2023 Annual International Conference on Emerging Research Areas: International Conference on Intelligent Systems (AICERA/ICIS) | 979-8-3503-0345-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/AICERA/ICIS59538.2023.10420298

Design and Implementation of a Chatbot for Automated Legal Assistance using Natural Language Processing and Machine Learning

Ashok Reddy Kandula
Associate Professor, *Dept. of AI&DS*
SR Gudlavalleru Engineering College
Gudlavalleru, India
ashokreddy.gec@gmail.com

Mothilal Tadiparthi
Assistant Professor, Dept. of AI&DS
SR Gudlavalleru Engineering College
Gudlavalleru, India
mothilal556@gmail.com

Pooja Yakkala
UG Student, *Dept. of AI&DS*
SR Gudlavalleru Engineering College
Gudlavalleru, India
ypooja72003@gmail.com

Sindhu Pasupuleti
UG Student, *Dept. of AI&DS*
SR Gudlavalleru Engineering College
Gudlavalleru, India
sindhu.gdlv@gmail.com

Poojitha Pagolu
UG Student, Dept. of AI & DS
SR Gudlavalleru Engineering College,
Gudlavalleru, India
pagolupoojitha135@gmail.com

Sai Mohana Chandrika Potharlanka
UG Student, Dept. of AI & DS
SR Gudlavalleru Engineering College,
Gudlavalleru, India
chandupotharlanka@gmail.com

***Abstract*— Legal research is a time-consuming and complex task that requires a deep understanding of legal language and principles. To assist lawyers and legal professionals in this process, an AI-based legal assistance system can be developed that utilizes natural language processing (NLP) and machine learning algorithms. This system would be capable of conversing with clients, including lawyers or general users, and retrieving the most similar or matched laws related to the query given by the client. The retrieved laws would be ranked based on the number of hits and losses and retrieved according to their similarity. With the use of these algorithms, the system would be able to recognize pertinent terms, ideas, and connections among legal texts, which would be used to retrieve and rank the most pertinent laws. To evaluate the effectiveness of this system, an accuracy rate of over 80% has been achieved. This level of accuracy is significant as it reduces errors in legal research and improves the quality of legal advice. Overall, the proposed AI-based legal assistance system has the potential to revolutionize the legal industry and bring about significant positive impacts by providing fast, efficient, and accurate legal assistance. Further research could focus on developing additional features, such as case law analysis, contract review, and legal drafting, to improve the system's capabilities and expand its usefulness.**

Keywords*— *Legal Assistance Chatbot, Natural Language Processing (NLP), Machine Learning (ML), Legal Consultation, Law Retrieval, Case law analysis, Pertinent Terms, Bag-of-Words (BoW), Similarity Matching Function, Ranking Algorithms

I. INTRODUCTION

With its widespread use of complicated terminologies and legal processes, the legal sector is one of the most challenging and sophisticated in the world. An estimated 5 billion people worldwide lack meaningful access to justice, with many of them belonging to low-income and marginalised communities, according to a new survey by the World Justice Project [9][1]. The usage of chatbots for legal support offers the potential to close this gap and increase everyone's access to legal advice. According to a new analysis by Grand View Research [10], the global market for legal chatbots will be worth USD 1.5 billion by 2025, expanding at a CAGR of 29.3% between 2019 and 2025. This expansion can be attributed to advancements in machine learning (ML) and natural language processing (NLP) technology as well as the surge in the demand for rapid and accurate legal advice. Use of automated technology, such as chatbots, may also help bring down the price of legal services by up to 80%, enabling companies to provide affordable legal services to a larger range of clients. According to a McKinsey & Company research, this is the case [11].

The design and development of the Lexi AI chatbot, which can deliver precise and timely answers to legal questions, is suggested by this study. The chatbot uses machine learning (ML) and natural language processing (NLP) algorithms to analyse and comprehend user queries, find pertinent legal knowledge from a sizable collection of legal resources, and give suitable legal advice. With an accuracy rate of over 80%, Lexi AI eliminates research errors and enhances the quality of legal advice, offering a viable response to the problems facing the legal sector. This chatbot has the power to transform the legal sector, increase access to justice, and lower the price of legal services. Additionally, it can fill the vacuum in legal assistance for underserved and marginalised populations, advancing social justice.

The possible utility of Lexi AI is illustrated by several examples. The chatbot, for instance, can be used by those looking for legal counsel on family law, property disputes, or consumer protection.

