Artificial Intelligence and Law
https://doi.org/10.1007/s10506-025-09472-8

ORIGINAL RESEARCH

TRACS-LLM: LLM-based traffic accident criminal sentencing prediction focusing on imprisonment, probation, and fines

Hyunsik Min[1] · **Byeongjoon Noh**[2]

Accepted: 26 June 2025
© The Author(s), under exclusive licence to Springer Nature B.V. 2025

Abstract

Fault determination in traffic accidents require a careful analysis of various factors that could influence sentence severity and fairness in judgement. Traditional methods are subjective and often time-consuming, necessitating the need for an objective solution. Recently, large language models (LLMs) have garnered attention in the legal field and incorporating them into the legal process is beneficial. Moreover, there is lack of studies on sentence prediction both in the context of the Korean legal system and LLMs. We propose a traffic accident criminal sentencing prediction method based on large language models (TRACS-LLM), which analyzes legal texts related to traffic accident cases to predict three major legal outcomes, namely the length of imprisonment, whether probation is granted, and the amount of fines, through training from a large dataset of criminal traffic accident cases. A parallel structure that combines A lite bidirectional encoder representations from transformer (ALBERT) and bidirectional long short-term memory (Bi-LSTM) with grouped query attention (GQA) is leveraged to capture both the general language context and domain-specific legal information. In addition, we employ a multi-task learning approach, in which the model simultaneously predicts multiple legal outcomes by sharing information across tasks. We validated the feasibility and applicability of the proposed method through two ablation experiments, comparing the performance of the proposed model against baseline models and evaluating the impact of the proposed parallel combination with ALBERT. The results demonstrated that the proposed ALBERT+Bi-LSTM with GQA model achieved superior performance across all tasks, yielding the lowest RMSE for imprisonment and fine predictions and the highest F1-score for probation prediction. This study provides valuable insights for legal professionals by offering a data-driven, objective approach to sentencing, with potential applications beyond the Korean legal domain.

Keywords Artificial intelligence in law · Large Language Models · Sentencing prediction · Criminal traffic accident · Legal context understanding

Extended author information available on the last page of the article

Published online: 23 July 2025

1 Introduction

Traffic accidents occur owing to various factors, such as driver negligence, road conditions, weather, and vehicle defects (Perrels et al. 2015; Lee and Mannering 2002). As such accidents cause significant economic and social damage, accurate assessments of liability and fault are required (Noh and Yeo 2021, 2022). Traffic accident cases are generally divided into criminal and civil cases, with civil cases typically focusing on claims for damages and criminal cases addressing the legal responsibility of drivers in serious offenses, particularly those involving injury or death. Common examples of criminal traffic offenses include speeding, driving under the influence (DUI), and crossing the centerline. In such cases, drivers may face substantial penalties, including fines, probation, or imprisonment, making the accurate prediction and determination of appropriate sentences critical.

The determination of fault involves a careful analysis of various interrelated factors, such as the circumstances of the accident, witness statements, the vehicle speed and direction, and road conditions (Rolison et al. 2018; Noh et al. 2021, 2022). The assignment of fault can directly influence the severity of the sentence, particularly in criminal cases, and errors in judgment may result in unfair outcomes, such as excessive fines or wrongful imprisonment. Traditional methods for determining fault and sentencing rely heavily on the subjective assessments of legal professionals, which can be time consuming and often lack consistency. An objective, efficient, and consistent method for predicting fault and sentencing outcomes, especially in criminal cases (Jin and Noh 2023), is necessary to address these issues.

The recent advances in large language models (LLMs) offer a promising avenue for addressing these challenges. In this paper, we use the term LLMs in a broad sense to include pretrained Transformer-based models with substantial parameter counts that can learn intricate linguistic patterns from vast corpora. These models encompass both understanding-focused models, such as Bidirectional Encoder Representations from Transformers (BERT) and A Lite BERT (ALBERT), and generative models including GPT-4o and LLaMA (Devlin et al. 2018; OpenAI 2024; Meta AI 2024). Although their architectural goals differ, these models share core capabilities in processing complex natural language, making them highly applicable to legal domains. This inclusive perspective reflects recent works that extends the role of LLMs beyond generation and highlights their versatility across tasks such as translation, information retrieval, and legal analysis (Naveed et al. 2023; Min et al. 2023).

The application of LLM-based approaches to the legal domain has received increasing interest. Legal language is often filled with complex terminology and logic-based structures, making LLMs particularly suitable for this field (Shu et al. 2024). Incorporating LLMs into legal processes can significantly enhance the efficiency of legal professionals, such as judges and lawyers, by accurately interpreting natural language inputs and generating appropriate legal responses. This can reduce the need for time-consuming manual reviews of extensive legal documents and offer new insights by providing critical details and perspectives in complex cases.

Recent developments in the legal field suggest that LLMs can improve legal judgment prediction and handle various legal tasks efficiently (Shu et al. 2024; Shui et al. 2023). Specifically, determining fault in traffic accidents requires an in-depth

analysis of numerous factors, and incorporating LLMs can make this process more objective and streamlined. For example, given input such as "A first-time drunk driver with a blood alcohol concentration (BAC) of about 0.12% exceeded the designated speed limit by approximately 25 km/h in a school zone and struck a child who was crossing the crosswalk without following the traffic signal, resulting in the child's death," LLMs can predict appropriate legal outcomes, such as the imprisonment duration (in months), whether probation should be granted, and fines. These predictions can serve as valuable support for judges in sentencing, assist prosecutors in recommending penalties, and aid defense attorneys in formulating legal strategies. Previous studies have primarily addressed case outcome prediction or legal consultation support, and relatively few works have directly tackled the task of sentencing prediction.

To fill this gap, we propose a novel framework named TRACS-LLM, which integrates legal domain modeling and language modeling in a unified architecture that is specifically designed for criminal traffic accident sentencing prediction. This method involves fine-tuning a pre-trained language model (specifically ALBERT) and applying additional model structuring for a deep understanding of legal language, context, and specific details of traffic accidents, thereby enabling the legal context and regulations to be reflected more precisely. Although the individual components such as ALBERT and bidirectional long short-term memory (Bi-LSTM) are well known, our method introduces a parallel structure that combines these elements with a grouped query attention (GQA) mechanism and multi-task learning. This architecture is tailored to reflect the multi-dimensional nature of sentencing in the Korean criminal legal context. To the best of our knowledge, no prior work has specifically modeled multiple sentencing components, namely imprisonment, probation, and fines, in an integrated manner for criminal traffic accident cases. This modeling strategy directly addresses the overlooked complexity and real-world demands of such sentencing tasks.

TRACS-LLM is trained to predict three key sentencing components simultaneously; that is, imprisonment duration, probation status, and fines, based on real-world legal documents. It consists of three main modules: data preprocessing, legal domain-specific embedding, and legal sentencing prediction. First, the preprocessing module extracts traffic accident-related legal judgment data and tokenizes the text. Subsequently, the embedding module generates legal domain-specific vectors using FastText. Finally, the sentencing prediction module applies a parallel architecture that combines ALBERT and Bi-LSTM with GQA to produce predictions for all three outcomes. This design departs from conventional stacking or sequential models by maintaining separate paths for global and task-specific processing, which are selectively integrated through attention.

