Enhancing Legal Aid in Aotearoa with Large Language Models

Nathan Bennett

***Abstract*—This paper presents the development of a legal aid application designed to address the barriers to accessing justice in Aotearoa, particularly for small civil disputes in consumer law. The high cost of legal services and the limitations of existing alternatives have created a need for an accessible, accurate, and cost-effective solution. Our approach leverages advanced large language models to create a conversational chatbot tailored to New Zealand's consumer law framework. Testing conducted found the project to meet responsiveness and cost requirements, but show varying accuracy of responses, with only some meeting the required thresholds.**

***Index Terms*—Legal aid, Chatbot, Large language models, Retrieval-augmented generation, Consumer law.**

I. Introduction

THE New Zealand Bar Association has identified legal aid as one of the most significant barriers to access to justice in Aotearoa [1]. Hiring a lawyer is the most prominent form of obtaining legal aid, however, hourly rates for lawyers in Aotearoa typically vary between $200 and $800 per hour [2]. This high price makes hiring a lawyer for a small legal dispute impractical as the legal fees will likely exceed the value gained from winning the dispute. The first alternative for Aotearoa citizens is to request advice online through a source such as Citizens Advice Bureau, [3] but receiving a response can be slow and the service requires large funding, largely through government departments. The other alternative is to research through journals and articles, however, applying this information to personal circumstances can be difficult for people not educated in law.

A. Background

The field of large language models (LLMs) has evolved significantly since the debut of early models such as the Eliza Language model in 1966 [4]. Recent advancements, particularly in fine-tuning and prompt engineering, have enabled models to perform tasks beyond the purpose they were created. This was initially exemplified by breakthrough models such as BERT and further utilised with more sophisticated models such as Claude and GPT-4 [5]. These breakthroughs have made it possible to develop sophisticated applications, such as chatbots, that can offer specialised assistance.

B. Solution

A legal aid application with a chatbot has been identified as a cost-effective solution to improving access to legal aid in small civil disputes, providing Aotearoa citizens with timely legal assistance. This project develops a conversational chatbot designed to provide advice based on a user's situation. Users interact with the chatbot through a text-based interface, allowing for a seamless conversation. Additionally, the application offers tools to help users take appropriate actions based on their legal situations. With government funding, this could become a free and accessible platform for all citizens to obtain legal aid.

The functionality is created by leveraging existing LLMs and drawing from New Zealand's legal framework. The scope of the application has been narrowed to focus solely on consumer law with the application utilising resources taken from New Zealand's consumer law framework. Consumer law was chosen as disputes tend to be over low-value items where citizens can be discouraged from taking action due to the inherent high cost brought by lawyers. This specification of scope serves to reduce the resources required to develop a comprehensive database of relevant material, and potentially increase performance.

The chatbot utilises existing LLMs for their ability to analyse and respond to complex queries by processing large amounts of text data. These models are potentially suited for legal queries due to their capacity to interpret language and offer responses specific to their training data.

C. Goals

- The purpose of the project is to provide accurate and comprehensive legal aid. A high accuracy, particularly with limited inaccuracies, is important because giving incorrect advice can potentially cause harm and introduce liabilities. As such, the primary goal for the project is for every response to achieve a minimum accuracy of "nearly all correct" and minimum completeness of "adequate". A minimum threshold is more important than the average due to the potential harm caused by inaccurate responses producing incorrect information.
- Performance is measured further through the average time taken for responses with a goal of a median response time of 2.3 with a deviation of 0.5 seconds, giving a target range between 1.8 and 2.8 seconds. This is done to ensure that responses are timely, whilst maintaining user trust.
- The average cost per user query also requires minimisation with a target of less than $0.10 per query. Cost is introduced through the price of accessing LLMs through APIs and is unavoidable for a project with remote LLM access. A low cost will increase the likelihood of the project being suitable for adoption by a government de-

This project was supervised by Dr Andrew Lensen.

partment, which would allow for free access for Aotearoa citizens.

D. Sustainability

This project is designed with considerations towards social sustainability, aligning with some of the United Nations Sustainable Development Goals (SDGs) [6]. The project aims to democratise access to legal information, benefiting individuals who cannot afford traditional legal services. Deployment of the chatbot as a free service provides a cost-free alternative to professional legal advice. This initiative addresses economic inequalities, contributing to SDG Goal 10 [6]. By enhancing access to justice, the project also supports SDG Goal 16 [6]. Furthermore, the usage of compute resources, particularly in the training of models, has been considered and minimised, as is reflected in the design decisions made later in this report.

II. Related Work

A. Existing Approaches

Because the field of generative AI is still emerging, there are a limited number of existing similar solutions for this problem. One such solution, outlined in the paper "Transforming Legal Aid With AI" similarly attempts to address the difficulties of navigating legal systems [7]. To do so the authors outline a framework to adapt LLMs to be used in legal contexts.

The implementation of this approach involves the development of a framework that integrates both intention and context elicitation, guiding the system to ask follow-up questions. Prompts and in-context learning techniques help the LLM generate specific questions to elicit more comprehensive responses from users. The chatbot developed gathers all useful information from users before generating a legal response. This methodology in the paper is similar to that being developed in this project, but a significant difference is the lack of an external database used by the solution outlined in the paper. This results in the chatbot relying solely on the resources included in its LLM's training data.

This approach produces enhanced accuracy as more details on the client's situation lead to more precise responses. And importantly, the model also succeeds in eliminating required human interaction from a legal expert, as the model is completely independent, outside of the client. However, no testing had yet been done on the system meaning any comments on the accuracy are only hypothetical. Ensuring the LLM does not produce hallucinations is also critical, and not mitigated within the solution. Any client using the system would still be advised to have responses reviewed by attorneys. This is an innate problem faced with any applications of LLMs to legal contexts. Due to this, validation, including expert reviews by legal professionals, will be used to mitigate this problem in our project.

Another attempt at legal aid is outlined in the paper "Lawformer: A pre-trained language model for Chinese legal long documents" [8]. The paper develops a new LLM trained on Chinese legal resources for the purpose of answering questions related to Chinese law. Tens of millions of case documents published by the Chinese government were collected and used as training data. They then utilise an encoder and train the model with a learning rate of $5{\times}10^{-5}$, a sequence length of 4,096, and a batch size of 32.

The paper was successful in its goal of creating a LLM and their evaluation proved a slight increase in the accuracy of answering Chinese legal queries when compared to existing, generic LLMs such as BERT. However, as this paper was published in 2021 BERT is no longer a good benchmark for performance. Another model would likely have to be retrained using newer transformer methods to keep up with the performance of more recent LLMs. This is an issue facing all domain-specific LLMs as the constant advancements of generative AI constantly increase what is considered to be high-performing. The cost of training such a model is also too expensive for this project due to the monetary cost of hiring GPUs to perform the training. Additionally, these GPUs consume a significant amount of electricity, which does not align with the environmental goals of this project.

B. Ethical Analysis

The integration of LLMs into the legal field poses significant ethical challenges that are worth noting. One primary concern is the potential for bias within these models. Data used to train an LLM may not properly represent a population, causing generalisation of groups, leading to biased results [9]. An example of this is seen through LLMs such as ChatGLM2 showing a tendency to convict males more frequently than females in similar circumstances [10]. The nature of LLM decision-making processes can make it difficult to identify biases, necessitating evaluation of the system to detect any inaccuracies in results.