A. Case 1: Hit and Run

Ravi Kumar, a resident of Banglore, has been accused of a hit and run case, and a First Information Report (FIR) has been filed against him at the local police station. Ravi is worried about the legal procedures involved in the case and wants to consult a lawyer to understand his legal rights and obligations. A practical solution is a chatbot that provides legal advice. Ravi can ask queries like "What are my legal rights in a hit-and-run case?" via a messaging service or smartphone app. His rights under the Indian Penal Code and the Criminal Procedure Code are outlined by the chatbot, which uses NLP and ML to offer prompt and precise responses. It also discusses the legal process and any possible repercussions. Ravi saves time and money by receiving easily available legal counsel.[4]

B. Case 2: Theft Accusation

Mr. Ram is facing theft allegations in Delhi and is in need of emergency aid but is confused by the legal system. The attorney retained by the court is preoccupied with other cases. As a helpful tool, a chatbot for legal advice enters the picture.

Mr. Ram can access it via his smartphone to learn more about his charges, the bail process, and the court system. His confusion and worries are reduced by the chatbot's precise advice. Furthermore, the chatbot's capacity to provide essential legal data helps Mr. Ram's attorney develop a compelling defence. In the end, this chatbot for legal aid helps Mr. Ram receive a fair trial, improve access to justice, and respect the principles of India's Criminal Procedure Code.

C. *Case 3: Theft Allegations*

In Mumbai, Krishna faces theft accusations, and his family seeks legal guidance but can't afford a lawyer. Our proposed legal assistance chatbot comes to the rescue. Accessible through a website or mobile app, it offers quick and precise responses regarding the Criminal Procedure Code (CrPC) in India. Krishna's family can input queries like "What are the legal procedures for a person in police custody?" or "What are Ravi's rights as an accused?" The chatbot, powered by NLP and ML algorithms, comprehends these questions and retrieves pertinent information from an extensive legal database, including the CrPC and relevant case laws.

It supplies accurate answers, such as "Under the CrPC, a person in police custody has the right to legal representation" or "Ravi, as an accused, has the right to be informed of the charges against him." Furthermore, the chatbot offers insights into legal aid services and pro-bono lawyers who can represent Krishna. It outlines eligibility criteria and application procedures for these services.

II. Literature Review

The literature review highlights the growing interest in the use of chatbots for legal assistance, reflecting their potential to transform the legal industry. Several studies have examined the effectiveness of legal chatbots in different contexts and jurisdictions.

In terms of chatbot methodologies, two predominant categories are rule-based and machine learning-based approaches. Rule-based chatbots rely on predefined decision trees developed by legal experts, which guide users through a set of predetermined rules. These chatbots excel at providing basic legal information and answering straightforward queries but may struggle with complex or unanticipated questions.[6][7]

Contrarily, chatbots powered by machine learning use natural language processing (NLP) and machine learning strategies to comprehend and address customer inquiries. These chatbots can recognise patterns and give precise responses because they have been educated on large quantities of legal data. Additionally, they can keep up with regular training to enhance their performance. IBM Watson Legal, LawBot, DoNotPay, and ABA Free Legal Answers are a few examples of chatbots that provide legal assistance and each one has a different strategy and set of features.

The choice between rule-based and machine learning-based chatbots depends on the specific use case and complexity of legal queries, with machine learning-based chatbots offering broader capabilities and adaptability.