The key contributions of this study are as follows: (1) We propose a multi-output sentencing prediction model that predicts imprisonment, probation, and fines simultaneously, which has rarely been explored in prior artificial intelligence (AI) and law research. (2) We develop a parallel architecture that combines ALBERT and Bi-LSTM with a GQA mechanism to extract both general linguistic and domain-specific cues. (3) The GQA mechanism is used to enhance the computational focus on case-relevant legal information. (4) A multi-task learning strategy is employed to

allow shared learning across interrelated sentencing tasks. By providing structured and interpretable predictions, the proposed TRACS-LLM is expected to aid legal professionals in improving the fairness, efficiency, and consistency of criminal traffic sentencing decisions.

2 Related work

This section provides an overview of prior research in two relevant areas: (1) natural language processing (NLP) techniques applied to the legal domain, and (2) sentencing prediction using LLMs. We also contextualize our proposed TRACS-LLM approach by comparing it with representative LLM-based legal studies.

2.1 NLP in law

NLP has become a critical component in legal AI applications, facilitating the analysis, retrieval, and summarization of complex legal texts. Early legal NLP systems were limited to keyword-based searches, but recent developments in Transformer-based architectures have significantly improved the ability to capture the semantic and contextual relationships within statutes, case law, and legal rulings (Zhou et al. 2023; Goodson and Lu 2023; Sun 2023; Katz et al. 2017; Greco and Tagarelli 2024).

One key area of progress lies in legal judgment summarization. Recent models combine extractive and generative summarization strategies to preserve essential legal content while improving the readability and reducing the length (Benedetto et al. 2025). This hybrid strategy is particularly effective in multilingual and comparative legal settings.

Legal NLP also contributes to tasks that bridge textual interpretation and quantitative decision-making, such as sentencing. For example, increasing efforts are being made to encode qualitative legal reasoning, including aggravating and mitigating circumstances, into structured formats to support fairness, consistency, and transparency (Rodríguez Rodríguez et al. 2024). The need for explainable and trustworthy AI has further accelerated NLP adoption across applications such as legal analytics, trial assistance, and policy evaluation (Wei et al. 2025).

2.2 Sentencing prediction using LLMs

The prediction of sentencing outcomes has traditionally relied on structured data and conventional machine learning models such as Random Forest, which are valued for their interpretability and robustness (Katz et al. 2017). However, these approaches are often constrained in handling unstructured legal texts and complex linguistic reasoning.

The emergence of LLMs has enabled more sophisticated methods for modeling legal prediction tasks. Pretrained Transformer models that have been fine-tuned on legal corpora can learn nuanced legal semantics and statutory logic. For example, LawLLM (Shu et al. 2024) applies in-context learning to appellate documents for

outcome forecasting, while other methods incorporate temporal motifs (Cao et al. 2024) and fine-tuning on legal benchmarks such as DISC-Law-Eval (Yue et al. 2023; Satterfield et al. 2025) and LegalBench.

Despite these advances, most prior studies treated sentencing prediction as a single-label classification problem, overlooking the complex structure of real-world sentencing decisions. Few works have attempted to model multiple sentencing outcomes, such as imprisonment duration, probation status, and fines, simultaneously within an integrated learning framework.

To fill this gap, our study proposes TRACS-LLM, which is a parallel-structured, multi-task learning model that is tailored for criminal traffic accident sentencing in Korea. TRACS-LLM combines ALBERT and Bi-LSTM with GQA, capturing both domain-general and case-specific legal cues. It performs multi-output prediction through regression and classification branches to model the interdependencies among sentencing components.

Table 1 prediction tasks, datasets, and modeling strategies. In contrast to general outcome prediction or legal assistance tasks, to the best of our knowledge, our study is the first to address multi-dimensional sentencing prediction using LLMs in the Korean legal domain directly.

3 Methodology

In this section, we describe the proposed TRACS-LLM. As shown in Fig. 1, The proposed method mainly consists of three modules: 1) preprocessing, 2) legal domain-specific embedding, and 3) legal sentencing prediction. First, the preprocessing module structures the legal outcomes related to sentencing from the textual cases and builds a word vocabulary through word tokenization. Subsequently, the legal

Table 1 Comparison with studies using LLMs in other legal domains

Ref.	Prediction target	Datasets	Methods
Shu et al. (2024)	Defendant wins, Plaintiff wins, Settlement, Case dismissal	CaseLaw (Harvard)	In-context learning
Cao et al. (2024)	Case outcomes	ECHR2023	Temporal pattern mining
Cui et al. (2023)	Legal assistant	LawBench, Qualification Exam	Mixture-of-Experts
Yue et al. (2023)	Case outcomes	DISC-Law-Eval	Fine-tuning
Satterfield et al. (2025)	Legal assistant	LegalBench	Fine-tuning
Benedetto et al. (2024)	Case outcomes	Indian Judicial System	Fine-tuning, NER
Ours	**Imprisonment, Probation, Fines**	**Korean criminal traffic accident cases**	**Parallel architecture, GQA, Multi-task learning**

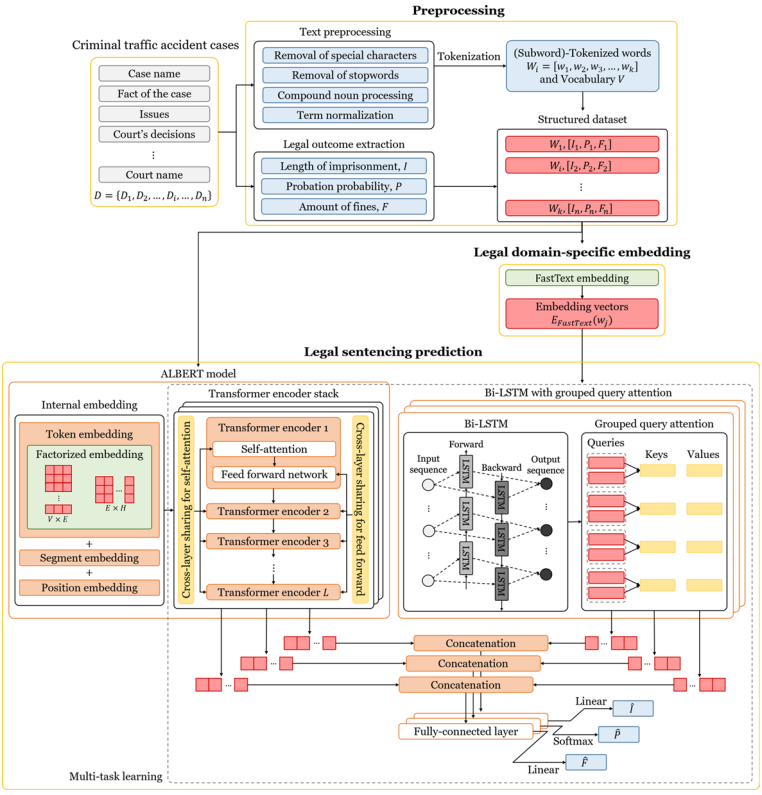

Fig. 1 Overall structure of TRACS-LLM

domain-specific embedding module captures the key legal terminologies through FastText embedding. Finally, the legal sentencing prediction module consists of two parallel models: the ALBERT model and Bi-LSTM with GQA. Each model extracts feature vectors from the given legal cases based on contextual understanding and domain-specific information extraction, and simultaneously predicts three legal outcomes through task-specific layers.