Another ethical issue is the reliability and trustworthiness of LLMs in providing legal advice [11]. Legal professionals are subject to strict ethical standards such as maintaining client confidentiality and representing clients competently. However, LLMs can produce hallucinations and cite non-existent sources, potentially misleading users and resulting in severe consequences [12]. Models have been found to tend to exaggerate effects that are present in humans, in part by reducing variance, leading to outcomes that may be incorrect [13]. As a result, the evaluation of suitable LLMs for this project includes the evaluation of both reliability and trustworthiness. Furthermore, we have a legal expert assisting with the project to provide advice regarding legal aid.

C. Retrieval-Augmented Generation

The field of Retrieval-Augmented Generation (RAG) has seen rapid development in recent years, primarily driven by the limitations of LLMs in handling domain-specific and knowledge-intensive tasks. RAG is a technique that integrates dynamic retrieval mechanisms with LLMs, allowing models to access and incorporate up-to-date information from external sources. This serves to enhance the factual accuracy and relevance of generated content [14]. The technique is commonly applied to enhance the performance of LLMs on knowledge-intensive generation tasks, like document-based question answering [15]. Given a question, a retriever module is used to obtain multiple relevant passages across potentially different documents, then input to the model as additional context for generating an answer [16].

This applies to the project for introducing relevant legal information that was not part of the dataset the LLM was trained on. The LLM's dataset will likely lack specific knowledge of New Zealand consumer law, making the incorporation of pertinent legal information a necessity. Another problem addressed by RAG is that LLM models are typically bound by their training end date, which means they lack awareness of new developments beyond that point. Consequently, they operate with outdated knowledge, which RAG solves by providing real-time information from external sources [14]. In the context of the project, this would allow the system to remain relevant and be easily updated in the case that the law changes.

D. LLM Selection Metrics

Many different metrics can be used to evaluate LLMs, with the following metrics contributing to the model selection process of this project. The two most common metrics are accuracy and F1 score. Accuracy measures the percentage of correct responses in tasks such as text classification or question answering, indicating how well the expected output aligns with the model's results. F1 score combines precision and recall providing a more balanced assessment by considering the true positive results among all positive predictions and all actual positives [17].

Latency is another crucial metric for LLMs. Latency measures the time taken for an LLM to generate a response to a prompt and is particularly important for use cases involving real-time interactions. This includes our application, where response times can affect the user experience.

Trustworthiness of LLMs encompasses many perspectives such as toxicity, stereotypes, and privacy. To measure the trustworthiness of LLMs, metrics have been created, such as those found in the Hugging Face LLM Safety Leaderboard [18]. Consideration of these measurements is crucial for applications in sensitive domains, such as the legal domain. These metrics all warrant consideration in any project using LLMs and are relevant to this project and its considerations over which LLM to use.

E. Chatbot Response Time

The paper "Opposing Effects of Response Time in Human–Chatbot Interaction" [19] evaluates the effects of chatbot response times on users. By evaluating how experienced and inexperienced chatbot users react to varying response times they were able to gauge which response times are most effective at maximising user experience. The paper concludes that lower response times have positive effects on users with large chatbot experience but can have negative effects on inexperienced users. However, too long of a response time was also suggested to have a negative impact. Experienced users reported frustration with longer response times, but no loss of trust whereas inexperienced users reported loss of trust with shorter response times. A response time of 2.3 seconds was identified to not lose any trust in inexperienced users whilst minimising frustration among experienced users.

III. Tools and Methodology

A. AutoGen

AutoGen is a library developed and open-sourced by Microsoft that provides a multi-agent conversation framework allowing users to build LLM workflows [20]. AutoGen agents are customisable, conversable, and can operate in various modes that use combinations of LLMs, human inputs, and tools [21].

By integrating dynamic retrieval mechanisms, AutoGen enhances LLMs' ability to provide accurate and contextually relevant responses. Specifically, AutoGen's architecture includes a Retrieval-augmented Assistant, which works in tandem with other agents to retrieve relevant documents from a vector database using embedding models such as Sentence transformers. This retrieval process is integrated into the generative process of the LLM, improving the factual accuracy and relevance of the output. This is a direct implementation of RAG, which this project aims to utilise. The Retrieval-augmented Assistant will allow for any resources regarding New Zealand consumer law to be included in the database. This will increase efficiency in development, as well as create a database that can be updated if there are any amendments to the law.

Dense passage retrieval (DPR) is the leading method used for retrieval in Q&A tasks, responsible for retrieving the most relevant passages [15]. AutoGen's interactive retrieval uses an iterative approach to prompt additional retrieval attempts, increasing the chances of finding relevant information. This leads to up to 10% increased accuracy and reliability over DPR when evaluated with the F1 performance metric [21], [17].

Another benefit of the AutoGen framework is the direct support for OpenAI models through API access. Other settings such as tokenizers can also be specified and accessed through API's. Hugging Face models are also accessible through AutoGen; however, they are required to be run locally through software such as LM Studio [20]. The support for multiple models is beneficial for optimisation purposes as it allows for comparisons of a vast range of models.

B. Hugging Face

Hugging Face is a platform that provides access to models and libraries, with a notable library being Transformers [22]. This has the potential to be used as an alternative to the AutoGen framework. The Hugging face transformer library provides support for multiple models such as BERT, GPT, and LLaMa, allowing for testing across different architectures. However, a notable challenge is that there are unique implementations for different models, adding complexity to the development process.

Additionally, Hugging Face's library provides support for RAG techniques, however, this implementation differs from that of AutoGen [23]. Where AutoGen provides a specialised

Retrieval-Augmented Assistant, Hugging Face implements RAG within its broader library network, providing less support for development [21]. Both the Hugging Face and AutoGen platforms support the same models, as both have access to all models stored within Hugging Face, such as LLaMa and BERT.

C. Fast API Framework

Fast API is a web framework that uses asynchronous programs with Python's Asyncio library. The framework is one of the fastest Python web frameworks available and experiences little performance decrease as applications are scaled. The framework is also modern with support for recent Python versions that the project intends to use. This framework would be able to be used in conjunction with both AutoGen and Hugging Face to deploy the web portion of the application.

D. Flask Framework

Flask is a lightweight and flexible Python web framework. Unlike asynchronous frameworks like FastAPI, Flask operates in a synchronous manner, which can reduce complexity for developing small to medium-sized applications. Similarly to Fast API Flask has support for recent Python versions. This framework could also be used in conjunction with both AutoGen and Hugging Face to deploy the web portion of the application.

E. Methodology

Agile was chosen as the methodology for this project as its flexibility and iterative nature have allowed for continuous improvement based on discoveries and feedback. The recent emergence of the field and the developing nature of LLMs meant that the initial scope of the project was identified to likely be modified during development. Breaking the project into manageable sprints through GitLab issues and milestones with clear goals has ensured focused and organised progress. Weekly check-ins outlining objectives and reviewing progress, with obstacles and priorities identified, have been used to manage the project life cycle.

IV. Design

The system is composed of five core modules: the user interface, Application, Chatbot, Model, and Database. Each module serves a distinct function but works together to process user inputs, generate responses, and then return them through the user interface. The architecture is user-centric, ensuring that the user only interacts with the system through the designated interface. This encapsulation is done to increase usability and performance by preventing unintended interactions. The architecture diagram in Figure 1 provides an overview of how these components are connected and interact with one another.