III. Methodology

A. *Proposed Method*

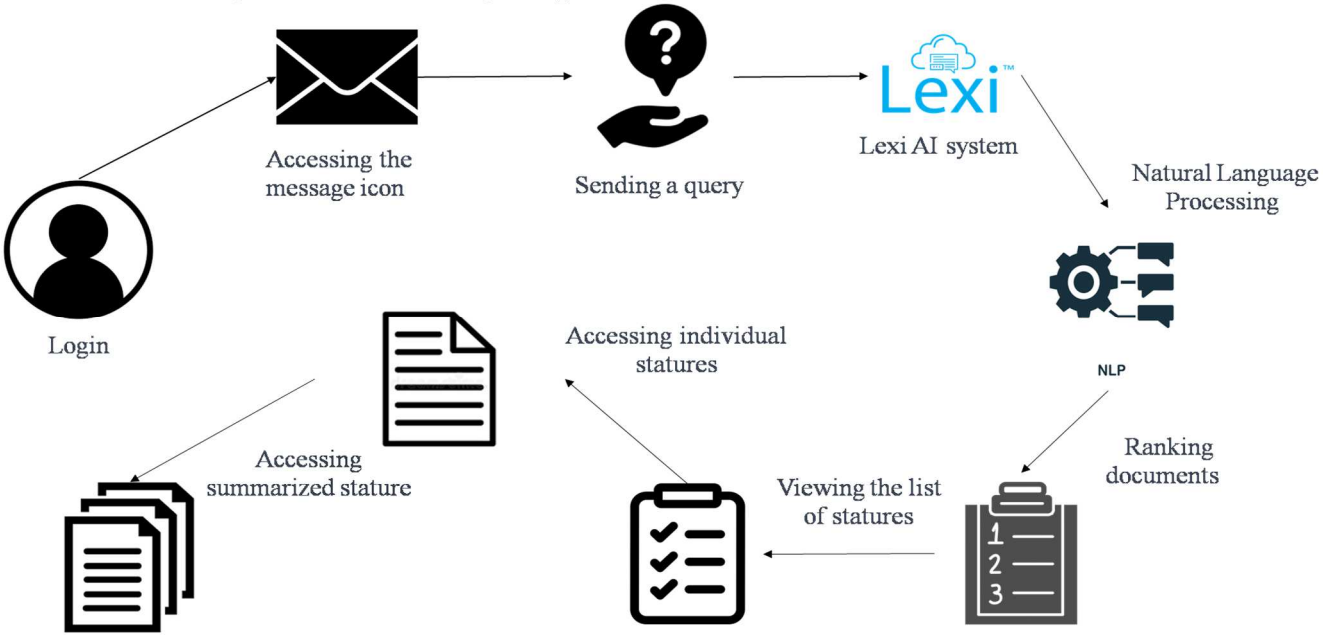

Fig. 1. Architecture diagram of Design and Implementation of a Chatbot for Automated Legal Assistance using Natural Language Processing and Machine Learning

The figure 1 shows the following points -

Logging into the account: Users access the application by entering their login credentials, granting them access to all its features.

Accessing the message icon: Users find the message icon on the application's interface, usually in the toolbar or main menu, to initiate communication with the Lexi AI system.

Sending a query to the Lexi AI system: In the chat window, users type their queries and send them to Lexi AI. Lexi AI's NLP algorithms understand the query and retrieve relevant documents.

Ranking documents: Lexi AI employs machine learning algorithms to prioritize documents, presenting the most relevant ones first for easier information retrieval.

Viewing the list of statutes: Users can see a list of statutes with chapter names and titles, facilitating quick document identification.

Accessing individual statutes: Clicking on a statute from the list opens the full text, enabling users to read it in its entirety.

Accessing summarized statutes: Lexi AI's machine learning algorithms create concise summaries of statutes, offering users key points for quick comprehension without the need to read the entire text.

B. Implementation

The Lexi AI chatbot makes use of a set of advanced algorithms that work across four different components to deliver a convenient and effective user experience. Each of these parts carries out a specific task that adds to the system's overall functionality.

- NLP Text Analysis and Pre-Processing
- Stature Similarity Calculation
- Stature Ranking Algorithm
- Stature Summarization Engine

The pre-processing of the user's query utilising sophisticated text analysis techniques including tokenization, stemming, and part-of-speech tagging, as well as the elimination of irrelevant words and phrases, are described in this approach. The pre-processed data is then saved in a .JSON file format for easy access and quicker retrieval by the algorithm, which repeats similar processes for each of the system's statistics.

NLP Text Analysis and Pre-Processing:

Input: User Query

Output: Pre-Processed Data in .JSON Format

1. Tokenize the user query into individual words, phrases, or other meaningful units.
2. Perform stemming to reduce words to their base form to improve matching accuracy.
3. Use part-of-speech tagging to assign a label to each word in the query to identify its grammatical role in the sentence.
4. Remove any stop words or noise words that are not relevant to the query.
5. Repeat steps 1-4 for all the statures (laws) in the system.
6. Save the pre-processed data in a .JSON file for easy access and faster retrieval.
7. Return the pre-processed data in. JSON format.

The NLP (Natural Language Processing) pre-processing component is the initial step in the Lexi AI chatbot's information retrieval process. The Algorithm's primary role is to analyze the user's query and extract essential information for retrieving pertinent documents.

