LAHEL: An AI-Generated Content Approached LAwHELper to Personal Legal Advice

Yisu Wang[1,3,†], Nanxi Hou[2,†], Kaiyuan Xu[1,3], Zepu Ni[4] and Guofeng Wu[5]

[1]Xidian University, 2 South Taibai Road, Xi'an, Shaanxi, 710071, People's Republic of China
[2]Georgetown University, 37th and O Streets NW, Washington, D.C., 20057, United States
[3]DeepBrainCongress Intelligent Technology Co., Ltd. XianYang, Shaanxi, 712023, People's Republic of China.
[4]Central University of Finance and Economics, 39 South College Road, Haidian District, Beijing,100081, People's Republic of China.
[5]Beijing university of posts and telecommunications, 10 West TuCheng Road, Haidian District, Beijing,100081, People's Republic of China.

Abstract

In certain developing countries, public awareness of legal rights is increasing, leading to a growing demand for legal consultation. However, the time and monetary costs associated with consulting professional lawyers remain high.

Concurrently, there are two major impacts of computer science on the current legal sector. First, within government and public prosecution systems, information systems have accumulated vast amounts of structured and semi-structured data, offering significant economic value and potential for exploration. However, few people have attempted to mine these data resources. Second, intelligent dialogue systems have matured, but dialogue systems specifically tailored for the legal domain have not yet emerged.

Considering these two trends, we introduce LAHEL, a legal consultation system developed by a team of nine individuals over the course of two years, dedicated to addressing the aforementioned issues. The system comprises three components: search, human dialogue systems, and robot dialogue systems. Its primary contributions are twofold: exploring the application of AI in legal consultation and summarizing lessons learned from the design of legal consultation systems.

Keywords

Legal Consultation, Artificial Intelligence, Dialogue Systems

1. Introduction

The advent of the technological revolution has led to the rapid development of artificial intelligence (AI) [1], which has gradually empowered various sectors of human society, including education, entertainment, and healthcare. In 2023, the release of the chatbot, ChatGPT[2], brought public attention back to AI, as it showcased the transition of AI from being a mere auxiliary tool to demonstrating independence and creativity. ChatGPT, though primarily a human-machine interaction robot, has widespread applications, including translation, text analysis, chart analysis, songwriting, email composition, and even coding and debugging computer programs.

The core of AI lies in its algorithms. In the legal industry, AI's current applications mainly involve rapid case information retrieval and document management, which greatly reduce the workload of lawyers and allow them to focus on core case issues, improving service quality. Considering the increasing public awareness of legal rights and the growing demand for legal consultations, AI can potentially serve as a basic legal service provider.

Although AI has its limitations, it can provide general legal guidance and address common legal queries. For individuals lacking legal knowledge, AI-based legal consultation offers a convenient avenue for seeking legal advice. For those hesitant to seek legal counsel due to cost concerns, free AI-powered legal services are an efficient and trustworthy option. Thus, exploring and developing AI applications in legal consultation holds significant practical value. In this paper, we will discuss the potential benefits and challenges of integrating AI into the legal sector, as well as its implications for both legal professionals and the general public.

The motivation behind developing LAHEL is to address the difficulties that individuals and small businesses face when trying to access legal resources. The legal industry has a reputation for being complex, slow-moving, and expensive, which can be a significant barrier for many people seeking legal assistance[3]. Additionally, legal jargon and complex procedures can make it difficult for individuals to navigate the legal system without professional help [4].

LAHEL aims to bridge the gap between legal resources and those who need them by providing a user-friendly platform that simplifies legal jargon and procedures. It utilizes cutting-edge natural language processing and machine learning technologies to provide personalized legal advice to users, which can help them make informed decisions about legal matters.

