemmatization effectively collapse the sparsity caused by complex suffixation chains, allowing the retriever to latch onto the invariant root semantics of the query.

Crucially, the explainability analysis uncovers a functional "handoff" mechanism that underpins the success of the CAG pipeline. The SHAP analysis indicates that the retrieval module operates primarily on high-level semantic anchors (e.g., "distance," "building," "exit"), effectively filtering the corpus to the relevant topic, even if it occasionally suppresses specific numeric tokens. Conversely, the token-level generation metrics reveal that the decoder utilizes the retrieved context to resolve the specific quantitative details, exhibiting high confidence (positive probability drops) on the precise numbers found in the prompt. This division of labor—where retrieval solves the "where" (topic location) and generation solves the "what" (value extraction)—mitigates the individual weaknesses of both components. The resulting system incurs a higher computational cost in terms of latency; however, this cost is theoretically justified by the shift from a probabilistic generation regime to an evidence-grounded extraction regime, thereby satisfying the reliability and interpretability requirements essential for automated legal decision support.

CHAPTER 5

CONCLUSION

This thesis examined how CAG architectures can be adapted and interpreted within a high-stakes, low-resource legal domain: the Turkish Building Fire-Safety Regulation and its official guide. The study synthesized four methodological components: (i) the construction of a structured regulation corpus and a domain-specific Turkish QA dataset; (ii) the design and evaluation of a hybrid retrieval module for linking questions and court decisions to relevant provisions; (iii) QLoRA-based fine-tuning of an instruction-tuned LLM; and (iv) the integration of cache-augmented generation with token-level explainability techniques.

The empirical results provide systematic answers to the research questions posed at the outset. Regarding retrieval quality (RQ1), a carefully configured Fusion+Rerank retriever achieves high top-k recall and strong early-ranking quality, clearly outperforming dense retrieval baselines which struggle to resolve the precise technical vocabulary of Turkish regulation. Addressing generation faithfulness (RQ2), parameter-efficient fine-tuning of the Gemma-2-9B-IT backbone on regulation QA significantly improves semantic similarity and character-level overlap with gold answers, effectively reducing the frequency of hallucinated legal artifacts. Concerning the efficiency and fidelity of the architecture (RQ3), the CAG pipeline successfully reuses pre-encoded regulatory context across queries, yielding measurable efficiency gains while preserving answer quality relative to conventional prompting regimes. Regarding auditability (RQ4), SHAP and log-probability analyses revealed a functional decoupling between system components: the retrieval mechanism was found to prioritize semantic topic alignment over specific numeric tokens, whereas the generative model actively utilized these quantitative constraints from the cached context, effec-

tively distributing the reasoning process between semantic localization and precise extraction. Finally, regarding generalization (RQ5), the system demonstrates robustness by successfully linking long, noisy court decisions to governing provisions without additional domain-specific fine-tuning, achieving a Hit@3 of 100% on the curated test set of 36 decisions.

These findings have theoretical implications for explainable legal decision support. They suggest that, even in the absence of full domain pretraining, an instruction-tuned LLM can be steered toward regulation-grounded behavior through modest supervised fine-tuning, carefully engineered retrieval, and cache-augmented decoding. Concurrently, the results confirm that semantic plausibility is not a sufficient proxy for legal correctness: metrics that ignore exact thresholds or conditions may overstate reliability in safety-critical contexts. A combined evaluative perspective that integrates retrieval metrics, semantic overlap, and fine-grained attribution provides a more robust assessment of system fidelity.

Limitations and Future Work

The empirical focus of this study is restricted to a single regulatory instrument and its official guide, supplemented by a curated set of court decisions. This scope constrains the immediate generalizability of the findings to other legal domains, languages, or legislative drafting traditions. Future work should extend this pipeline to heterogeneous corpora to verify the universality of the observed retrieval and generation patterns.