This pre-processing phase employs various NLP techniques like tokenization, stemming, and part-of-speech tagging to dissect the user's query. Tokenization divides the query into meaningful units, while stemming reduces words to their base form for more accurate matching. Part-of-speech tagging assigns grammatical roles to query words.

Subsequently, non-relevant elements like stop words (common language fillers like "the," "and," "is") and noise words (e.g., pronouns) are eliminated from both the query and the system's database of statutes (laws). This enhances the system's ability to quickly access and retrieve pertinent information.

The pre-processed data is then stored in a .JSON file format, ensuring easy accessibility and faster retrieval in subsequent information retrieval steps. .JSON files are versatile and compatible with various programming languages and software tools, facilitating integration with other systems.

In this research paper, the dataset utilized is based on the Criminal Procedure Code, a pivotal legal document outlining procedures for investigating criminal cases, including arrest, bail, trial, and sentencing. The dataset comprises 484 sections across 37 chapters, accompanied by 2 schedules and 56 forms. Each section is stored in .txt files, with filenames reflecting the chapter, section title, and subsection if applicable (e.g., C1S1.txt).

Stature Similarity Calculation:

Input:

pre-processed query, database of statutes (laws), Bag-of-words model

Output:

Ranked list of statutes based on their similarity score with the user query

Algorithm:

1. For each statute in the database:
   a. Apply pre-processing to the statute text
   b. Calculate Jaccard similarity between pre-processed query and pre-processed statute text
   c. Calculate cosine similarity between pre-processed query and pre-processed statute text
   d. Calculate an overall similarity score by weighting Jaccard and cosine similarities
   e. Add the statute and its similarity score to a list
2. Sort the list of statutes by their similarity score in descending order
3. Return the sorted list of statutes

The stature similarity algorithm plays a pivotal role within the Lexi AI chatbot's architecture. After the user's query undergoes pre-processing, this function becomes instrumental in the information retrieval process. Its core purpose is to assess the similarity between the pre-processed query and the database of statutes (laws).

To fulfil this objective, Lexi AI employs two distinct similarity functions: Jaccard similarity and cosine similarity. The Jaccard similarity function evaluates the match between unique words present in both the query and the documents,

while the cosine similarity function employs a bag-of-words model, considering the frequency of unique words in the documents and query. Cosine similarity further evaluates the angle of similarity between the query and the documents.

Comparatively, the cosine similarity function proves to be more accurate than Jaccard similarity. It not only takes into account unique words but also considers their frequency within the documents. This comprehensive approach allows the system to precisely gauge the similarity between the user's query and the statutes in the database, leading to the retrieval of the most pertinent statutes.

Following the similarity calculations, the documents (statures) are ranked based on their similarity scores. This ranking function ensures that the statutes with the highest similarity to the user's query are positioned at the top of the list. This arrangement serves to save the user's time by presenting the most relevant documents as a priority.

Stature Ranking Algorithm:

Input:

Pre-processed query, List of documents and their associated metadata, Trained machine learning model for ranking

Output:

List of ranked documents in order of relevance

Steps:

1. Initialize an empty list to store the ranked documents.
2. For each document, calculate the similarity score between the pre-processed query and the document using the similarity function.
3. Feed the document and its associated metadata into the trained machine learning model to obtain a relevance score.
4. Calculate the final rank score for the document by multiplying the similarity score and the relevance score.
5. Append the document and its rank score to the list of ranked documents.
6. Sort the list of ranked documents in descending order based on the rank score.
7. Return the list of ranked documents.

The ranking algorithm within the Lexi AI chatbot system is a crucial element that ensures the retrieval of the most pertinent documents and their presentation to the user. It leverages machine learning algorithms to assess documents based on several criteria, including their relevance to the user's query, their authority, and their recency.

While the ranking function utilizes the similarity scores generated by the similarity function, it takes a more sophisticated approach by considering additional factors. Machine learning algorithms within the ranking function take into account parameters like the number of hits and misses for each document, document length, and how frequently the document has been viewed by other users. These algorithms are trained on a substantial dataset of documents and their associated metadata to accurately predict the relevance of each document to a given query.

The outcome of the ranking function is a meticulously ordered list of documents, with the documents most relevant to the user's query prominently featured at the top. This meticulous ranking process ensures that the chatbot system consistently delivers the most relevant documents to the user. This, in turn, enhances the user experience and bolsters the effectiveness of the Lexi AI chatbot system in providing precise and tailored information.