By making legal resources more accessible, LAHEL

asueeer@163.com (Y. Wang); nh616@georgetown.edu (N. Hou); xukaiyuan0107@gmail.com (K. Xu); zepuni@outlook.com (Z. Ni); guofengwu@bupt.cn (G. Wu)

CEUR Workshop Proceedings (CEUR-WS.org)

hopes to empower individuals and small businesses, allowing them to address legal issues in a more timely and cost-effective manner. Ultimately, our goal is to democratize access to legal resources and help level the playing field for those who might otherwise be at a disadvantage.

The paper's structure consists of a methodology section describing the design and implementation of the conversational system, a discussion of the implications and limitations of the system, and a conclusion summarizing the paper's contributions and recommendations for future work.

Our system is accessible at [5][6] for users to explore and interact with.

2. Challenges and Approach

LAHEL is a platform designed to provide legal advice and assistance to non-legal professionals. The main challenges we faced during the development process were: 1) designing a conversation system that could support both human and AI-based interactions; 2) ensuring that the AI advisor provided professional legal advice; 3) addressing specific requirements for special government-related systems, such as handling particular user inquiries; and 4) managing interactions between different system modules.

One of the most significant challenges we faced during the development process was managing the interaction between different system modules. Despite our developers' extensive experience in computer systems and AI, defining the API interactions between modules proved to be time-consuming. During product debugging, these module interactions were often the source of bugs that required significant time and effort to locate and fix. Throughout the development process, we accumulated valuable experience in reducing bug-fixing time and increasing development speed. These lessons will be discussed in detail in Chapter 5.

3. AI-Powered Legal Consultation System Design

3.1. Overview

This section of the paper elaborates on the design and implementation of a human-assisted dialogue system as part of LAHEL, aiming to provide personalized legal advice and support to users. We outline the architecture, interaction flow, and API design to facilitate a seamless connection between clients and legal customer service representatives (CSRs).

3.2. System Architecture

The human-assisted dialogue system architecture comprises several modules:

1. **Frontend User Interface:** Provides an interactive platform for users to request legal advice and communicate with legal CSRs.
2. **Frontend CSR Interface:** Enables legal CSRs to manage incoming conversations, view client messages, and respond to user inquiries.
3. **Backend User-side Messaging Service:** Manages user-related messages and communication between the frontend user interface and the conversation management module.
4. **Backend CSR-side Messaging Service:** Handles legal CSR-related messages and communication between the frontend CSR interface and the conversation management module.
5. **Conversation Management Module:** Facilitates the creation and management of conversations, message synchronization, and user-CSR matching.

3.3. Interaction

In our study, we have designed an AI-assisted dialogue system to facilitate seamless real-time communication between users and customer service representatives. The system's workflow, as shown in Figure 1, consists of various components such as the user-side messaging service, the AI helper, the conversation management module, the message synchronization queue consumer module, and the service-side messaging service, which work in tandem to ensure efficient and smooth interaction.

When a user requests human assistance by entering "human" in the dialogue box, a notification is sent to the legal CSR's conversation list. This method ensures that the CSRs are aware of new requests, allowing them to manage their workload and respond accordingly. Upon receiving a notification, the legal CSR can join the conversation by clicking the notification button, initiating a direct communication channel with the user. This approach provides a simple and straightforward way for users to connect with legal CSRs, ensuring prompt and efficient assistance.

The automatic CSR assignment method streamlines the process of connecting users with legal CSRs by identifying available and online CSRs, and then automatically assigning them to a conversation. This approach reduces the need for manual intervention, leading to faster response times and a more efficient use of CSR resources. The system continuously monitors CSR availability and assigns them to conversations based on their workload and user demand. This dynamic assignment process helps to maintain a balanced distribution of tasks