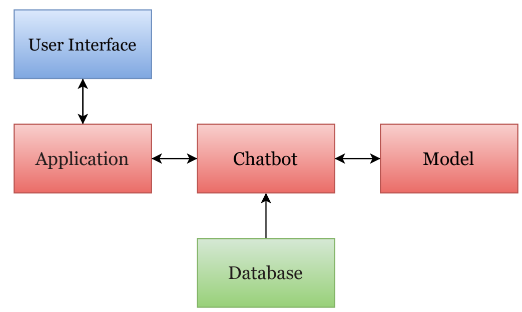


Fig. 1. Diagram of System architecture Components

A. User Interface

The user interface (UI) serves as the only point of interaction between the user and the system, providing a web-based platform that accepts user inputs and displays responses. The primary requirement of the UI is to be simple and clear to prevent confusion or input errors. The conversational chatbot is required to interact with users through a textbox for entering queries with an area to display responses generated by the system. These elements are organised in a way that relates the user's input to the system's output. Buttons are also required to allow access to the functionality for the action and complaint generators (shown in Figure 2).

To enhance user experience and confidence, real-time visual feedback (e.g. loading indicators) is required, to provide transparency about the system's processing status.

The primary requirement of the interface is to accept and display user inputs and application responses. To do this the user interface requires a text box to allow users to type out their input. The text entered and the result of the query is also required to be displayed to the user in a clear concise manner. For consistency purposes, the user's query and answer are designed to be related in their display so that the user can see what they have typed to help understand the output. To increase clarity the user interface is required to be simplistic and straightforward. Users should have minimal options available to decrease the potential for mistakes.

B. Chatbot

This project has chosen to use a conversational chatbot to provide legal aid. This consists of prompt engineering and RAG in between the application and an LLM to form the chatbot component. A chatbot is an appropriate solution as it can maintain constant availability without the downtime introduced by human legal aid. Scalability is also provided so that the application can handle multiple queries simultaneously whilst maintaining a short response time. Most importantly the conversational ability allows users to iteratively communicate with the system until they receive a comprehensive response that takes into account their specific situation. This targeted aid aligns with the overall goal of the project and makes a chatbot suitable for this application.

Support is further provided in the form of specific functions for "Action Generation" and "Complaint Generation". Action

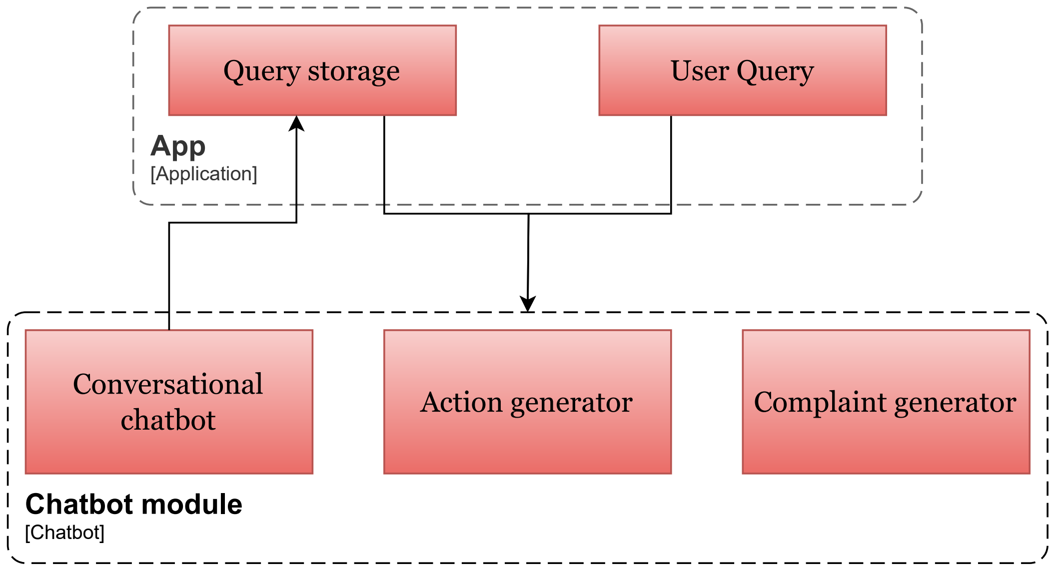


Fig. 2. Chatbot and Application Interaction

generation is an option available to users once they are satisfied with their conversation with the chatbot. The function generates a list of actions that the user can take based on the situation they have discussed with the chatbot. Complaint generation also uses the conversation as background knowledge to instead write out a formal complaint for the user. Together these two functions serve to further provide legal aid.

These functions all require calls to the systems model (shown in Figure 1) and prompt engineering to better the response. By using prompt engineering is it possible to improve on the generative capabilities of LLMs and control outputs. Prompt engineering works by attaching additional text to queries before parsing it to an LLM to give the model a greater understanding of the question being asked. In this context, prompt engineering reduces the margin of error for users as less comprehensive user prompts are required. This is particularly useful as the target audience for the application is regular citizens who are likely inexperienced with prompting language models.

C. Model

The model component of the system utilises an existing model as training a new LLM would require significant resources. The 2023 Artificial Intelligence Report by Stanford University estimated that training OpenAI's GPT-3 cost roughly $1.8 million [24]. This is infeasible for this project, especially when combined with the costs required to host such a model through cloud infrastructure. The report also found that the training of GPT-3 produced 502 tonnes of CO_2: a cost that does not align with this project's goal of minimising environmental impact.

Using existing LLMs also introduces more potential for applications to new languages. Ongoing development of LLMs proficient in languages other than English can introduce the possibility of extending this application to support multiple languages, such as Aotearoa's other written language: Te Reo Māori. An application with support for various languages would serve to further increase accessibility to legal aid, fulfilling the project goals. Price is also a factor with the project being posed as a cost-effective solution. Training a new model would require the project to be used significantly for individual queries to average out to be less than the goal set by the project.

LLMs can be accessed locally, through downloading a model, or remotely, through API access. Local access provides a slight reduction in latency as it eliminates the latency introduced by network calls. However, deploying locally requires significant memory, which can restrict the size and performance of the models that can be run. As a result, accessing LLMs through APIs allows for the use of bigger and better models without worrying about memory constraints. This does come with increased latency, and an introduced cost of accessing the models, however, the performance benefit is necessary for the system, leading this project to use LLMs through API access.

D. Application

The application component of the system is responsible for communication between the user interface and the backend of the system. Data is processed and rendered through this component with speed being crucial. To improve the user experience responses are required to be timely as unnecessary delays in the interface will contribute towards overall query duration, decreasing the likeliness of fulfilling the project's goal of 1.8 to 2.8 seconds per response.

The application further serves as a temporary data storage for the chatbot component as is outlined in Figure 2. The user's query and the chatbot's response are both stored by the application in the query storage to then be used for the two generator systems.

E. Database

To create answers that are relevant to New Zealand consumer law the system must use and reference relevant legal information, gathered in the database. The structure of the database is designed to facilitate efficient querying, allowing information to be retrieved quickly. A key concern is the handling of user data as queries may include personal or sensitive information. This introduces the constraint of not being able to store any user data. While anonymising data could be considered, the associated risks remain too high and could introduce additional variability. Therefore, all user queries will be processed without being stored or logged into the database, ensuring compliance with privacy requirements.