Summarizer:

Inputs:

Document text, Summary length (in number of sentences)

Outputs:

Summary text

1. Apply pre-processing techniques to the document text, such as removing stop words and punctuation.
2. Identify the most important sentences in the document using techniques such as text ranking and keyword extraction.
3. Select the top sentences that capture the essential information of the document, based on the summary length input.
4. Combine the selected sentences to create the summary text.
5. Return the summary text as the output.

The Lexi AI chatbot's summarizer algorithm plays a crucial role in delivering concise summaries of relevant documents to users. It employs machine learning algorithms to extract key points from documents, facilitating quick comprehension.

The summarizer targets the top-ranked documents identified by the ranking function. To generate a summary, it first applies pre-processing techniques, removing stop words and punctuation from the document. Next, it identifies critical sentences using methods like text ranking and keyword extraction. The summarizer then selects the most important sentences to construct a summary that encapsulates the document's essential information.

Furthermore, the summarizer considers the length of the summary and adjusts it as necessary. If the summary is excessively long, it may shorten it by omitting less critical sentences or phrases. The final summary is presented to the user in a concise and user-friendly format, enabling them to swiftly grasp the document's key takeaways. This not only saves time but also enhances the user experience by providing an efficient and effective means of accessing pertinent information.

IV. Results and Discussion

The provided data shows the similarity scores of pre-processed user queries with various sections of the Criminal Procedure Code. The sections are stored in .txt files with chapter name section title (Subsection name) like C5S51.txt, C12S167.txt, etc.

The similarity scores have been calculated using two types of similarity functions - Jaccard similarity and cosine similarity. Jaccard similarity considers unique words in the documents and query, and matches them together. Cosine

similarity, on the other hand, uses the bag of words model and counts the number of times the unique words are repeated in the document. It then checks the angle of similarity.

A. Jaccard Similarity & Cosine Similarity

Test Case 1:

TABLE.1. Test case 1 results for Jaccard similarity

Testcase	Jaccard Similarity
C6S61.txt	1
C6S62.txt	1
C6S63.txt	1
C6S64.txt	1
C6S65.txt	1
C6S66.txt	0
C6S67.txt	0

The above are the results of the Jaccard Similarity are of test case 1 for 30 files, showcasing the matching of words in the query with the documents. In the results, '1' indicates that a word in the query matches with the corresponding document, '2' indicates two words match, while '0' represents no matches between the query and the document.

Here Ravi of test case 1 sends the query to the system. 'Legal', 'Rights', 'Hit', 'Run', 'Police', 'custody', 'accused', 'charges', 'title' are the keywords in the query.

TABLE.2. Test case 1 results for Cosine similarity

Testcase	Cosine Similarity
C5S51.txt	0.371518808383564
C12S167.txt	0.358346677616947
C5S50.txt	0.354942603766444
C5S55A.txt	0.348155311911397
C33S436.txt	0.344725126663341
C33S441A.txt	0.333333333333333
C33S437.txt	0.331685753411095

The highest similarity score was obtained for section C5S51.txt, which has a score of 0.37151880838356854. This section is recommended as the most relevant section for Case 1. The other recommended sections, in order of decreasing relevance, are C12S167.txt, C5S50.txt, and C5S55A.txt.

The results of this query will be sent to Ravi, who can review the recommended sections and use them to further research and provide legal advice for Case 1.

However, some of the sections have a similarity score of 0.0, which means that there is no similarity between the pre-processed user query and the content of those sections. It is possible that the pre-processed query did not contain any relevant keywords or the sections themselves are not relevant to the user's query.

Test Case 2:

TABLE.3. Test case 2 results for Jaccard similarity

Testcase	Jaccard Similarity
C12S171.txt	1
C12S172.txt	1
C12S173.txt	2
C12S176.txt	1
C14S191.txt	1
C14S197.txt	1
C14S199.txt	1

Here are the results of the Jaccard Similarity for test case 2, demonstrating the matching of words between the query and the documents. Each case is represented by a binary value, where '1' indicates a word match between the query and the corresponding document, while '0' signifies no matches.

Here Ram of test case 2 sends the query to the system that contains the following keywords 'charges', 'bail', 'process', 'legal', 'procedures', 'accused', 'theft', 'Sections', '379', 'Indian', 'Penal', 'Code'.