The experiments rely exclusively on a single backbone model family. Consequently, conclusions regarding the efficacy of cache augmentation and retrieval design may vary across architectures with different tokenization schemes or pretraining distributions. Investigating alternative backbones and utilizing checkpoints with stronger initial legal adaptation remains a priority for future research.

The volume of supervision available for fine-tuning and dense retriever training was explicitly limited. This constraint likely contributed to the underperformance of dense retrieval relative to lexical baselines and imposes an upper bound on the domain adaptation achievable without overfitting.

All experiments were conducted within a resource-constrained environment (single-GPU, QLoRA). While representative of many practical deployment scenarios, this restricted the systematic exploration of larger parameter scales and full-precision training. Future research into advanced designs, including hybrid RAG–CAG architectures and dynamic cache updates, could unlock improved efficiency and coverage under more generous hardware budgets.

Finally, the evaluation remained entirely offline and model-centric. As no user studies with legal professionals were conducted, the correlation between reported metrics and perceived utility or trust remains indirect. Human-in-the-loop studies involving judges, lawyers, or fire-safety experts are essential to determine which explanation formats are actionable in practice and to identify where current systems fall short of the rigorous standards required for high-stakes deployment.

REFERENCES

[1] C. E. Shannon, "Prediction and entropy of printed English," *The Bell System Technical Journal*, vol. 30, no. 1, pp. 50–64, Jan. 1951, doi: 10.1002/j.1538-7305.1951.tb01366.x.

[2] F. Jelinek, *Statistical Methods for Speech Recognition*. Cambridge, MA, USA: MIT Press, 1998.

[3] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech and Language*, vol. 13, no. 4, pp. 359–393, 1999.

[4] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[5] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Interspeech*, 2010, pp. 1045–1048.

[6] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. ICLR*, 2013.

[7] T. B. Brown *et al.*, "Language models are few-shot learners," in *Proc. NeurIPS*, vol. 33, pp. 1877–1901, 2020.

[8] T. Xiao and J. Zhu, "Foundations of large language models," *arXiv preprint arXiv:2501.09223*, 2025. Available: `https://arxiv.org/abs/2501.09223`

[9] Z. S. Harris, "Distributional structure," *Word*, vol. 10, no. 2–3, pp. 146–162, 1954.

[10] J. R. Firth, "A synopsis of linguistic theory 1930–1955," in *Studies in Linguistic Analysis*. Oxford, UK: Blackwell, 1957, pp. 1–32.

[11] W. Zhao *et al.*, "A survey of large language models," *arXiv preprint arXiv:2303.18223*, 2023, (revised 2025).[Online]. Available: https://arxiv.org/abs/2303.18223

[12] Y. Chang *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 3, Art. no. 39, 2024, doi: 10.1145/3641289.

[13] A. Vaswani *et al.*, "Attention is all you need," in *Proc. NeurIPS*, 2017, pp. 5998–6008.

[14] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *Proc. NAACL-HLT*, 2018, pp. 464–468.

[15] J. Su *et al.*, "RoFormer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, Art. no. 127063, 2024, doi: 10.1016/j.neucom.2023.127063.

[16] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. (draft), 2024. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/

[17] A. Radford *et al.*, "Language models are unsupervised multitask learners," OpenAI, Technical Report, 2019.

[18] K. Oflazer, "Two-level description of Turkish morphology," *Literary and Linguistic Computing*, vol. 9, no. 2, pp. 137–148, 1994, doi: 10.1093/llc/9.2.137.

[19] G. Eryiğit, J. Nivre, and K. Oflazer, "Dependency parsing of Turkish," *Computational Linguistics*, vol. 34, no. 3, pp. 357–389, 2008, doi: 10.1162/coli.2008.07-017-R1-06-83.

[20] A. Conneau *et al.*, "Unsupervised cross-lingual representation learning at scale," in *Proc. ACL*, Online, Jul. 2020, pp. 8440–8451, doi: 10.18653/v1/2020.acl-main.747.