3.1 Preprocessing

The objective of the preprocessing module is to structure sentencing-related legal outcomes from extensive cases regarding criminal traffic accidents and to build a word vocabulary through word tokenization. The target case type in this study is criminal traffic accident cases, which were collected from various local courts in South Korea.

First, as shown in the Preprocessing section of Fig. 1, information regarding the level of legal penalties, such as the length of imprisonment, probation status, and amount of fines, is extracted from the collected criminal traffic accident cases, which reflect criminal traffic law violations. This extraction process undergoes human

observation for all cases. Subsequently, basic text preprocessing steps are performed, including the removal of special characters and stopwords, term normalization, and compound noun processing.

Thereafter, the provided sentences are tokenized to convert the text data into a format that can be processed by the model, with each word mapped to a unique numerical index. The pretrained ALBERT tokenizer (Lan 2019) is employed in this process. ALBERT operates by breaking down the input text into words or subword units and converting them into unique numerical indices using a WordPiece tokenization approach (Song et al. 2020) that splits the given word into smaller subword units if it is not present in the word vocabulary V. This effectively addresses the out-of-vocabulary (OOV) problem by allowing the model to learn and handle meaningful representations of both new and rare words through subword tokenization (Sennrich 2015). A detailed description of the ALBERT model is provided in Section 3.3.

Thus, given the dataset of cases D, where each case is denoted as $D_1, D_2, \ldots, D_n$, each D_i consists of I_i, P_i, F_i (representing the length of imprisonment, probation status, and fines, respectively), the sequence of tokenized words (with subwords) $W_i = [w_1, w_2, \ldots, w_k]$, all of which are mapped to unique numerical indices, and the vocabulary set $V = \{W_1, W_2, \ldots, W_n\}$. These sequences are subsequently transformed into input sequences in the following embeddings for model training and prediction tasks.

3.2 Legal domain-specific embedding

The legal domain-specific embedding module is designed to understand and represent the context and terminology of the legal domain more effectively, specifically in terms of criminal traffic accidents, thereby improving the legal sentencing prediction performance.

Although the pretrained internal embeddings in conventional LLMs such as ALBERT, BERT, and GPT are trained on common text corpora and can capture the context and vocabulary of specific domains to an extent, this study focuses on predicting legal sentencing and not on generating legal domain text. Therefore, the representation of the relationship between legal terminology and legal outcomes must be established.

We adopt the FastText embedding model (Joulin et al. 2016) for the legal domain-specific embedding. Subsequently, the embeddings that are generated by FastText are used as input to the GQA model in the legal sentencing prediction module. FastText represents words as subword-level n-grams, allowing it to capture the internal structure of words. This enables vectors based on both the word itself and its subword units to be generated so that new or rare words that are not encountered in the training data can be handled effectively. As a result, FastText can handle OOV words that the ALBERT internal embeddings cannot process.

In summary, the tokens that are generated during the preprocessing stage are fed into two different embedding pathways: the internal embeddings of the ALBERT model, $E_{ALBERT}(w_j)$, and the legal domain-specific embeddings of FastText, $E_{FastText}(w_j)$. The pretrained ALBERT embeddings focus on understanding and

representing the context of the input text, whereas the FastText-based legal domain-specific embeddings are fed into the GQA model to reflect the specific structure and context of legal documents, with the primary objective of improving the legal outcome prediction.

3.3 Legal sentencing prediction

As illustrated in the Legal sentencing prediction section of Fig. 1, the legal sentencing prediction module is designed with a parallel structure consisting of the ALBERT and Bi-LSTM with GQA models. The tokens that are generated in the preprocessing module are fed into two separate pathways: the ALBERT model and legal domain-specific embedding with FastText, which are processed through the Bi-LSTM with GQA model. The ALBERT model focuses on deeply understanding and representing the contextual information of the input text, whereas the FastText-based legal domain-specific embeddings are input into the GQA model, which specializes in understanding key legal terminologies in legal documents and improving the performance of legal outcome predictions. This parallel configuration allows the model to capture the deep contextual understanding and important vocabulary simultaneously, leading to rich feature extraction from the given textual legal cases.

In addition, to predict the three legal outcomes in this study, I, P, F, the Transformer encoder stack in the ALBERT model and Bi-LSTM with GQA model are designed in three task-specific layers as multi-task heads, thereby enabling multi-task learning.

3.3.1 ALBERT model

ALBERT, which is a lightweight version of the BERT model, applies the factorized embedding parameterization technique to the token embeddings and cross-layer parameter sharing to reduce the number of parameters while maintaining the performance. In addition, the model is structured with multiple Transformer stacks to enable multi-task learning, which is specifically tailored to predict values for the imprisonment duration, probability of probation, and fines. A detailed description of the multi-task learning is provided in Section 3.3.3.

Factorized embedding parameterization In the BERT model, the embedding of input tokens involves three components: token embedding, segment embedding, and position embedding (Vaswani 2017; Devlin et al. 2018). Token embedding converts each input token into a unique vector, whereas segment embedding is used to distinguish to which sentence each token belongs. Position embedding captures the positional information of each token. The vectors that result from applying these embeddings are combined via an element-wise sum and serve as the input to the Transformer encoder stack.

In this process, the token embedding converts each of the $|V|$ tokens in the vocabulary V into an embedding vector with a hidden dimension H. However, when $|V|$ is large, the number of parameters in the embedding matrix increases exponentially, imposing a significant burden on the memory efficiency and training speed of the model.

As opposed to the conventional token embedding in the BERT model, the ALBERT model employs a factorized embedding parameterization technique to reduce the number of parameters by decomposing the embedding matrix into two smaller matrices, reducing the embedding dimension to E and then expanding it to the hidden layer size H (Acharya et al. 2019). ALBERT uses two embedding matrices E^1_{ALBERT} and E^2_{ALBERT}, of sizes $|V \times E|$ and $|E \times H|$, respectively, when processing $|V|$ tokens. Thus, the total number of parameters is reduced significantly to $|V \times E| + |E \times H|$, thereby improving the memory efficiency of the model without sacrificing performance.