TABLE.4. Test case 2 results for Cosine similarity

Testcase	Cosine Similarity
C27S357B.txt	0.412822643772448
C21S260.txt	0.376186450430324
C34S455.txt	0.353328125587541
C17AS220.txt	0.335354593967646
C17AS211.txt	0.317287158485172
C17AS219.txt	0.285988002161266
C12S154.txt	0.276413629153884

Based on the results of the test case 2, it appears that the document C27S357B.txt is the most relevant document with a score of 0.4128226437724408. This could potentially be a document related to theft under Section 379 of the Indian Penal Code or have information that could be useful in Mr. Ram's case. The document C21S260.txt also received a relatively high score of 0.37618645043032134, indicating that it may also contain information relevant to the case. It is unclear what this document may contain, but it could be worth further investigation.

On the other hand, few documents received a score of 0.0, indicating that they may not be relevant to the case at hand. However, it is important to note that the algorithm used for scoring is not perfect and there is still a possibility that these documents could contain relevant information. Overall, it is recommended that the most relevant documents be thoroughly reviewed and analysed to determine if they contain information that could be useful in Mr. Ram's case.

Test Case 3:

TABLE.5. Test case 3 results for Jaccard similarity

Testcase	Jaccard Similarity
C6S74.txt	1
C6S75.txt	1
C6S76.txt	1
C6S77.txt	1
C6S78.txt	1
C6S79.txt	1
C6S80.txt	1

Above are the results of the Jaccard Similarity for test case 3, illustrating the word matching between the query and the documents. Each case is denoted by a binary value, where '1' indicates a word match between the query and the respective document, while '0' indicates no matches.

Here Ravi of test case 3 sends the query to the system. 'Legal', 'Rights', 'Hit', 'Run', 'Police', 'custody', 'accused', 'charges', 'procedure' are the keywords in the query.

TABLE.6. Test case 3 results for Cosine similarity

Testcase	Cosine Similarity
C5S51.txt	0.371518808383564
C12S167.txt	0.358346677616947
C5S50.txt	0.354942603766444
C5S55A.txt	0.348155311911397
C33S436.txt	0.344725126663341
C33S441A.txt	0.333333333333333
C33S437.txt	0.331685753411095

The results for test case 3 show that C5S51.txt and C12S167.txt have the highest similarity scores with the given scenario, with scores of 0.37151880838356854 and 0.3583466776169477, respectively. These documents may contain useful information related to legal procedures for a person under police custody, the rights of an accused, and the procedure for bail in theft cases. On the other hand, C1S4.txt has a score of 0.0, indicating that it may not be relevant to the given scenario.

However, the quality and accuracy of the retrieved information should be carefully assessed, and further legal advice should be sought before taking any action.

B. Resultant Statement

Cosine Similarity is widely regarded as a more effective similarity measure than Jaccard Similarity, especially when it comes to accurately matching files or documents. The superiority of Cosine Similarity becomes evident in its ability to capture more nuanced aspects of the text.

In Jaccard Similarity, the system focuses solely on identifying unique words in the files and matches them accordingly. While this approach can be useful in certain scenarios, it lacks the ability to account for the frequency or importance of words within the documents. Consequently, Jaccard Similarity may overlook the significance of certain terms that appear more frequently in a particular file. In the above test results, we can find a lot of files with '1' which means they have the same importance in ranking. This makes it hard for the system to identify the most accurate and correct file to retrieve for the user.

On the other hand, Cosine Similarity offers a more comprehensive analysis by considering not only the unique words in the files but also the frequency of each word. By taking into account the frequency distribution of words, Cosine Similarity provides a more accurate representation of the documents' content and their similarity to the query. This is shown in the above test cases in which there is a similarity ratio that gives us the accurate values for each file in dataset.

CONCLUSION

In conclusion, the Lexi AI chatbot consists of various algorithms working together to retrieve relevant information for the user's queries. The NLP pre-processing algorithms analyze the user's query to identify keywords, phrases, and other relevant information while removing any stop words or noise words. This pre-processed data is saved in a .JSON file for easier access and faster retrieval. The similarity function calculates the similarity between the user's query and the various documents in the database based on the number of words matching the individual stature with the query. The ranking function then ranks the documents based on their relevance and other criteria, such as authority and recency. Finally, the summarizer generates a concise summary of the most important points in the document.