V. Implementation

A. Model Selection

The implementation of the application first started by evaluating and deciding which LLM will be used by the chatbot (as shown in Figure 1). Various methodologies exist for evaluating the performance of LLMs with the main ones being response speed, correctness, and trustworthiness. Within the scope of this project, trustworthiness in particular has been identified as extremely important. There are concerns regarding misinformation generated by LLMs, potentially leading to severe problems when used in a legal context [12]. One valuable source that explores LLM trustworthiness is the LLM Safety Leaderboard uploaded to Hugging Face [18]. This leaderboard looks at multiple metrics such as non-toxicity and non-stereotype and produces an average across all metrics. The top-performing models include Anthropic's Claude-2.0, Vertexai's Gemini-Pro-1.0, Meta's Llama-2-7b-Chat-Hf, and OpenAI's GPT-3.5-Turbo. Notably, despite its advanced reasoning and formatting capabilities, GPT-4 performed on average 3.21% worse on safety metrics such as non-toxicity and non-stereotype when compared to its predecessor GPT-3.5-Turbo.

The pivotal consideration with model selection lies in balancing trustworthiness whilst attempting to minimise LLM latency. Table 1 shows a comparison of the top four models from a study on the lowest latency (time to first chunk (TTFC)) [25]. As GPT-3.5 Turbo was shown to be one of the top four models in terms of both safety and TTFC it was selected to be used in this project. It was the slowest of the four by 0.14 seconds, however, this time is negligible, especially when factoring in the additional, variable latency of API calls.

During this project's development, OpenAI released a new model called GPT-4o. This model was shown by OpenAI to have the lowest latency and highest accuracy of any of their existing models [26]. It is also discussed to have built-in safety interventions to improve the fairness and trustworthiness of the model over previous OpenAI models such as GPT-3.5-Turbo and GPT-4. This should make the model more suited to the project than the previously best-identified model: GPT-3.5-Turbo. However, the recency of the model has resulted in the external metrics and leaderboards used to evaluate other models not being applied to GPT-4o yet. As a result, GPT-3.5 Turbo and GPT-4o were both evaluated to determine which performs better in the context of the project.

TABLE I
EVALUATION OF MODEL LATENCY (TIME TO FIRST CHUNK - TTFC)

Model	Latency (TTFC in seconds)
Mistral 7B	0.23
Mixtral 8B	0.24 - 0.29
Llama 3	0.24 - 0.29
GPT-3.5 Turbo	0.37

The evaluation of these models also served as an evaluation of the project, and the method and results are discussed in the evaluation section of this report. The results from the evaluation (displayed in Table 2) show that the two models produced similar performance. According to the accuracy metrics GPT-4o had a 10% higher success rate for both accuracy and completeness. As a result, GPT-4o was chosen as the final model to be used for the project as this test determines it to be more accurate, as well as more responsive and trustworthy.

Due to the time GPT-4o was released, the development of the project was completed with GPT-3.5-Turbo, with GPT-4o only being used for the final evaluation and system. Additionally, the implementation of the model was done by specifying the model version and API access key. By specifying the exact version of the model we can prevent configuration drift and prioritise the consistency of the chatbot.

B. Chatbot

When deciding the framework to be used to develop the chatbot, the LLM was first taken into consideration. GPT-3.5-Turbo is solely accessible through API calls as the model requires too much memory to be deployed locally. As both the HuggingFace and Autogen frameworks can provide access to the LLM, each system was considered for development. Both support RAG, however, Autogen's version of RAG was shown to be up to 10% more accurate than the method used by HuggingFace, leading the project to use Autogen to implement the chatbot system.

The iterative chatbot works by using two "Agents" which are initialised with prompts utilising prompt engineering. When a user asks a query in the UI it is passed to the Process User Query module in the application. The response is then processed by the chatbot, with the following process, displayed in Figure 3: The initial Retrieval-augmented User Proxy agent uses a form of RAG called Dense Passage Retrieval to extract the information from the database which appears to be relevant to the query. This information is prepended to the user query before being given to the second agent called the Retrieval-augmented User Assistant. This agent makes an API call to the GPT-4o model with the query. The response from the LLM is then compared to the original prompt by the Retrieval-augmented User Assistant, by using another call to the LLM to ask whether the response answers the original question. If it is deemed to not comprehensively answer the user's prompt then more context is requested from the original agent. Once the answer is deemed to answer the user's prompt, it is returned to the application module to be displayed to the user.

To mitigate hallucinations the agents also request the model to cite the source of the most relevant resource from the database to the query. This allows for transparency as it shows which of the legal aid resources are being used in the answer. It also provides users with the location of the most relevant primary resources they can access to further understand their original consumer law query.

This chatbot forms the primary aid available to the user. The conversational feature of the chatbot is implemented by allowing follow-up questions once a response is generated. This allows users to iteratively converse until they are satisfied that their question is answered comprehensively. This is performed by using short-term memory containing records of all questions and responses in the current conversation which is prepended to the user query before being prompted to the response generation system. This database is cleared every session to remove the persistence of personal user data, which this project identified as a safety risk. This conversational ability mimics a human conversation, serving to increase the usability of the system. As the accuracy and comprehensiveness of results to an individual's situation depend somewhat on the detail included by the user, this conversational ability promotes comprehensive answers, contributing towards fulfilling the project goals.

In order to provide more aid, the follow-up functions (displayed in Figure 2) have been created to assist users in their next responses. The first function "Generate Action" is defined within the Action Generator module and the second "Generate Complaint" is defined within the Complaint Generator module.

After understanding their situation, the next step for a citizen is to identify what actions they can take to resolve the situation. The "Generate Action" function assists this by generating the specific actions users can take next. Furthermore, writing a letter of complaint is commonly the first legal action citizens can take when they have a legitimate claim. The system streamlines this process by providing a function that incorporates relevant legal details and user-specific information into the template, ensuring the letter is accurate to the user's situation. An example of generated results can be seen in Figure 4.

Initially, these functions attempted to use the existing conversational chatbot system, outlined in Figure 3, by prompting it to generate a list of actions the users can take given their previous context. It did this as a single prompt, rather than using the conversational functionality of the chatbot. However, this resulted in the RAG system being unnecessarily used again. As the prompt already contained data obtained through RAG this additional process was multiplying the data which seemed to decrease performance.

In response to this, a slightly new system was developed specifically for the Action Generator and Complaint Generator modules. This system uses the final response generated by the Retrieval-augmented User Proxy as context. As this response contains all of the pertinent information extracted from both the user query and the database the prompt is more concise. This also reduces the time taken to generate the answer by preventing unnecessary re-querying of the database. This context is then included in the prompts used for both "Generate Action" and "Generate Complaint". For the later purposes of testing the system, a custom function was developed where the system was prompted to answer the user's question, using this context. This was used to evaluate whether this system performs better than the original chatbot system.

C. User Interface

The application is displayed as a website where users are greeted with the chatbot web page. The page prominently features a text box for users to input their queries, along with a dynamic display that shows a running record of both user prompts and system responses. The function for generating actions is implemented on the same web page so that users can generate the action from their current situation. This function then routes users to a separate web page which displays: The user's conversation with the chatbot, the newly generated action, and a function to generate a complaint. The generate complaint function then also calls the backend function before displaying the response on the current page. An example of the web page displaying an example user question, the answer generated, actions generated, and complaint generated, is displayed in Figure 4.

JavaScript was initially considered for rendering the front end of the application using frameworks such as Node.js or React. However, given the project's limited need for a complex user interface, the focus shifted toward clarity and speed of development. FastAPI and Flask, both Python web frameworks, were then compared. FastAPI was ultimately chosen due to its ease of use and the minimal learning curve required, especially since the performance differences between the two frameworks are negligible for an application of this size.