Cross-layer parameter sharing Cross-layer parameter sharing improves the parameter efficiency by sharing the same parameters across multiple Transformer encoder layers, thereby reducing the overall model size while simultaneously optimizing the computational resources and memory usage (Takase and Kiyono 2021). This parameter sharing can be considered in two parts: that in the attention mechanism and that in the feed-forward network. Specifically, parameter sharing in the attention mechanism means that the attention modules in each layer use the same weights and computational methods. Similarly, parameter sharing in the feed-forward network refers to the use of identical parameters across the sides of the network in all Transformer layers.

In the BERT architecture, the attention and feed-forward parameters in each of the L Transformer layers of the Transformer encoder stack are defined independently for each layer l:

$$\text{Attention parameters} : \{\theta^1_{\text{att}}, \theta^2_{\text{att}}, \ldots, \theta^L_{\text{att}}\} \tag{1}$$

$$\text{Feed forward parameters} : \{\theta^1_{\text{ffn}}, \theta^2_{\text{ffn}}, \ldots, \theta^L_{\text{ffn}}\} \tag{2}$$

In contrast, ALBERT reduces the number of parameters by employing global parameter sharing, in which the parameters across all layers are shared.

$$\text{Attention parameters sharing} : \theta_{\text{att}} = \theta^1_{\text{att}} = \theta^2_{\text{att}} = \cdots = \theta^L_{\text{att}} \tag{3}$$

$$\text{Feed forward parameters sharing} : \theta_{\text{ffn}} = \theta^1_{\text{ffn}} = \theta^2_{\text{ffn}} = \cdots = \theta^L_{\text{ffn}} \tag{4}$$

As a result, the parameters for both the attention mechanism and feed-forward network are shared across layers, significantly reducing the total number of parameters. In standard BERT, the total number of parameters is $L \times (\text{size of } \theta_{\text{att}} + \text{size of } \theta_{\text{ffn}})$, whereas in ALBERT, the total number of parameters is reduced to size of θ_{att} + size of θ_{ffn}, marking an exponential reduction in the model size.

In this study, the ALBERT model consists of three task-specific layers of Transformer encoder stacks for multi-task learning, each of which is responsible for predicting the length of imprisonment, probation status, and fines. Each task-specific layer outputs the vectors containing the information that is required to predict the corresponding legal outcome combined with the Bi-LSTM with GQA outputs, thereby ensuring deep contextual legal domain-specific understanding.

3.3.2 GQA mechanism in Bi-LSTM model

We propose a model that combines Bi-LSTM and the GQA mechanism, in which the input sequence has been previously processed by legal domain-specific embedding with FastText. The legal domain-specific embedding captures important domain-specific information from the input and is specifically tailored for legal texts.

Bi-LSTM is a bidirectional recurrent neural network (RNN) that can process the given sequences in both the forward and backward directions, capturing richer contextual information by considering both the preceding and succeeding information in a sequence (Schuster and Paliwal 1997). This structure is particularly useful for handling long texts such as legal documents, as it helps to extract richer contextual information.

Given an input sequence from the legal domain-specific embedding, the Bi-LSTM model produces the forward and backward hidden states, which are subsequently concatenated to form the output vector at each time step.

The GQA mechanism is a variant of the attention mechanism that is designed to improve the computational efficiency of large-scale models (Ainslie et al. 2023). It is similar to the standard multi-head but introduces a key modification: instead of using the entire query vector for each attention head, GQA divides the query into multiple groups and performs attention calculations within each group independently. This method reduces the computational complexity while maintaining the benefits of multi-head attention.

The standard multi-head attention mechanism transforms the input vector X (in this study, the output of the Bi-LSTM) into the query $\mathrm{Q} = \mathrm{XW}_Q$, key $\mathrm{K} = \mathrm{XW}_K$, and value $\mathrm{V} = \mathrm{XW}_V$ matrices, performing multiplication with the weight matrices. Each head independently computes an attention score, as follows (Vaswani 2017):

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

where d_k is the dimensionality of the key vector. Subsequently, once the attention scores have been computed for each head, the outputs from each head are concatenated for h heads, as follows:

$$\mathrm{O}_{\mathrm{concat}} = \mathrm{concat}\left(\mathrm{Attention}_1, \mathrm{Attention}_2, \ldots, \mathrm{Attention}_h\right)$$

Thereafter, the concatenated output is passed through a linear layer (a learnable weight matrix W^O) to produce the final output, $\mathrm{O}_{\mathrm{final}} = \mathrm{O}_{\mathrm{concat}}\mathrm{W}^O$. This step combines the results from all heads into a single output of the same dimensionality as the input.

In GQA, the query vector Q is divided into G groups, each of which performs attention independently. The attention for each group g is calculated as follows:

$$\mathrm{Q}_g = \mathrm{XW}_{Q_g}, \quad g = 1, 2, \ldots, G$$

where W_{Q_g} is the weight matrix for group g. The attention for each group is computed as

$$\text{Grouped Attention}_g(\mathrm{Q}_g, \mathrm{K}, \mathrm{V}) = \text{softmax}\left(\frac{\mathrm{Q}g\mathrm{K}^T}{\sqrt{dk_g}}\right)\mathrm{V}$$

where d_{k_g} is the dimensionality of the query for group g. Subsequently, the outputs from all groups are concatenated to form the final output:

$$\mathrm{O} = \text{concat}\left(\text{Grouped Attention}_1, \text{Grouped Attention}_2, \ldots, \text{Grouped Attention}_G\right)$$

The subsequent process is similar to that of the standard attention mechanism.

In this study, the outputs of Bi-LSTM with the GQA mechanism in each task-specific layer are combined with the corresponding outputs of the multi-task heads in the ALBERT model to predict the length of imprisonment, probation status, and fines.

3.3.3 Legal outcome prediction with multi-task learning

Multi-task learning focuses on simultaneously predicting multiple output types from the same dataset (Chen et al. 2024). This approach improves the data utilization efficiency and improves the generalization performance of the model by leveraging information that is shared across tasks. In this study, we aim to predict multiple outputs, namely the length of imprisonment, probation status, and fines, concurrently by applying multi-task learning.

In the proposed architecture, the internal embeddings in the ALBERT model and FastText embedding in the legal domain-specific embedding module are trained within the shared representations. The model is divided into task-specific layers from the Transformer encoder stack in the ALBERT model and Bi-LSTM with GQA mechanism, each of which is dedicated to predicting one of the three legal outcomes.

While the embedding stage of the ALBERT model is not split into task-specific layers, the Transformer encoder stack is followed by independent task-specific layers, which extract the feature vectors that are required for predicting the length of imprisonment, probation status, and fines. Similarly, the Bi-LSTM with GQA mechanism is structured into three task-specific layers, where the input from the FastText embedding is processed to extract feature vectors that are tailored to each legal outcome.

Thereafter, the outputs from the task-specific layers of both the ALBERT model and Bi-LSTM with GQA mechanism are concatenated to form a final feature vector, which is subsequently processed through fully connected layers that act as multi-task heads to generate the final predictions for each legal outcome. This multi-task learning structure effectively enables the simultaneous prediction of multiple value types in legal sentencing, and can be extended to accommodate additional tasks by incorporating further task-specific layers.