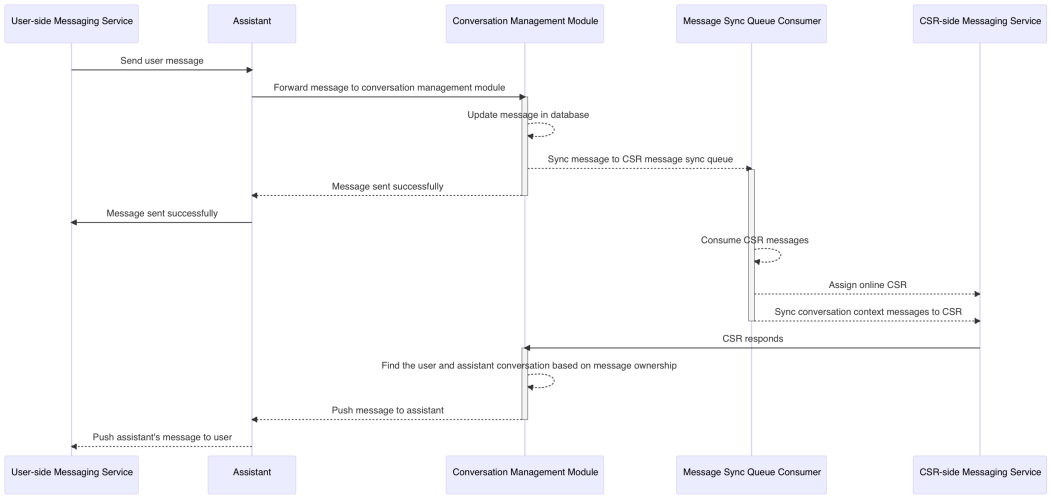

Figure 1: AI-Assisted Dialogue System Workflow. This diagram illustrates the workflow of an AI-assisted dialogue system, depicting the interactions between the user-side messaging service, the AI helper, the conversation management module, the message synchronization queue consumer module, and the service-side messaging service for seamless real-time communication between users and customer service representatives.

among legal CSRs, optimizing the overall customer service experience.

The virtual legal assistant approach introduces an intermediary between users and legal CSRs, aiming to enhance user engagement and create a more interactive experience. In this method, a virtual legal assistant relays messages between users and legal CSRs, giving users the impression that they are conversing with the assistant. The assistant forwards user messages to the legal CSR, who then responds on behalf of the assistant.

This approach offers several advantages, including the ability to handle multiple conversations simultaneously and providing a consistent user experience. The virtual legal assistant can also be augmented with AI capabilities, such as automated responses for frequently asked questions or simple tasks, reducing the workload on legal CSRs and allowing them to focus on more complex inquiries.

The API design includes several key data structures, such as CreateConversationRequest, SendMessageRequest, LoadConversationsRequest, and LoadConversationDetailRequest. These data structures are designed to facilitate the creation and management of legal consultations, sending and receiving messages, and loading conversation details.

The system is designed to provide real-time legal support and seamless interaction between users and legal CSRs. By leveraging modular architecture and a well-defined API, the proposed system offers a scalable and extensible solution for improving legal consultation services in various domains.

4. Design of a Conversational Robot System

The LAHEL AI conversation system aims to provide a one-stop legal consultation service for non-legal professionals. It is designed to accurately address users' legal concerns and assist them in handling various tasks. The main implementation strategy involves constructing a task-oriented dialogue system that combines conversation system architecture with multi-task models, such as GPT[7].

Unlike GPT, which is a large model trained on massive dialogue datasets using multiple GPUs, our approach adopts a multi-model multi-task strategy. We collect data from government and court websites, including policy documents and judicial decisions, to train our models. Initially, we considered using distributed GPU cards to train the models on government data. However, we found that due to the smaller volume of government documents and judicial decisions compared to public dialogue datasets, sequentially training multiple models on a single GPU card and then combining the results proved to be a more cost-effective and efficient solution.

As the dataset is focused on a single legal domain and not overly large, using multiple models for training yields excellent results. LAHEL is capable of accurately recognizing and responding to specific questions related

to critical tasks. Furthermore, it can engage in multi-turn conversations with users, taking into account the entire conversation to provide comprehensive answers. The combination of the task-oriented dialogue system and the multi-model multi-task approach enables LAHEL to deliver precise and effective legal advice to non-legal professionals.