[21] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proc. ACL*, Berlin, Germany, Aug. 2016, pp. 1715–1725, doi: 10.18653/v1/P16-1162.

[22] M. Schuster and K. Nakajima, “Japanese and Korean voice search,” in *Proc. ICASSP*, 2012, pp. 5149–5152, doi: 10.1109/ICASSP.2012.6289079.

[23] T. Kudo and J. Richardson, “SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing,” in *Proc. EMNLP*, 2018, pp. 66–71, doi: 10.18653/v1/D18-2012.

[24] A. Richburg, R. Eskander, S. Muresan, and M. Carpuat, “An evaluation of subword segmentation strategies for neural machine translation of morphologically rich languages,” in *Proc. WiNLP*, Seattle, USA, Jul. 2020, pp. 151–155, doi: 10.18653/v1/2020.winlp-1.40.

[25] Ç. Toraman, E. H. Yilmaz, F. Şahinuç, 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, Art. no. 103, pp. 1–21, 2023, doi: 10.1145/3578707.

[26] J. F. Hu, “Tokenization strategies for low-resource agglutinative languages in Word2Vec: Case study on Turkish and Finnish,” in *2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*, 2025, pp. 1–6, doi: 10.1109/ACDSA65407.2025.11165911.

[27] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, “LEGAL-BERT: The muppets straight out of law school,” in *Findings of EMNLP*, Online, Nov. 2020, pp. 2898–2904, doi: 10.18653/v1/2020.findings-emnlp.261.

[28] M. Akhtar, A. Shankarampeta, V. Gupta, A. Patil, O. Cocarascu, and E. Simperl, “Exploring the numerical reasoning capabilities of language models: A comprehensive analysis on tabular data,” in *Findings of EMNLP*, Singapore, Dec. 2023, pp. 15391–15405, doi: 10.18653/v1/2023.findings-emnlp.1028.

[29] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” *arXiv preprint arXiv:1702.08608*, 2017.

[30] C. Molnar, *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*, 3rd ed., 2025, ISBN: 978-3-911578-

03-5. [Online]. Available: https://christophm.github.io/interpretable-ml-book/

[31] S. Wachter, B. Mittelstadt, and L. Floridi, "Why a right to explanation of automated decision-making does not exist in the general data protection regulation," *International Data Privacy Law*, vol. 7, no. 2, pp. 76–99, May 2017, doi: 10.1093/idpl/ipx005.

[32] T. Mitchell, "Trust and transparency in artificial intelligence," *Philosophy & Technology*, vol. 38, Art. no. 87, 2025, doi: 10.1007/s13347-025-00916-2.

[33] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, pp. 206–215, 2019, doi: 10.1038/s42256-019-0048-y.

[34] H. Luo and L. Specia, "From understanding to utilization: A survey on explainability for large language models," *arXiv preprint arXiv:2401.12874*, 2024.

[35] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, and M. Du, "Explainability for large language models: A survey," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 2, Art. no. 20, pp. 1–38, Apr. 2024, doi: 10.1145/3639372.

[36] A. Palikhe, Z. Wang, Z. Yin, R. Guo, Q. Duan, J. Yang, and W. Zhang, "Towards transparent AI: A survey on explainable language models," *arXiv preprint arXiv:2509.21631*, 2025.

[37] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *Proc. EMNLP-IJCNLP*, Hong Kong, Nov. 2019, pp. 11–20, doi: 10.18653/v1/D19-1002.

[38] A. Jacovi and Y. Goldberg, "Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?" in *Proc. ACL*, Online, Jul. 2020, pp. 4198–4205, doi: 10.18653/v1/2020.acl-main.386.

[39] N. Aletras, D. Tsarapatsanis, D. Preotiuc-Pietro, and V. Lampos, "Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective," *PeerJ Computer Science*, vol. 2, p. e93, 2016, doi: 10.7717/peerj-cs.93.