These algorithms work together to provide a seamless experience for the user, allowing them to quickly and easily access the information they need. With an accuracy rate of over 80%, the system reduces errors in legal research and improves the quality of legal advice. The potential positive impact of this system on the legal industry is significant. It could save legal professionals a significant amount of time and effort, thereby increasing productivity and reducing costs. Future research could focus on developing additional features, such as case law analysis, contract review, and legal drafting, to further enhance the system's capabilities and expand its usefulness. The proposed AI-based legal assistance system has the potential to revolutionize the legal industry and bring about significant positive impacts.

REFERENCES

[1] Mohit Khatri, Arnab Agrawal and Ajay Garg, "PoliceBOT - An Informative RASA Powered Chatbot based Crime Registration and Crime Awareness System", *International Research Journal of Engineering and Technology*, vol. 08, no. 06, June 2021.

[2] Shraddha Surana, Jayanth vikram Chekkala and Prateek Bihani, "Chatbot based Crime Registration and Crime Awareness System using a custom Named Entity Recognition Model for Extracting Information from Complaints", *International research journal of engineering and technology*, vol. 08, no. 04, Apr 2021.

[3] Robert Ambrogi, *This Week in Legal Tech: Everyone's Talking About Chatbots*, Above the Law (Apr. 17, 2017), https://abovethelaw.com/2017/04/this-week-in-legal-tech-everyones-talking-about-chatbots/?rf=1.

[4] Surbhi Khokhawat, Ishita Jain, Garvita Shekhar, Harshshrimali, Shifa Choudhary and Preeti Narooka, "EFFICIENT CHATBOT FOR CRIME AWARENESS SYSTEM", *International Research Journal of Modernization in Engineering Technology and Science*, vol. 03, no. 05, May 2021.

[5] Sudarshan Jagdale, Piyush Takale, Pranav Lonari, Shraddha Khandre and Yogesh Mali, "Crime Awareness and Registration System", *International Journal of Scientific Research in Science and Technology*, vol. 5, no. 8, November-December 2020.

[6] Choi, Jonathan H. and Schwarcz, Daniel B., AI Assistance in Legal Analysis: An Empirical Study (August 13, 2023). Minnesota Legal Studies Research Paper No. 23-22, Available at SSRN: https://ssrn.com/abstract=4539836 or http://dx.doi.org/10.2139/ssrn.4539836

[7] Kim, S.; Kwon, O.-W.; Kim, H. Knowledge-Grounded Chatbot Based on Dual Wasserstein Generative Adversarial Networks with Effective Attention Mechanisms. *Appl. Sci.* **2020**, *10*, 3335. https://doi.org/10.3390/app10093335.

[8] Shaun V. Ault, Rene J. Perez, Chloe A. Kimble and Jin Wang, "On Speech Recognition Algorithms", *International Journal of Machine Learning and Computing*, vol. 8, no. 6, December 2018.

[9] World Justice Project. (2020). Measuring the Justice Gap: The World Justice Project's 2020 Global Insights on Access to Justice. Retrieved from https://worldjusticeproject.org/sites/default/files/documents/Measuring-the-Justice-Gap_0.pdf

[10] Grand View Research. (2019). Legal Chatbots Market Size, Share & Trends Analysis Report by Type (Software, Services), By Deployment Mode (On-premise, Cloud), By Organization Size, By Application, By Region, And Segment Forecasts, 2019-2025. Retrieved from https://www.grandviewresearch.com/industry-analysis/legal-chatbots-market

[11] McKinsey & Company. (2019). The Business Case for Automation in the Legal Industry. Retrieved from https://www.mckinsey.com/industries/professional-services/our-insights/the-business-case-for-automation-in-the-legal-industry.

[12] Brooks, C., Gherhes, C., and Vorley, T. (2020). Artificial intelligence in the legal sector: pressures and challenges oftransformation. Cambridge Journal of Regions, Economy and Society, 13(1):135–152.

[13] Alarie, B., Niblett, A. and Yoon, A. H. (2018) How artificial intelligence will affect the practice of law,University of Toronto Law Journal, 68: 106–124.

[14] Dzienkowski, J. S. (2013) The future of big law: alternative legal service providers to corporate clients,Fordham Law Review, 82: 2995–3040.

[15] McGinnis, J. O. and Pearce, R. G. (2014) The great disruption: how machine intelligence will transform the role of lawyers in the delivery of legal services, Fordham Law Review, 82: 3041–306.