5. Implementation

In this chapter, we will provide a detailed description of the module implementation of the LAHEL system. The entire system consists of 8,000 lines of code, with 4,000 lines dedicated to the frontend and user interaction (using javascript+typescript), and the other 4,000 lines dedicated to the development of the artificial legal consulting system (using Go programming language).

5.1. Artificial Legal Consulting System (ALS)

Although manual consultation is essentially an instant messaging system, developing such a system from scratch is not easy. We have seen instant messaging systems like WhatsApp and Telegram, but their server-side code is closed-source. Moreover, open-source chat systems do not support customization, making it difficult to adapt them for our automated chatbot system. We also referred to blog posts by Facebook and Instagram's technical departments about IM development, and found that their main challenges were in splitting system interfaces and designing message queues. We, therefore, conducted detailed analysis on these two issues and provided design solutions. Furthermore, large-scale instant messaging requires considering high concurrency, i.e., how to handle a large number of simultaneous users. However, our system is designed for government departments and legal consulting firms, which will not have such high concurrency requirements. Therefore, high concurrency issues were not considered in our design. In addition to contributing a chat system for the legal field, our messaging system also provides an additional reference for the open-source community.

When designing a chat system, there are three design patterns to consider: push mode vs pull mode vs push-pull combined mode.

- Push mode: When there is a new message, the server actively pushes it to all clients (iOS, Android, PC, etc.).
- Pull mode: The frontend initiates a request to pull messages. To ensure message timeliness, push mode is generally used, while pull mode is usually used for retrieving historical messages.

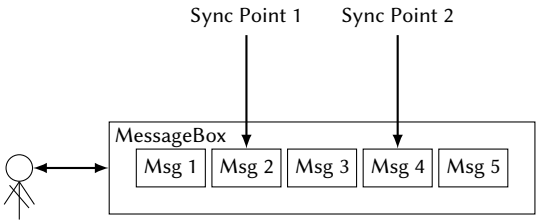

Figure 2: MessageBox with Interaction and Synchronization Points

- Push-pull combined mode: When there is a new message, the server first pushes a new message notification to the frontend, and the frontend then pulls the message from the server.

Under normal circumstances, user messages will be pushed to all receiving clients after being stored and processed by the server. However, push messages can be lost, and the most common situation is when users appear to be online (i.e., the push service is based on a long connection, but the connection may have been disconnected, meaning the user has gone offline. The server will mistakenly assume the user is still online). Therefore, using push mode alone can result in lost messages.

To avoid losing messages, we adopted the push-pull combined mode. The interaction is shown in the diagram. In addition, when loading a conversation on the frontend, it needs to know which time period of the conversation to pull. We introduced the concept of synchronization points, which is the timestamp of the latest synchronized message for that conversation. However, due to the possibility of multiple customer service staff viewing the same conversation and a single staff member logging in on multiple devices, the backend cannot effectively record frontend synchronization points. To solve this problem, we coded the frontend page so that the frontend calculates the synchronization point after receiving the message list, i.e., checking the largest timestamp. The schematic diagram of synchronization points is presented in Figure 2.

When a new message is generated, the server sends a notification to the client. The client then pulls the latest message list (the pull method differs from the initial screen loading method) after receiving the notification. When pulling, the client needs to bring the stored synchronization point as a parameter and update the synchronization point after receiving the latest message list.