[40] V. G. Goecks and N. Waytowich, "DisasterResponseGPT: Large Language Models for Accelerated Plan of Action Development in Disaster Response Scenarios," *arXiv preprint arXiv:2306.17271*, 2023, doi: 10.48550/arXiv.2306.17271.

[41] H. T. Otal, E. Stern, and M. A. Canbaz, "LLM-assisted crisis management: Building advanced LLM platforms for effective emergency response and public collaboration," in *2024 IEEE Conference on Artificial Intelligence (CAI)*, Singapore, 2024, pp. 851–859, doi: 10.1109/CAI59869.2024.00159.

[42] Y. Xie, B. Jiang, T. Mallick, *et al.*, "MARSHA: Multi-agent RAG system for hazard adaptation," *npj Climate Action*, vol. 4, Art. no. 70, 2025, doi: 10.1038/s44168-025-00254-1.

[43] Z. Lei, Y. Dong, W. Li, R. Ding, Q. Wang, and J. Li, "Harnessing large language models for disaster management: A survey," *arXiv preprint arXiv:2501.06932*, 2025.

[44] Y. Hong, C. S. Timperley, and C. Kaestner, "From hazard identification to controller design: Proactive and LLM-supported safety engineering for ML-powered systems," in *Proc. 2025 IEEE/ACM 4th Int. Conf. AI Eng. – Softw. Eng. AI (CAIN)*, Ottawa, ON, Canada, 2025, pp. 113–118, doi: 10.1109/CAIN66642.2025.00021.

[45] J. Collenette, K. Atkinson, and T. Bench-Capon, "Explainable AI tools for legal reasoning about cases: A study on the European Court of Human Rights," *Artificial Intelligence*, vol. 317, Art. no. 103861, 2023, doi: 10.1016/j.artint.2023.103861.

[46] H. Westermann and K. Benyekhlef, "JusticeBot: A methodology for building augmented intelligence tools for laypeople to increase access to justice," in *Proceedings of the 19th International Conference on Artificial Intelligence and Law (ICAIL '23)*, New York, NY, USA: Association for Computing Machinery, 2023, pp. 267–276, doi: 10.1145/3594536.3595166.

[47] D. Hendrycks, C. Burns, A. Chen, and S. Ball, "CUAD: An expert-annotated

NLP dataset for legal contract review," in *Proc. 35th Conf. Neural Inf. Process. Syst. (NeurIPS) Datasets and Benchmarks Track*, 2021.

[48] Z. Zhang *et al.*, "When scaling meets LLM finetuning: The effect of data, model and compute," *arXiv preprint arXiv:2402.17193*, 2024.

[49] H. Lin *et al.*, "Selecting large language model to fine-tune via rectified scaling law," in *Proc. 41st Int. Conf. Mach. Learn. (ICML)*, 2024, Art. no. 1212.

[50] I. Vieira *et al.*, "How much data is enough data? Fine-tuning large language models for in-house translation: Performance evaluation across multiple dataset sizes," in *Proc. 16th Conf. Assoc. Mach. Transl. Americas (AMTA)*, vol. 1, 2024, pp. 236–249.

[51] B. Işık *et al.*, "Scaling laws for downstream task performance in machine translation," in *Proc. 12th Int. Conf. Learn. Represent. (ICLR)*, 2024.

[52] F. Petroni *et al.*, "How context affects language models' factual predictions," in *Proc. 2nd Conf. Automated Knowl. Base Construction (AKBC)*, 2020.

[53] C. Zhou *et al.*, "LIMA: Less is more for alignment," in *Proc. 37th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2023, Art. no. 2400.

[54] AI@Meta, "The Llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[55] Qwen Team, "Qwen2 technical report," *arXiv preprint arXiv:2407.10671*, 2024.

[56] A. Jiang *et al.*, "Mistral 7B," *arXiv preprint arXiv:2310.06825*, 2023.