4 Experiments

4.1 Experimental settings

The dataset, baseline models, and performance evaluation metrics used in the experiments are described in this section.

4.1.1 Dataset

Actual criminal traffic accident case rulings were collected from the legal document repository LBox (LBox 2023) to train and validate the proposed model. We gathered approximately 37,406 court rulings from various metropolitan regions, such as Seoul, Uijeongbu, Ulsan, and Changwon in South Korea, covering the period from November 2020 to September 2023. Detailed information regarding the courts and number of cases collected is presented in Table 2. The dataset was divided into training, validation, and test sets at a ratio of 8:1:1.

We analyzed both the ruling lengths and distributions of the three target sentencing outcomes to characterize the dataset of criminal traffic accident rulings prior to model training. Figure 2 depicts the probability density functions (PDFs) of the ruling word counts across district courts, highlighting inter-court differences in the mean and variance. Figure 3 presents vertically stacked subplots of the imprisonment duration (PDF with an overall mean of 12.8 months indicated by a dashed line), fine amount (PDF in $10 units with a mean of 676.5 units), and probability of probation, enabling a direct comparison of distributional shapes and central tendencies. Figure 3d compares the ruling length distributions by probation status, with light green indicating no probation, dark green indicating probation, and dashed lines marking the group means, suggesting a potential relationship between document length and probation outcome.

Table 2 District courts where judgments were collected

No.	District court name	Number of cases[1]
1	Seoul Southern District Court	2,964
2	Seoul Eastern District Court	2,015
3	Seoul Northern District Court	2,099
4	Seoul Western District Court	1,548
5	Seoul Central District Court	2,912
6	Ulsan District Court	2,040
7	Uijeongbu District Court	12,461
8	Uijeongbu District Court Goyang Branch	2,247
9	Uijeongbu District Court Namyangju Branch	330
10	Changwon District Court	4,782
11	Changwon District Court Geochang Branch	248
12	Changwon District Court Masan Branch	942
13	Changwon District Court Miryang Branch	542
14	Changwon District Court Jinju Branch	1,242
15	Changwon District Court Tongyeong Branch	1,034

[1]Only for criminal traffic accident cases

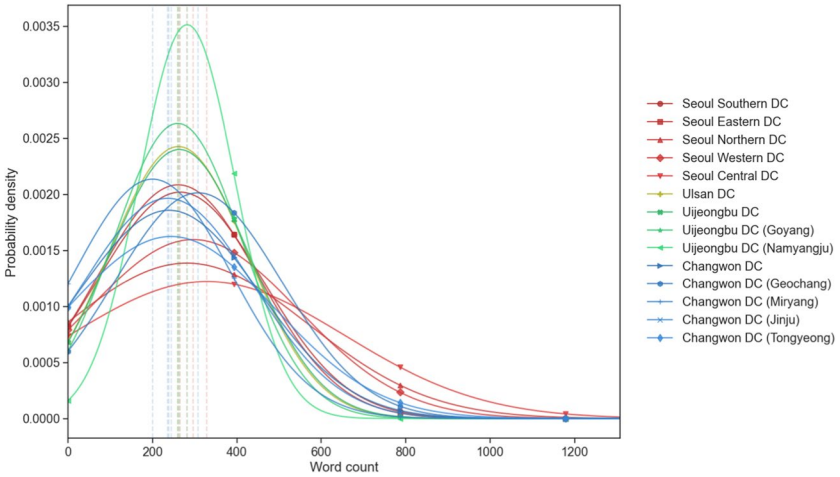

Fig. 2 PDFs of sentence word counts across district courts

4.1.2 Baseline models

A total of 12 baseline models were employed for the comparative experiments: a naïve classifier/regressor (Katz et al. 2017) that predicts the most frequent class or mean value as a statistical baseline, LSTM (Hochreiter 1997), Bi-LSTM (Schuster and Paliwal 1997), GRU (Cho 2014), Bi-GRU, their attention mechanism versions, KoBERT (Devlin et al. 2018), T5 (Raffel et al. 2020), Llama 3.1 (Meta AI 2024), and ALBERT (Lan 2019). These models are specialized in sequence processing and/or based on Transformer architectures. Table 3 presents the baseline models used and their descriptions. Key hyperparameters, selected based on preliminary validation performance rather than through separate optimization, are summarized in Table 4.

4.1.3 Evaluation metrics

The three target variables in this study were the length of imprisonment, probation status, and amount of fines. The prediction of the imprisonment length and fines was treated as a regression problem; thus, the root mean square error (RMSE) and mean absolute percentage error (MAPE) were employed as the evaluation metrics. Conversely, the probation status indicates whether or not probation is granted, framing this task as a binary classification problem, for which the accuracy, precision, recall, and F1-score were used as evaluation metrics. The mathematical calculations for each metric are provided in the following equations.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5}$$