5.2. Robot Conversational System

The specific process of the robot conversational system is illustrated in the diagram. First, user input is processed by the common question module, which returns answers

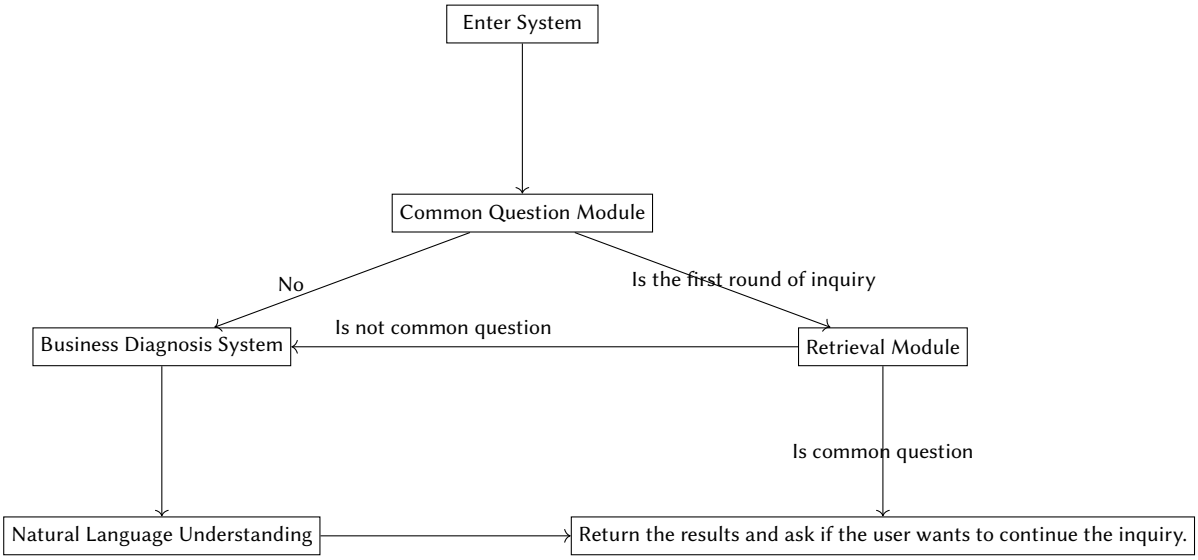

Figure 3: Flowchart of the Robot Conversational System

to frequently asked questions or transfers the request to the retrieval module if there is no relevant answer available. If the user feedback is still incorrect, the business diagnosis system will initiate a multi-turn dialogue to identify the specific issue the user wants to address. The system extracts the most relevant candidate item from the options and asks the user questions accordingly, providing answers and repeating the process until the system identifies the specific issue the user needs to address, such as "registering a new car."

Next, after obtaining the specific issue the user wants to address, natural language understanding algorithms are used to determine the user's intent, including retrieval needs, question-and-answer needs, and judgment needs based on application conditions, and assign corresponding tasks (such as retrieval, question-and-answer, and conditional reasoning) to provide accurate answers. The retrieval task retrieves relevant information from government documents, while the question-and-answer task extracts precise details from the relevant content of the user's query. The conditional reasoning task determines whether the user can carry out the desired task based on the given conditions. Finally, the system recommends related business types and frequently asked questions about the user's specific issue. If the user continues to ask questions, a multi-turn dialogue will be initiated to determine the relevancy of the user's query to the previous task, and the process will be repeated.

The business diagnosis system segments the user input into words and normalizes them to standard vocabulary. It uses a reinforcement learning model based on the DQN algorithm [8] to extract the most relevant candidate item from the options and initiates a multi-turn dialogue to identify the specific issue the user wants to address. The system asks questions based on the selected item and provides answers accordingly, repeating the process until the system identifies the specific issue the user needs to address.

The intent classification algorithm uses a Bert-based [9] intent classification model trained on a corpus. During the inference process, the user input is first matched against candidate intents based on the grammatical structure and domain-specific features of the corpus. If a match is found, the corresponding intent is returned. Otherwise, the user input is segmented and encoded using a word-embedding technique, and then fed into the model to predict the most probable intent with the highest probability as the output.

The subtask modules include question-and-answer based on business content, information retrieval based on business needs, and natural language reasoning based on application conditions. The question-and-answer module extracts precise details from relevant domain-related texts and returns them to the user, for example, returning the expiration year when the user asks about the validity period of a passport. This module uses a Bert-based answer extractor, segmenting and encoding the question input, and then passing it into the model to produce the specified answer output.