As a result, the front end of the application is rendered using HTML templates, as they provide a more straightforward way to structure the interface. This approach minimises the complexity of the development process while ensuring that the necessary functionality is delivered efficiently. HTML templates allow for faster iteration, easier integration with backend logic, and reduced overhead compared to heavier JavaScript frameworks, making them ideal for this project's requirements.

These templates are integrated with the FastAPI framework, which handles requests triggered by user actions. When a user submits a query, the system processes the request asynchronously, leveraging FastAPI's ability to handle real-time interactions with minimal delay. To enhance the user experience, a loading symbol is displayed while the system processes the query, providing visual feedback and reassuring the user that their request is being handled. For additional functionality, such as generating legal actions or writing formal letters of complaint, separate web pages are used to reduce clutter and improve clarity.

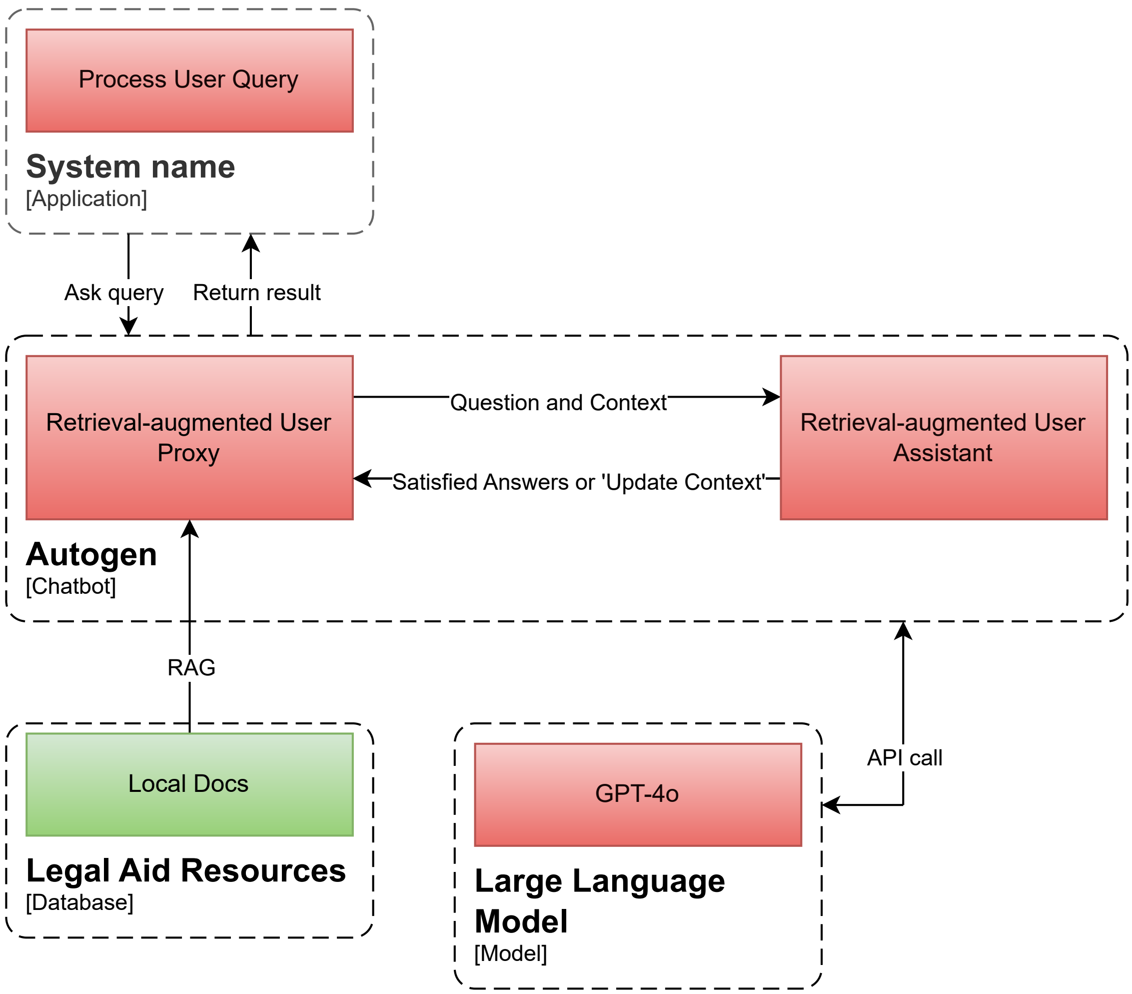


Fig. 3. Conversational Chatbot Diagram.