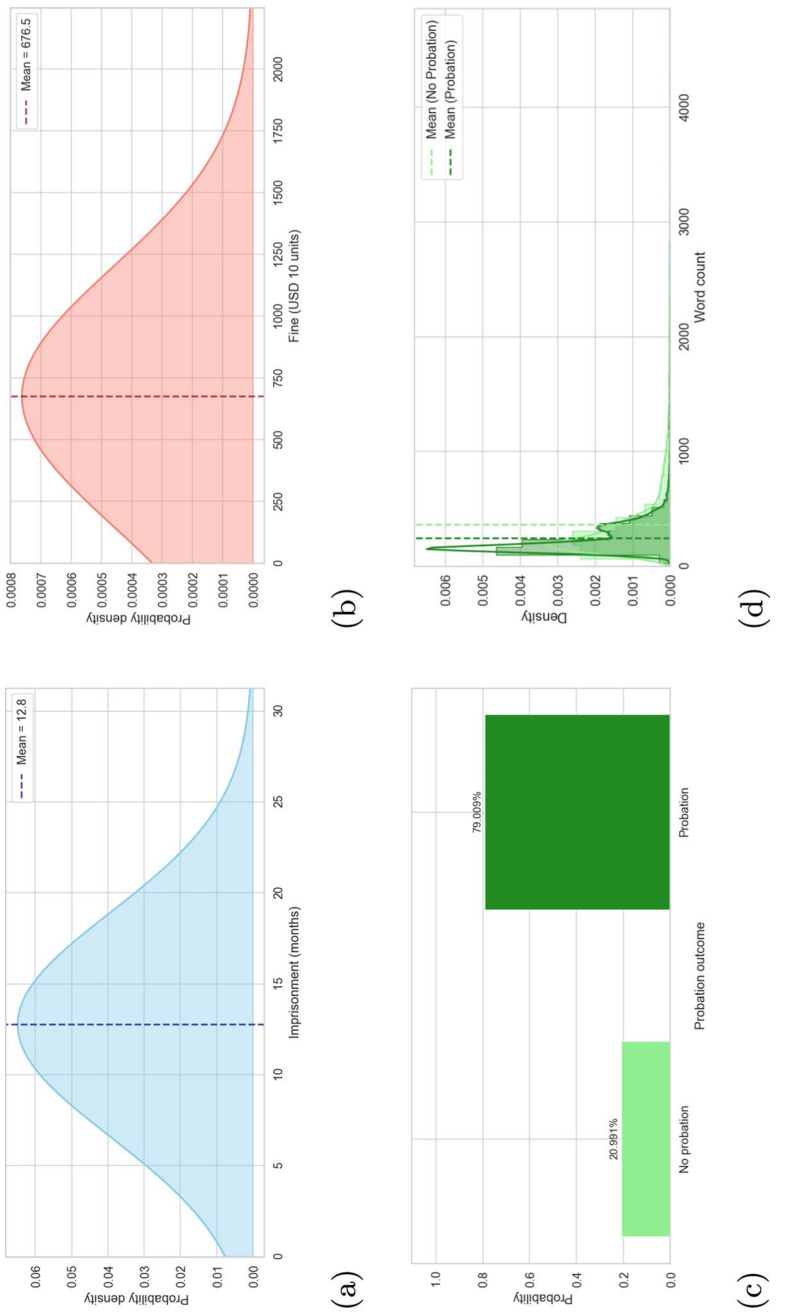

Fig. 3 Distributional characteristics of key sentencing variables: (a) length of imprisonment, (b) fine amounts, (c) probation outcomes, and (d) word counts by probation status

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{6}$$

where y_i denotes the ground truth values, $\hat{y}_i$ are the values estimated by the model, and n is the total number of test data points.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

4.1.4 Computational environment

We conducted all experiments on a single workstation equipped with an NVIDIA RTX 4070 GPU (12GB VRAM), using PyTorch 2.0.1 with Python 3.8 on Ubuntu 22.04, to ensure reproducibility and assess practical feasibility. This moderate hardware setting demonstrates that the proposed TRACS-LLM model can be trained and evaluated without requiring high-end infrastructure, suggesting its scalability and potential for practical deployment in legal domains.

4.2 Ablation study

We conducted experiments to analyze and compare the performance of the proposed model.

The first experiment compared the performance of traditional RNN models, such as LSTM, Bi-LSTM, and GRU, along with their variants incorporating attention mechanisms, as baseline models against the proposed model. The second experiment evaluated the impact of replacing the Bi-LSTM with GQA in parallel with ALBERT to examine how different model combinations affect the performance.

As shown in the performance comparison of the baseline models under "Ablation study 1" in Table 5, the AT-Bi-LSTM model achieved an RMSE of approximately 4.775 and 126.651 for the imprisonment and fine predictions, respectively, outperforming both the LSTM and GRU models. Moreover, in the probation prediction task, the Llama 3.1 model achieved an F1-score of 0.957, surpassing the basic RNN with attention mechanism models. Overall, the models with attention mechanisms outperformed the basic models. These results indicate that the use of

Table 3 Description of the models used in the comparative experiments

Model	Description
LSTM	Traditional LSTM network
AT-LSTM	LSTM with attention
Bi-LSTM	Bidirectional LSTM for forward and backward dependencies
AT-Bi-LSTM	Bi-LSTM with attention
GRU	Gated recurrent unit, simpler than LSTM
AT-GRU	GRU with attention
Bi-GRU	Bidirectional GRU for dual-directional context
AT-Bi-GRU	Bi-GRU with attention
Ko-BERT (Devlin et al. 2018)	Korean-adapted BERT model for language tasks
T5 (Raffel et al. 2020)	Transformer for text-to-text tasks
Llama3.1 (Meta AI 2024)	A version of the Llama model
ALBERT (Lan 2019)	A lightweight version of BERT (A Lite BERT)
Proposed method	Models combining ALBERT and GQA Bi-LSTM

Table 4 Key hyperparameters used for the baseline (shared) and proposed models

Category	Hyperparameter	Value / description
Training (shared)	Batch size	2
	Number of epochs	300
	Learning rate	0.001 (Adam optimizer)
Model Architecture (proposed only)	Embedding dimension	200 (FastText pre-trained vectors)
	Bi-LSTM hidden size	128 (bidirectional)
	Number of GQA groups	127
	Token length limit	512 (ALBERT input truncation)

attention mechanisms improved the performance. However, the best-performing baseline model still lagged behind the ALBERT-based models in terms of the overall performance.

The results of the second experiment, in which we evaluated different parallel combinations with ALBERT, are listed under "Ablation study 2" in Table 5. It can be observed that the ALBERT+Bi-LSTM with GQA model achieved the best performance. This model exhibited the lowest MAPE, at approximately 22.24, for imprisonment prediction, and the lowest RMSE, at approximately 142.197, for fine prediction, significantly outperforming the other combinations. In addition, in the probation prediction task, it achieved an accuracy and F1-score of approximately 0.965 and 0.978, respectively, highlighting the effectiveness of the parallel structure that combines ALBERT and Bi-LSTM with GQA. Although the ALBERT-only model exhibited the weakest performance across all tasks, combining ALBERT with Bi-LSTM or attention mechanisms improved its performance substantially. Notably, the model incorporating

Table 5 Results of performance evaluation with statistical significance

Model	Imprisonment		Fines		Probation			
	RMSE[1]	MAPE	RMSE[1]	MAPE	Accuracy[2]	Precision	Recall	F1-score
Ablation study 1: Baseline models								
Naïve regressor	6.182	5.37×10^{258}	258.099	3.00×10^{17}		-	-	-
Naïve classifier	-	-	-	-	0.783	0.783	1.000	0.878
LSTM	4.940*	28.167	134.625**	158.305	0.903**	0.919	0.961	0.940
AT-LSTM	4.993**	28.780	127.438**	125.084	0.914**	0.930	0.963	0.946
Bi-LSTM	4.930**	28.350	126.915**	168.941	0.918**	0.927	0.973	0.949
AT-Bi-LSTM	4.775**	26.812	126.651**	166.059	0.923**	0.944	0.959	0.951
GRU	4.846**	26.643	126.507**	180.205	0.920**	0.934	0.973	0.950
AT-GRU	4.878**	27.742	137.321**	154.912	0.913**	0.937	0.959	0.945
Bi-GRU	4.812**	28.205	126.799**	150.867	0.923**	0.937	0.966	0.951
AT-Bi-GRU	4.929**	32.576	123.943**	139.603	0.921**	0.927	0.976	0.951
KoBERT	5.378**	25.648	253.992**	116.356	0.886**	0.892	0.972	0.930
T5	5.387**	25.312	254.059**	118.949	0.885**	0.888	0.975	0.930
Llama 3.1	4.953**	27.399	193.590**	297.812	0.931**	0.940	0.975	0.957
Ablation study 2: ALBERT + other models in parallel								
ALBERT only	6.188	26.816	258.127	218.669	0.783	0.783	**1.000**	0.878
LSTM	4.012	24.280	149.137	238.152	0.962	0.969	0.983	0.976
Bi-LSTM	3.935	23.322	142.451	209.594	0.962	0.973	0.980	0.976
AT-LSTM	3.979	23.733	148.933	231.207	0.959	0.969	0.979	0.974
AT-Bi-LSTM	**3.916**	22.947	142.376	212.014	0.960	0.969	0.979	0.974
Proposed	3.917	**22.240**	**142.110**	**207.792**	**0.965**	**0.978**	0.983	**0.977**

[1] Paired Student's t-test; [2] Bootstrap test (percentile-based) $^{*}p < 0.05$, $^{**}p < 0.01$

Bi-LSTM with GQA outperformed all other models across all metrics, demonstrating that the introduction of GQA is the key factor in improving the predictive performance.