The retrieval module returns corresponding statements from documents based on user queries, such as handling procedures. The natural language reasoning module uses a knowledge-based stance detection model KB-BERT [10] trained on BERT network architecture

data to compare the user's input with each application condition to determine if it can meet the user's needs. The module then performs logical reasoning based on each inference result (including Entailment, Contradiction, and Neutral) and provides the final reasoning result based on each inference result (such as "able to handle" if all conditions are met, "unable to handle" if at least one condition is not met, or "showing all application conditions" if neither of the first two conditions are met).

5.3. Lessons Learned and Future Work

The development of this system involved a total of 9 people, taking two years, and consuming 18 person-years. The development process was not achieved overnight, but rather through endless cycles of discussions, weekly meetings, testing, bug reporting, and bug fixing. In the beginning, our developers each worked on the modules they were good at, with the developers of the artificial conversation system providing interfaces for the frontend developers, who would then create frontend pages based on the interface documentation. By the second version, the entire system was essentially in place, and we decided to build, train, and integrate the legal robot conversation system based on this platform. To save frontend developers the cost of secondary development, we had the conversation system directly connect to the robot's automatic reply system, meaning that from the frontend code's perspective, there would be no interaction with the robot's automatic reply system. As a result, from a data flow standpoint, messages would flow from the frontend to the artificial conversation system assistant, then through the artificial conversation system assistant to determine whether to send the message to the robot, and if so, send it to the automatic conversation system. The increase in the data flow chain led to the drawback of all problems being accumulated in the interaction between the two systems, making our bug-fixing time in this interaction particularly lengthy. As a result, we provided the community with a lesson: avoid piling up a large amount of work on a few developers responsible for certain modules, as this can lead to an especially long development cycle.

In the third version, we also introduced a plan to integrate the artificial conversation system with the robot system. Legal advisors would tag certain conversations and responses, applying a string-type label to the messages in the conversation, and then feed the tagged data back into the robot's training model, enabling the system's continuous evolution.

In the future, as we continue to develop the system, we may use formal verification methods to model and analyze the system to reduce the number of bugs generated during development and increase system security. This approach will further reduce debugging time and help us develop a reliable conversation system.

6. Related Works

Language modeling has been an active area of research in natural language processing (NLP) for several decades. One of such models is the recurrent neural network (RNN) [11], which uses a recurrent connection to maintain a hidden state that can encode information from previous words. Another popular RNN variant is the long short-term memory (LSTM) model [11], which uses gated units to selectively remember or forget information over time. Among those, a milestone model is the transformer, which uses self-attention to allow each word to attend to all other words in the input sequence.

Recently, large-scale pretraining based on transformer has emerged as a powerful technique for improving language models. Models such as BERT and GPT-2[12] are pretrained on massive amounts of text data and fine-tuned on downstream tasks such as question answering and text classification. There also have been several attempts to develop learning-based legal consultation systems in recent years. One such system is Ailira[13], a deep learning based legal assistant that can answer legal questions and provide advice based on Australian tax law. Another example is DoNotPay[14], a chatbot that helps users contest parking tickets and other minor legal issues. However, these systems are limited in their scope and may not be applicable to all legal issues.

Besides, the legal industry is gradually adopting artificial intelligence (AI) technologies to improve efficiency and reduce costs. Among these, natural language processing and machine learning are the most commonly used techniques, applied in areas such as contract review[15] [16], legal research[17], and document drafting[18] [19] [20].

Unlike the aforementioned applications, LAHEL takes a unique approach by focusing on addressing the pain points of users seeking legal advice. It provides an automated question-and-answer system specifically designed to cater to legal inquiries.