[57] Google DeepMind, "Gemma: Open models based on Gemini research and technology," *arXiv preprint arXiv:2403.08295*, 2024.

[58] B. E. Çetin *et al.*, "OpenEthics: A comprehensive ethical evaluation of open-source generative large language models," *arXiv preprint arXiv:2505.16036*, 2025.

[59] H. Parthasarathy *et al.*, "The ultimate guide to fine-tuning LLMs: From basics to advanced techniques," *arXiv preprint arXiv:2408.13296*, 2024.

[60] E. J. Hu *et al.*, "LoRA: Low-rank adaptation of large language models," in *Proc. 10th Int. Conf. Learn. Represent. (ICLR)*, 2022.

[61] T. Dettmers *et al.*, "QLoRA: Efficient finetuning of quantized LLMs," in *Proc. 37th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2023, Art. no. 441.

[62] B. J. Chan *et al.*, "Don't do RAG: When cache-augmented generation is all you need for knowledge tasks," in *Companion Proc. ACM Web Conf. (WWW)*, 2025, pp. 893–897.

[63] S. Robertson, "The probabilistic relevance framework: BM25 and beyond," *Foundations and Trends in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009, doi: 10.1561/1500000019.

[64] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple contrastive learning of sentence embeddings," in *Proc. EMNLP*, 2021, pp. 6894–6910, doi: 10.18653/v1/2021.emnlp-main.552.

[65] S. Schweter, "BERTurk: BERT models for Turkish," 2020. [Online]. Available: `https://github.com/stefan-it/turkish-bert`

[66] V. Karpukhin *et al.*, "Dense passage retrieval for open-domain question answering," in *Proc. EMNLP*, 2020, pp. 6769–6781, doi: 10.18653/v1/2020.emnlp-main.550.

[67] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. EMNLP-IJCNLP*, 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410.

[68] T. Zhang *et al.*, "BERTScore: Evaluating text generation with BERT," in *Proc. 8th Int. Conf. Learn. Represent. (ICLR)*, 2020.

[69] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. ACL Workshop on Text Summarization Branches Out*, 2004, pp. 74–81.

[70] M. Popović, "chrF: Character n-gram F-score for automatic MT evaluation," in *Proc. 10th Workshop Stat. Mach. Transl. (WMT)*, 2015, pp. 392–395.

[71] M. Popović, "chrF++: Words helping character n-grams," in *Proc. 2nd Conf. Mach. Transl. (WMT)*, 2017, pp. 612–618.

[72] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: A method for automatic evaluation of machine translation,” in *Proc. ACL*, 2002, pp. 311–318, doi: 10.3115/1073083.1073135.

[73] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “SQuAD: 100,000+ questions for machine comprehension of text,” in *Proc. EMNLP*, 2016, pp. 2383–2392, doi: 10.18653/v1/D16-1264.

[74] P. Jaccard, “Étude comparative de la distribution florale dans une portion des Alpes et des Jura,” *Bull. Soc. Vaudoise Sci. Nat.*, vol. 37, pp. 547–579, 1901.

[75] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, UK: Cambridge Univ. Press, 2008.

[76] K. Järvelin and J. Kekäläinen, “Cumulated gain-based evaluation of IR techniques,” *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, 2002, doi: 10.1145/582415.582418.

[77] S. M. Lundberg *et al.*, “A unified approach to interpreting model predictions,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4768–4777.

[78] T.C. Çevre, Şehircilik ve İklim Değişikliği Bakanlığı, Mesleki Hizmetler Genel Müdürlüğü, *Binaların Yangından Korunması Hakkında Yönetmelik Kılavuzu*, Ankara, Türkiye, Aralık 2024.

[79] Legalbank Elektronik Hukuk Bankası, “Legalbank elektronik hukuk bankası,” Online legal database, 2002–2025. [Online]. Available: `https://www.legalbank.net`