4.3 Discussion

The proposed legal sentencing prediction method for traffic-related criminal cases that combines ALBERT and Bi-LSTM with GQA in a parallel structure with a multi-task learning approach includes LLM fine-tuning as well as additional model structuring for the deep understanding of legal language, context, and specific details of traffic accidents. This enables the module to reflect the legal context and regulations more precisely. The model is designed to predict key legal outcomes such as the length of imprisonment, probation status, and amount of fines.

In our experiments, we validated the feasibility and applicability of the proposed method through two ablation experiments. The first experiment compared the performance of the proposed model against baseline models including LSTM, Bi-LSTM, and GRU, as well as their attention-based variants. In the second experiment, Bi-LSTM with GQA was replaced with various models to evaluate the impact of its parallel combination with ALBERT on the performance. The results demonstrated that the proposed ALBERT+Bi-LSTM with GQA model achieved superior performance across all tasks, exhibiting the lowest RMSE for imprisonment length and fine predictions and the highest F1-score for probation prediction.

We applied paired Student's t-tests for the RMSE-based regression metrics (imprisonment and fines) and bootstrap-based percentile tests for the classification accuracy (probation prediction) to assess the statistical significance of the observed performance differences. The resulting p-values, detailed in Table 5, demonstrate that the proposed TRACS-LLM model consistently outperformed the baseline models. In particular, the majority of comparisons yielded $p < 0.01$, indicating that the performance improvements were not attributable to random variation but were statistically significant across both the regression and classification tasks.

However, these experiments relied primarily on legal rulings written in Korean and trained within the framework of Korean law, which may limit the generalizability of the findings to other languages or legal systems. As shown in Table 1, numerous studies have explored the use of LLMs in the legal domain, with most methods tailored to the legal systems and languages of their respective countries. Furthermore, a large proportion of research has focused primarily on tasks such as summarizing legal documents, predicting only case outcomes, or providing legal assistance through AI-based chatbots. This study focused specifically on criminal traffic accident cases in South Korea, with the prediction of legal sentencing outcomes such as the imprisonment length, probation status, and fines, rather than summaries of legal rulings or decisions. To the best of our knowledge, this research represents the first attempt to report the use of LLMs in predicting legal sentencing outcomes within the Korean legal domain.

In addition, the findings of this study offer practical applications for legal professionals, including attorneys, judges, and parties who are involved in criminal traffic accidents. As shown in Table 6, the proposed model aligns closely with actual legal rulings based on the extracted context of traffic incidents, yielding predictions that mirror real sentencing outcomes. In particular, the model effectively captures the complexity of legal decisions in cases such as repeated DUI offenses or incidents with fatal consequences, providing reliable predictions. This demonstrates the potential of the proposed methodology to serve as a supportive tool in legal decision-making by helping to reduce the cognitive burden on legal professionals while improving consistency and fairness in sentencing judgments.

5 Conclusion

This study has presented a criminal traffic accident sentencing prediction method known as TRACS-LLM, which combines ALBERT and Bi-LSTM with GQA in a parallel structure to predict legal outcomes such as the imprisonment length, probation status, and fines. The model leverages shared information across these tasks by employing a multi-task learning approach to enhance the prediction accuracy. The integration of the general language modeling capabilities of ALBERT with the domain-specific expertise of Bi-LSTM with GQA enables the model to capture both the general linguistic context and legal-specific information effectively.

Comprehensive experiments demonstrated the effectiveness of the proposed method, which outperformed traditional models and baseline approaches. Although the performance gains were not always large in absolute terms, statistical tests

Table 6 Examples of actual and predicted legal sentencing

No.	Situations*					
	Actual sentencing			Predicted sentencing		
	Imprison[1]	Probation[2]	Fine[3]	Imprison[1]	Probation[2]	Fine[3]
1	After leaving a restaurant in front of a villa while intoxicated, the defendant was caught driving under the influence. The defendant has a previous record of similar offenses.					
	12.00	1	0.0	11.04	1	2.00
2	The defendant has been punished for DUI offenses in the past. In this case, the defendant was again caught driving a passenger car under the influence near Seongnam City, which was confirmed through a breathalyzer test and police reports. The defendant supports their family and is facing economic difficulties, and their past offenses have not exceeded a fine.					
	0.00	0	700.0	1.68	0	797.19
3	The defendant, while driving a car under the influence in Changnyeong County, collided with a bicyclist, causing the victim's death. The defendant then fled the scene. The defendant has a prior record of DUI offenses and caused serious consequences in this case.					
	36.00	0	0.0	34.98	0	35.27
4	The defendant was caught driving under the influence in Namyangju City, with a blood alcohol level exceeding the legal limit. The defendant had been previously punished for a DUI and had received a summary order.					
	12.00	1	0.0	12.28	1	7.52
5	The defendant was caught driving a Bongo truck under the influence in front of an apartment building. The defendant has a significant number of previous DUI offenses.					
	12.00	1	0.0	13.09	1	0.23

* Examples were randomly selected from the test set [1] Length of Imprisonment (months) [2] Probation status: 1: probation, 0: no probation [3] Unit: USD 10 (e.g., 7 represents USD 70)

confirmed that these improvements were not attributable to random variation. In addition, the application of the GQA mechanism contributed to the computational efficiency, allowing the model to focus on critical legal details in each case. These results suggest the potential for supporting legal professionals by providing data-driven insights that reduce the cognitive load and enhance the consistency and fairness of sentencing decisions.

Studies that directly address sentencing prediction are relatively limited, both within the Korean legal system and across LLM-based legal research. Many existing studies focus on outcome prediction, case summarization, or legal assistance tasks, with sentencing prediction remaining a relatively underexplored area. This study helps to fill this gap and offers a meaningful contribution to AI and law by addressing this overlooked task.

Although this study focuses on traffic accident-related cases within the Korean legal framework, its methodology has broader implications. The system can be adapted to other legal domains and jurisdictions, creating opportunities for the application of LLMs in legal decision-making processes. The adaptability of the model to different languages and legal frameworks can be explored in future research, further enhancing its utility across diverse legal contexts.

Acknowledgements This work was supported by Soonchunhyang University Research Fund.