7. Discussion

LAHEL's main system consists of two primary components: the human dialogue system and the robot dialogue system. By leveraging models such as BERT and DQN algorithms for reinforcement learning, we have developed an automatic question-answering system that combines multiple models. Additionally, if users are unsatisfied with the robot's automatic response, they can switch to the human system for more professional answers. LAHEL, running on a 1-core, 2GB x86 commercial server,

can support up to 5,000 simultaneous users with a response latency not exceeding 400ms.

Currently, LAHEL has not yet been officially launched. After its official launch, more user experience data will be collected. Future work will focus on user experience testing and optimization for LAHEL.

8. Conclution

LAHEL represents a significant advancement in the field of legal consultation services by leveraging cutting-edge artificial intelligence technologies, such as BERT and DQN algorithms for reinforcement learning. By offering both a human dialogue system and a robot dialogue system, LAHEL caters to a wide range of user needs and preferences. It has the potential to democratize access to legal knowledge, particularly in developing countries, where traditional legal services may be financially or logistically prohibitive for many individuals.

The development and implementation of LAHEL also highlight the growing importance of interdisciplinary collaboration between the legal and computer science fields, which has led to novel solutions addressing real-world challenges. As LAHEL continues to evolve and mature, it is expected to not only provide more accessible legal services to the public but also contribute to increasing the work efficiency of legal professionals, thereby transforming the legal landscape.

Future research on LAHEL may focus on enhancing its performance and capabilities through the integration of additional advanced algorithms, further personalization of user experiences, and the exploration of potential applications in other areas of legal practice. By addressing these challenges and embracing future opportunities, LAHEL stands to become an invaluable tool for both legal professionals and the general public.

References

[1] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, Cambridge, MA, 2016.

[2] OpenAI, GPT-4 Technical Report (2023).

[3] G. Ogonjo, Florence, et al., Utilizing ai to improve efficiency of the environment and land court in the kenyan judiciary: Leveraging ai capabilities in land dispute cases in the kenyan environment and land court system, ASAIL LegalAIIA 2021 (2021) 59–68. URL: https://ceur-ws.org/Vol-2888/paper9.pdf.

[4] T. P. Sebastian Felix Schwemer, Letizia Tomada, Legal ai systems in the eu's proposed artificial intelligence act, ASAIL LegalAIIA 2021 (2021) 51–58. URL: https://ceur-ws.org/Vol-2888/paper8.pdf.

[5] Lahel-code (2023). URL: https://github.com/lawhelper.

[6] Lahel-project (2023). URL: https://miner.picp.net/#/im.

[7] T. B. Brown, et al., Language models are few-shot learners, arXiv preprint arXiv:2005.14165 (2020).

[8] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, volume 518, Nature Publishing Group, 2015, pp. 529–533.

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

[10] M. Malmsten, L. B"orjeson, C. Haffenden, Playing with words at the national library of sweden–making a swedish bert, arXiv preprint arXiv:2007.01658 (2020).

[11] A. Sherstinsky, Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network, arXiv:1808.03314 (2018).

[12] A. Radford, et al., Language models are unsupervised multitask learners, OpenAI blog 1.8 (2019): 9 (2019).

[13] ailira (2023). URL: https://www.ailira.com/build-a-legal-chatbot.

[14] Donotpay (2023). URL: https://donotpay.com/.

[15] Casetext - cocounsel (2023). URL: https://casetext.com/.

[16] lawgeex (2023). URL: https://www.lawgeex.com.

[17] rossintelligence (2020). URL: https://rossintelligence.com/features.

[18] K. D. Betts, K. R. Jaep, The dawn of fully automated contract drafting: Machine learning breathes new life into a decades–old promise, Duke Law Technology Review, 216–233 (2017).

[19] M. Legg, F. Bell, Artificial intelligence and the legal profession: Becoming the ai-enhanced lawyer, University of Tasmania Law Review, 38, 2, 34 –59 (2019).

[20] B. Barton, The lawyer's monopoly: What goes and what stays., Fordham Law Review, 82, 6, 3067 – 3090 (2014).