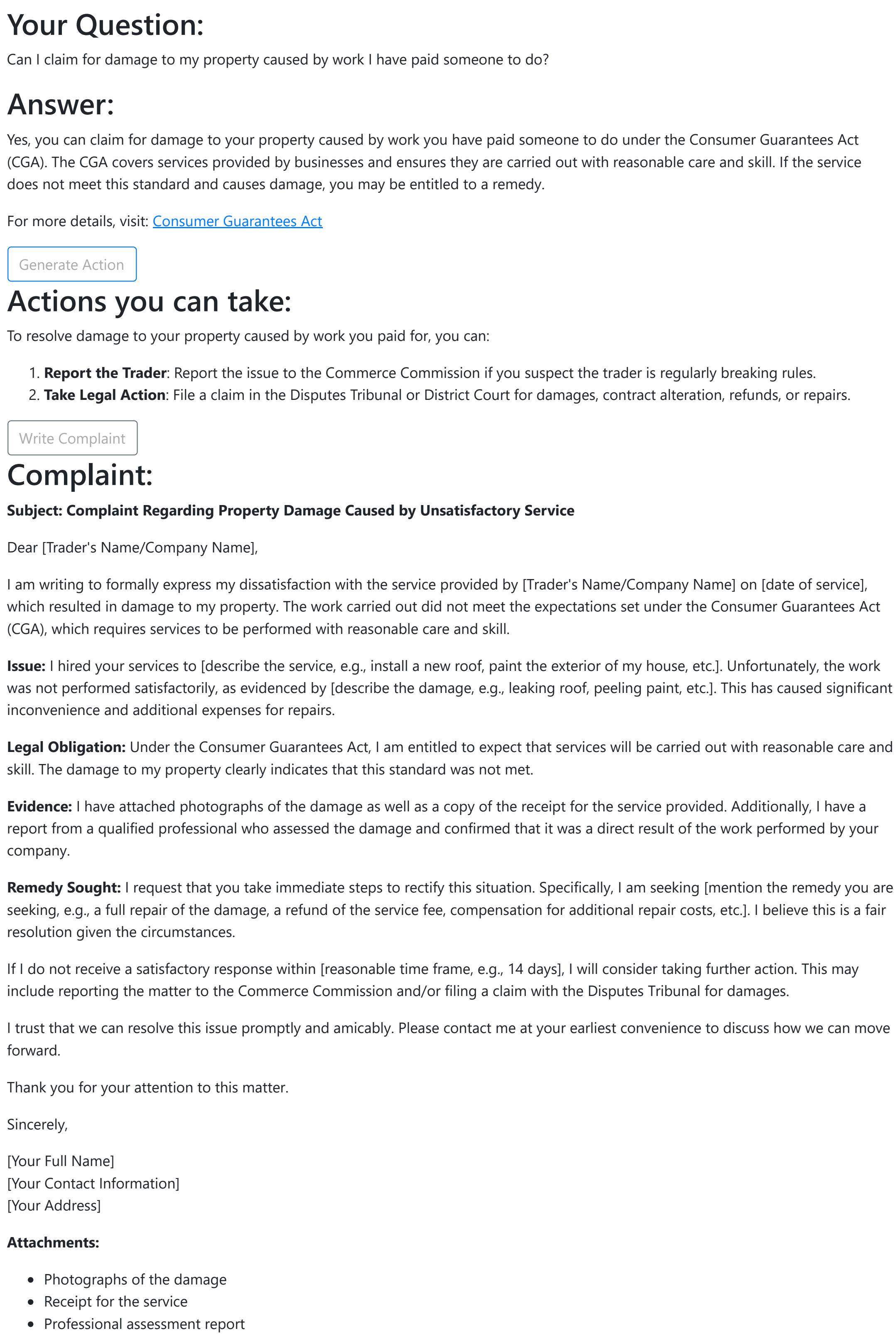
D. Database

The database is used by the RAG to be included in the prompt for the LLM to provide context and improve the accuracy of responses. It consists of pages from the Consumer Guarantees Act (CGA) [27] and the Commerce Commission [28]. The CGA is the primary database for all New Zealand consumer law outlining all New Zealand consumer law acts such as the Consumer Guarantees Act and the Fair Trading Act. It was selected for the project due to its comprehensive coverage of consumer law. The Commerce Commission's website is additionally used because it contains more in-depth interpretations of consumer law acts with details on specific scenarios. A combination of these two sources ensures that the application's responses are accurate to New Zealand's law standards. Community Law [29] was also considered as another resource, but was discarded due to overlap with the Commerce Commission rendering it unnecessary for the project.

Web pages from the websites are downloaded in HTML format to be used by the RAG system. The alternative to this is storing links to the web pages and then viewing the information through an API call when initialising the database. The storage of local copies of web pages is preferable to retaining external links for two primary reasons. Retrieving web pages via API calls to external sources introduces unnecessary latency, which can decrease the program's responsiveness. Also, relying on external links carries the inherent risk of encountering dead links over time, potentially compromising the long-term reliability and accessibility of the content. The disadvantage of the method is that in the event of a legislative amendment, these pages would require manual updating.

E. Embedding Function

Embedding functions are used to transform the user query into a representation that is readable by the LLM. The "text-embedding-3-large" model by OpenAI was selected due to its increased performance over alternative embedding functions. This function was tested to be at least 2.3% more accurate than all alternative OpenAI functions [30]. As the price difference is negligible, this performance is the only priority.



Your Question:

Can I claim for damage to my property caused by work I have paid someone to do?

Answer:

Yes, you can claim for damage to your property caused by work you have paid someone to do under the Consumer Guarantees Act (CGA). The CGA covers services provided by businesses and ensures they are carried out with reasonable care and skill. If the service does not meet this standard and causes damage, you may be entitled to a remedy.

For more details, visit: Consumer Guarantees Act

Generate Action

Actions you can take:

To resolve damage to your property caused by work you paid for, you can:

1. **Report the Trader**: Report the issue to the Commerce Commission if you suspect the trader is regularly breaking rules.
2. **Take Legal Action**: File a claim in the Disputes Tribunal or District Court for damages, contract alteration, refunds, or repairs.

Write Complaint

Complaint:

Subject: Complaint Regarding Property Damage Caused by Unsatisfactory Service

Dear [Trader's Name/Company Name],

I am writing to formally express my dissatisfaction with the service provided by [Trader's Name/Company Name] on [date of service], which resulted in damage to my property. The work carried out did not meet the expectations set under the Consumer Guarantees Act (CGA), which requires services to be performed with reasonable care and skill.

Issue: I hired your services to [describe the service, e.g., install a new roof, paint the exterior of my house, etc.]. Unfortunately, the work was not performed satisfactorily, as evidenced by [describe the damage, e.g., leaking roof, peeling paint, etc.]. This has caused significant inconvenience and additional expenses for repairs.

Legal Obligation: Under the Consumer Guarantees Act, I am entitled to expect that services will be carried out with reasonable care and skill. The damage to my property clearly indicates that this standard was not met.

Evidence: I have attached photographs of the damage as well as a copy of the receipt for the service provided. Additionally, I have a report from a qualified professional who assessed the damage and confirmed that it was a direct result of the work performed by your company.

Remedy Sought: I request that you take immediate steps to rectify this situation. Specifically, I am seeking [mention the remedy you are seeking, e.g., a full repair of the damage, a refund of the service fee, compensation for additional repair costs, etc.]. I believe this is a fair resolution given the circumstances.

If I do not receive a satisfactory response within [reasonable time frame, e.g., 14 days], I will consider taking further action. This may include reporting the matter to the Commerce Commission and/or filing a claim with the Disputes Tribunal for damages.

I trust that we can resolve this issue promptly and amicably. Please contact me at your earliest convenience to discuss how we can move forward.

Thank you for your attention to this matter.

Sincerely,

[Your Full Name]
[Your Contact Information]
[Your Address]

Attachments:

- Photographs of the damage
- Receipt for the service
- Professional assessment report

Fig. 4. Example Response Displaying Example Question with Chatbot Answer, Actions Taken, and Complaint Generated.

TABLE II
COMPARISON OF SYSTEMS BASED ON ACCURACY RATINGS

Model	**Completely Incorrect**	**More Incorrect than Correct**	**Approx. Equal Correct and Incorrect**	**More Correct than Incorrect**	**Nearly All Correct**	**Completely Correct**
GPT-4o		1	2	2	5	
GPT-3.5-Turbo		2	1	3	2	2
Generator			1	3	3	3

VI. EVALUATION

To determine whether the project meets its initial goals, four separate forms of testing have been performed to test the accuracy, completeness, responsiveness, and cost of responses.

To assess the system's accuracy and completeness, sample responses have been evaluated by lawyer Matt Farrington, whose legal expertise qualifies him as a domain expert. The chatbot was prompted with queries related to New Zealand consumer law, and its responses were recorded for evaluation. Each response was then assessed using Likert scales, which measured both the accuracy and completeness of the information provided.

The scales used are based on the study "Accuracy and Reliability of Chatbot Responses to Physician Questions" [19]. This paper attempts the evaluation of a similar system to that created in this project, allowing the scales used by the study to be appropriately applied to assessing responses to legal questions.

To measure the accuracy of responses, a 6-point Likert scale was used (with 1 indicating completely incorrect; 2, more incorrect than correct; 3, approximately equal correct and incorrect; 4, more correct than incorrect; 5, nearly all correct; and 6, completely correct). To measure completeness a 3-point Likert scale was used (with 1 indicating incomplete [addresses some aspects of the question, but significant parts are missing or incomplete]; 2, adequate [addresses all aspects of the question and provides the minimum amount of information required to be considered complete]; and 3, comprehensive [addresses all aspects of the question and provides additional information or context beyond what was expected]). The accuracy Likert scale used more points because accuracy was deemed more nuanced than completeness.

A minimum accuracy threshold of "nearly all correct" was established as the benchmark for determining whether the system is reliable enough for production use. Given the potential harm caused by incorrect legal advice, any inaccuracies are unacceptable, making a strict threshold necessary rather than relying on an average. "Nearly all correct" was chosen over "completely correct" to allow for minor, non-critical errors that do not affect the overall quality of the advice provided. Similarly, the completeness testing requires a minimum threshold of "adequate." While "comprehensive" answers are ideal, "adequate" responses still offer sufficient guidance to be useful for citizens seeking legal aid. In contrast, "incomplete" answers risk misleading users and are therefore unacceptable.