Author Contributions Conceptualization: Hyunsik Min, Byeongjoon Noh; Methodology: Hyunsik Min; Writing– original draft preparation: Hyunsik Min; Writing– review and editing: Byeongjoon Noh; Funding acquisition: Byeongjoon Noh; Resources: Byeongjoon Noh; Supervision: Byeongjoon Noh

Declarations

Conflicts of Interest The authors declare no conflicts of interests.

References

Acharya A, Goel R, Metallinou A, Dhillon I (2019) Online embedding compression for text classification using low rank matrix factorization. In: Proceedings of the Aaai Conference on Artificial Intelligence, vol 33, pp 6196–6203

Ainslie J, Lee-Thorp J, Jong M, Zemlyanskiy Y, Lebrón F, Sanghai S (2023) Gqa: Training generalized multi-query transformer models from multi-head checkpoints. arXiv:2305.13245

Benedetto I, Koudounas A, Vaiani L, Pastor E, Cagliero L, Tarasconi F, Baralis E (2024) Boosting court judgment prediction and explanation using legal entities. Artif Intell Law, 1–36

Benedetto I, La Quatra M, Cagliero L (2025) Legitbart: a summarization model for italian legal documents. Artif Intell Law, 1–31

Cao L, Wang Z, Xiao C, Sun J (2024) Pilot: Legal case outcome prediction with case law. arXiv:2401.15770

Chen S, Zhang Y, Yang Q (2024) Multi-task learning in natural language processing: An overview. ACM Comput Surv 56(12):1–32

Cho K (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv:1406.1078

Cui J, Li Z, Yan Y, Chen B, Yuan L (2023) Chatlaw: Open-source legal large language model with integrated external knowledge bases. arXiv:2306.16092

Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805

Goodson N, Lu R (2023) Intention and context elicitation with large language models in the legal aid intake process. arXiv:2311.13281

Greco CM, Tagarelli A (2024) Bringing order into the realm of transformer-based language models for artificial intelligence and law. Artif Intell Law 32(4):863–1010

Hochreiter S (1997) Long short-term memory. Neural Comput MIT-Press

Jin Z, Noh B (2023) From prediction to prevention: Leveraging deep learning in traffic accident prediction systems. Electronics 12(20):4335

Joulin A, Grave E, Bojanowski P, Mikolov T (2016) Bag of tricks for efficient text classification. arXiv:1607.01759

Katz DM, Bommarito MJ II, Blackman J (2017) A general approach for predicting the behavior of the supreme court of the united states. PloS one 12(4):0174698

Lan Z (2019) Albert: A lite bert for self-supervised learning of language representations. arXiv:1909.11942

LBox (2023) https://lbox.kr/v2

Lee J, Mannering F (2002) Impact of roadside features on the frequency and severity of run-off-roadway accidents: an empirical analysis. Accident Anal Prevent 34(2):149–161

Meta AI (2024) Introducing Llama 3.1: Our Most Capable Models to Date. Accessed: 2024-07-27. https://ai.meta.com/blog/meta-llama-3-1/

Min B, Ross H, Sulem E, Veyseh APB, Nguyen TH, Sainz O, Agirre E, Heintz I, Roth D (2023) Recent advances in natural language processing via large pre-trained language models: A survey. ACM Comput Surv 56(2):1–40

Naveed H, Khan AU, Qiu S, Saqib M, Anwar S, Usman M, Akhtar N, Barnes N, Mian A (2023) A comprehensive overview of large language models. arXiv:2307.06435
Noh B, Ka D, Lee D, Yeo H (2021) Analysis of vehicle-pedestrian interactive behaviors near unsignalized crosswalk. Transport Res Record 2675(8):494–505
Noh B, Park H, Yeo H (2022) Analyzing vehicle-pedestrian interactions: Combining data cube structure and predictive collision risk estimation model. Accident Anal Prevent 165:106539
Noh B, Yeo H (2021) Safetycube: Framework for potential pedestrian risk analysis using multi-dimensional olap. Accident Anal Prevent 155:106104
Noh B, Yeo H (2022) A novel method of predictive collision risk area estimation for proactive pedestrian accident prevention system in urban surveillance infrastructure. Transport Res Part C: Emerg Technol 137:103570
OpenAI (2024) Hello GPT-4o. Accessed: 2024-07-27. https://openai.com/index/hello-gpt-4o/
Perrels A, Votsis A, Nurmi V, Pilli-Sihvola K (2015) Weather conditions, weather information and car crashes. ISPRS Int J Geo-Inf 4(4):2681–2703
Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2020) Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res 21(140), 1–67
Rodríguez Rodríguez CR, Amoroso Fernández Y, Zuev DS, Peña Abreu M, Zulueta Veliz Y (2024) M-lamac: a model for linguistic assessment of mitigating and aggravating circumstances of criminal responsibility using computing with words. Artif Intell Law 32(3):697–739
Rolison JJ, Regev S, Moutari S, Feeney A (2018) What are the factors that contribute to road accidents? an assessment of law enforcement views, ordinary drivers' opinions, and road accident records. Accident Anal Prevent 115:11–24
Satterfield N, Holbrooka P, Wilcoxa T (2025) Fine-tuning llama with case law data to improve legal domain performance
Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45(11):2673–2681
Sennrich R (2015) Neural machine translation of rare words with subword units. arXiv:1508.07909
Shui R, Cao Y, Wang X, Chua T-S (2023) A comprehensive evaluation of large language models on legal judgment prediction. arXiv:2310.11761
Shu D, Zhao H, Liu X, Demeter D, Du M, Zhang Y (2024) Lawllm: Law large language model for the us legal system. arXiv:2407.21065
Song X, Salcianu A, Song Y, Dopson D, Zhou D (2020) Fast wordpiece tokenization. arXiv:2012.15524
Sun Z (2023) A short survey of viewing large language models in legal aspect. arXiv:2303.09136
Takase S, Kiyono S (2021) Lessons on parameter sharing across layers in transformers. arXiv:2104.06022
Vaswani A (2017) Attention is all you need. Adv Neural Inf Process Syst
Wei B, Yu Y, Gan L, Wu F (2025) An llms-based neuro-symbolic legal judgment prediction framework for civil cases. Artif Intell Law, 1–35
Yue S, Chen W, Wang S, Li B, Shen C, Liu S, Zhou Y, Xiao Y, Yun S, Huang X, et al (2023) Disc-lawllm: Fine-tuning large language models for intelligent legal services. arXiv:2309.11325
Zhou Y, Huang H, Wu Z (2023) Boosting legal case retrieval by query content selection with large language models. In: Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, pp 176–184

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Hyunsik Min[1] · **Byeongjoon Noh**[2]

✉ Byeongjoon Noh
powernoh@sch.ac.kr

[1] Department of Future Convergence Technology, Soonchunhyang University 22 Soonchunhyang-ro, Asan 31538, South Korea

[2] Department of AI and Big data, Soonchunhyang University, Asan 31538, South Korea