The queries the chatbot was prompted with have been obtained from the Citizens Advice Bureau (CAB) [3]. The CAB contains previous consumer rights questions asked by New Zealanders. As this project is being posed as an alternative to the CAB, the questions asked by users are likely similar, deeming these questions to be appropriate to be used to evaluate the chatbot.

For comprehensive results of the system, ten different queries were each evaluated on responses from three separate versions of the system. The first two versions both use the chatbot system outlined in Figure 3, whereas the third system tested is a modified version of the system for action and complaint generation, discussed earlier in this report, at the end of the implementation section. Version 1 uses GPT4-o in its API model call and version 2 uses GPT-3.5-Turbo.

A. Accuracy

When comparing these two versions we observe similarities in the accuracy of the results. By assigning a score from 1 to 6 for each response's accuracy and averaging the results, both versions yielded an identical score of 4.1. This similarity indicates that the chatbot is resilient to changes in the underlying model, suggesting it relies more on the system's architecture than the specific model used. The test shows that when using GPT-4o only 50% of responses met the required threshold of "nearly all correct" and when using GPT-3.5-Turbo only 40% of responses met the threshold, as seen in Table 2. While this demonstrates that the chatbot can produce accurate responses, it does so in only half of the cases.

There was also a lot of variation in response accuracy, suggesting that the system's performance may be influenced by factors beyond the model itself, such as the complexity of the legal queries, the phrasing of user inputs, or the chatbot's ability to interpret nuanced legal language. Interestingly, GPT-4o has a tighter distribution of responses than GPT-3.5-Turbo, suggesting GPT-4o is more consistent in its responses.

The Generator system, seen in Figure 2, is the third system tested and concentrates on producing actions and complaints. This model performed best in terms of accuracy, with an average score of 4.8. Furthermore, 60% of its responses met the minimum accuracy threshold, and none fell below the "approximately equal correct and incorrect" level. This indicates better performance than the first two systems, suggesting that the algorithm is more effective.

B. Completeness

Completeness testing is displayed in Table 3 and yielded results similar to those from the accuracy tests, with both of the first two systems receiving an average completeness score of 1.7 when assigning values from 1-3 for each completeness value. This similarity further corroborates the idea that the chatbot is resistant to changes in the underlying model. The test results show that, when using GPT-4, 50% of the responses met the project's "adequate" completeness threshold, while GPT-3.5-Turbo achieved this in only 40% of responses. As with accuracy, GPT-4 performed slightly better, but both models still fell short of meeting the desired standard.

The third system again demonstrated better performance than the first two with 80% of responses meeting the required completeness threshold. Notably, 50% of the responses were rated as "comprehensive." This indicates that the generator systems demonstrate improved performance over the chatbot system in both accuracy and completeness.

TABLE III
COMPARISON OF SYSTEMS BASED ON COMPLETENESS RATINGS

Model	Incomplete	Adequate	Comprehensive
GPT-4o	5	3	2
GPT-3.5-Turbo	6	1	3
Generator	2	3	5

C. Responsiveness

Based on the research outlined in the related work [19], an optimal response time of 2.3 seconds was identified as appropriate. Higher response times resulted in increased frustration and decreased user experience. Lower response times had a negative effect on inexperienced users as they tended to trust the response less. Consequently, this project has a target median response time of 2.3 with an allowable deviation of 0.5 seconds, establishing a target range between 1.8 and 2.8 seconds. As there is no concrete standard established for ideal chatbot response time, the choice of 2.3 seconds was informed by the fact that inexperienced users responded favorably to this response time. As a result, maintaining a median close to this value should sustain user trust across all experience levels.

The results of responsiveness testing on the chatbot are illustrated in the boxplot in Figure 5. Testing yielded a median response time of 2.1 seconds which falls within the intended target range. This indicates the successful achievement of the project's target time taken per response. Furthermore, the lower quartile was 1.9 seconds and the upper quartile was 2.2 seconds. The narrow interquartile range of 0.3 seconds indicates that the response times are closely clustered, reflecting high consistency. The minimum response time of 1.7 seconds is just outside the lower bound of the target range, while the maximum response time, recorded at 6.97 seconds, is an outlier. This outlier is attributed to the iterative nature of the chatbot's response system (shown in Figure 3), where additional context is provided if the initial response is insufficient, leading to increased response times, as seen in the 6.97-second instance.

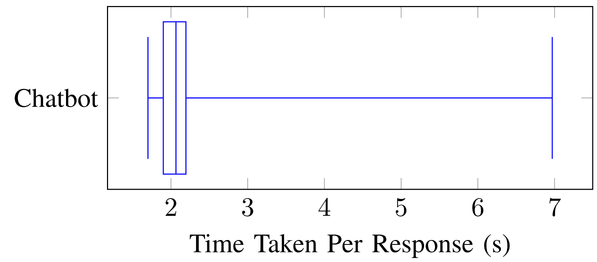


Fig. 5. Box and Whiskers Plot of Time Taken per Response for Chatbot

D. Cost

Due to the application's access to models through APIs, a reoccurring cost is present. Each query requires access to OpenAI's LLM GPT-4o as well as their embedding model text-embedding-3-large. To ensure that each query costs less than $0.10 per query, 100 separate queries were performed with the total cost observed. The total cost was measured at $0.95, with $0.70 allocated for the LLM and $0.25 for the embedding model. This results in an average of $0.095 per query, which is well within the project's budgetary goal. By keeping per-query costs low, the project ensures that it remains financially viable in the long term, even with higher usage.

E. Limitations

The accuracy and completeness tests indicate that the chatbot was unsuccessful in meeting the desired benchmarks for both metrics. These results highlight limitations not only in the chatbot's performance but also in the scope and design of the evaluation itself. The tests used responses where the chatbot was evaluated based on single, isolated answers to individual queries. This approach did not capture the full conversational capabilities of the system, as it lacks the dynamic back-and-forth interaction typical in real-world use.

Additionally, cost considerations present another limitation. As the chatbot relies on API access to models, throttling may be required to manage expenses, especially if user interactions involve multiple exchanges per query. Allowing unlimited questions by users could incur significant costs if exploited maliciously. Any real-world deployment would require access limits for users to prevent excessive costs.

Another important area for future evaluation would be user testing for comprehension. The responses used in the tests were derived from a controlled, "professionalised" set of queries. This means the evaluation did not account for the variability in user input that would naturally occur in real-world scenarios. Additionally, understanding how well users grasp the chatbot's responses could further help in assessing its overall effectiveness at providing legal aid.

VII. CONCLUSION

The objective of this project was to enhance legal aid in Aotearoa using LLMs. The solution has fulfilled the purpose of the project by developing an application, featuring a chatbot, to provide legal aid relevant to a user's situation. The final solution successfully meets the goals for both average cost per query and average time taken per response. However, the system only achieved a maximum response accuracy of 50% and a maximum response completeness of 80%. These both fall short of the desired accuracy of the project.

A limitation of the project's accuracy and responsiveness goals is the high standard expected of AI when compared to humans. The testing performed permits zero tolerance for undesirable answers. However, human lawyers are not 100% accurate and can have differing opinions on a matter. The current lack of explainability in LLMs makes errors harder to justify and accept than human error. While this problem can currently be solved with the band-aid solution of a disclaimer stating "This advice may be incorrect", or a similar alternative, a more comprehensive approach is needed. Extensive research into the explainability of LLMs and the process taken for producing answers can potentially foster more trustworthiness for AI and increased tolerance for errors.

This project serves as a proof of concept for a legal aid solution utilising LLMs. Future work could potentially broaden the range of legal topics beyond consumer law and into other small claims matters. Multi-lingual support to include responses in Te Reo Māori is another future work prospect that could benefit citizens of Aotearoa if executed with correct respect to Te Reo Māori.

The constant advancements in AI and software development make it more of a matter of when, rather than if, there is an accessible source of legal aid, using Large Language Models, available to Aotearoa citizens.

Acknowledgments

I would like to thank my supervisor Dr Andrew Lensen for making this project possible and providing your continuous support. I would also like to thank Matt Farrington for the research support and legal guidance and Damien O'Neill for your technical advice. Finally, I would like to thank the ENGR489 cohort for their support throughout this journey.

References

[1] *Access to justice, diversity and the legal profession - nz bar association - ngā ahorangi motuhake o te ture*, www.nzbar.org.nz. [Online]. Available: https://www.nzbar.org.nz/access-justice-diversity-and-legal-profession (visited on 05/30/2024).

[2] J. Prebble, *How much does a nz business lawyer cost?* LegalVision New Zealand, Dec. 2020. [Online]. Available: https://legalvision.co.nz/disputes-and-litigation/how-much-does-a-business-lawyer-cost-in-new-zealand/.

[3] *Citizens advice bureau*, Cab.org.nz, 2019. [Online]. Available: https://www.cab.org.nz/.

[4] M. Malec, *Large language models: Capabilities, advancements, and limitations [2024]*, HatchWorks, Mar. 2024. [Online]. Available: https://hatchworks.com/blog/gen-ai/large-language-models-guide/ (visited on 05/29/2024).

[5] R. Carter, *The best large language models in 2023: Top llms*, UC Today, Nov. 2023. [Online]. Available: https://www.uctoday.com/unified-communications/the-best-large-language-models-in-2023-top-llms/ (visited on 05/30/2024).

[6] U. Nations, *The 17 sustainable development goals*, United Nations, 2024. [Online]. Available: https://sdgs.un.org/goals (visited on 05/28/2024).

[7] N. Goodson and R. Lu, *Intention and context elicitation with large language models in the legal aid intake process*, arXiv.org, Nov. 2023. DOI: 10.48550/arXiv.2311.13281. [Online]. Available: https://arxiv.org/abs/2311.13281 (visited on 05/20/2024).

[8] C. Xiao, X. Hu, Z. Liu, C. Tu, and M. Sun, "Lawformer: A pre-trained language model for chinese legal long documents," *AI Open*, vol. 2, pp. 79–84, 2021. DOI: 10.1016/j.aiopen.2021.06.003.

[9] I. O. Gallegos, R. A. Rossi, J. Barrow, *et al.*, *Bias and fairness in large language models: A survey*, Arxiv.org, 2024. [Online]. Available: https://arxiv.org/html/2309.00770v2.

[10] R. Zhang, H. Li, Y. Wu, *et al.*, *Evaluation ethics of llms in legal domain*, arxiv.org, Mar. 2024. [Online]. Available: https://arxiv.org/html/2403.11152v1 (visited on 05/27/2024).

[11] B. Wang, W. Chen, H. Pei, *et al.*, *Decodingtrust: A comprehensive assessment of trustworthiness in gpt models*, arXiv.org, Jun. 2023. DOI: 10.48550/arXiv.2306.11698. [Online]. Available: https://arxiv.org/abs//2306.11698.

[12] R. Maruf, *Lawyer apologizes for fake court citations from chatgpt — cnn business*, CNN, May 2023. [Online]. Available: https://edition.cnn.com/2023/05/27/business/chat-gpt-avianca-mata-lawyers/index.html (visited on 05/30/2024).

[13] G. F. Almeida, J. L. Nunes, N. Engelmann, A. Wiegmann, and M. de Araújo, "Exploring the psychology of llms' moral and legal reasoning," *Artificial intelligence*, vol. 333, May 2024. DOI: 10.1016/j.artint.2024.104145. (visited on 05/27/2024).

[14] *What is rag? - retrieval-augmented generation explained - aws*, Amazon Web Services, Inc. [Online]. Available: https://aws.amazon.com/what-is/retrieval-augmented-generation/ (visited on 05/30/2024).

[15] V. Karpukhin, B. Oguz, S. Min, *et al.*, *Dense passage retrieval for open-domain question answering*, Sep. 2020. [Online]. Available: https://arxiv.org/pdf/2004.04906.

[16] J. Hsia, A. Shaikh, Z. Wang, and G. Neubig, *Ragged: Towards informed design of retrieval augmented generation systems*, Mar. 2024. [Online]. Available: https://arxiv.org/pdf/2202.01110.

[17] V. Adlakha, P. BehnamGhader, X. H. Lu, N. Meade, and S. Reddy, *Evaluating correctness and faithfulness of instruction-following models for question answering*, arXiv.org, Jul. 2023. DOI: 10.48550/arXiv.2307.16877. [Online]. Available: https://arxiv.org/abs/2307.16877.

[18] *Llm safety leaderboard - a hugging face space by ai-secure*, huggingface.co. [Online]. Available: https://huggingface.co/spaces/AI-Secure/llm-trustworthy-leaderboard (visited on 05/24/2024).
[19] R. S. Goodman, J. R. Patrinely, C. A. Stone, *et al.*, "Accuracy and reliability of chatbot responses to physician questions," *JAMA network open*, vol. 6, e2336483–e2336483, Oct. 2023. DOI: 10.1001/jamanetworkopen.2023.36483.
[20] *Autogen*, Microsoft Research. [Online]. Available: https://www.microsoft.com/en-us/research/project/autogen/ (visited on 05/30/2024).
[21] Q. Wu, G. Bansal, J. Zhang, *et al.*, *Autogen: Enabling next-gen llm applications via multi-agent conversation*, arXiv.org, Oct. 2023. DOI: 10.48550/arXiv.2308.08155. [Online]. Available: https://arxiv.org/abs/2308.08155.
[22] T. Wolf, L. Debut, V. Sanh, *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *arXiv:1910.03771 [cs]*, Feb. 2020. [Online]. Available: https://arxiv.org/abs/1910.03771.
[23] *Retrieval augmented generation with huggingface transformers and ray*, huggingface.co. [Online]. Available: https://huggingface.co/blog/ray-rag (visited on 05/26/2024).
[24] S. University, *Artificial intelligence index report 2023 introduction to the ai index report 2023*, 2023. [Online]. Available: https://aiindex.stanford.edu/wp-content/uploads/2023/04/HAI_AI-Index-Report_2023.pdf.
[25] K. Talamadupula, *A guide to llm inference performance monitoring*, Symbl.ai, Mar. 2024. [Online]. Available: https://symbl.ai/developers/blog/a-guide-to-llm-inference-performance-monitoring (visited on 05/31/2024).
[26] *Hello gpt-4o*, Openai.com, 2024. [Online]. Available: https://openai.com/index/hello-gpt-4o/.
[27] *Consumer laws — consumer protection*, www.consumerprotection.govt.nz. [Online]. Available: https://www.consumerprotection.govt.nz/general-help/consumer-laws.
[28] C. C. N. Zealand, *Consumers*, comcom.govt.nz. [Online]. Available: https://comcom.govt.nz/consumers.
[29] *Law manual online*, Community Law. [Online]. Available: https://communitylaw.org.nz/law-manual-online/.
[30] *Openai api*, platform.openai.com. [Online]. Available: https://platform.openai.com/docs/guides/embeddings/what-are-embeddings (visited on 05/31/2024).