AI-powered Legal Virtual Assistant: Utilizing RAG-optimized LLM for Housing Dispute Resolution in Finland

Student: Md Irfan Rafat

Haaga-Helia University of Applied Sciences
MBA, Degree Prorgramme in Leading Business Transformation
Specialisation: Digital Business Opportunities
Master's Thesis
2024

Abstract

Author
Md Irfan Rafat

Degree
Master of Business Administration, Leading Business Transformation

Thesis title
AI-powered Legal Virtual Assistant: Utilizing RAG-optimized LLM for Housing Dispute Resolution in Finland

Number of pages and appendix pages
55+7

Driven by advancements in Artificial Intelligence (AI), a wave of transformation is sweeping across numerous industries, and the legal sector is well-positioned to capitalize on these developments. This thesis explores the feasibility of applying recent AI advancements, enhancing the performance of Large Language Models (LLMs) combined with the information retrieval capabilities of Retrieval-Augmented Generation (RAG), to resolve housing disputes in Finland.

The study provides an overview of chatbots or virtual assistants, associated technologies LLM and RAG, the opportunities chatbots offer for providing user-centric solutions, and the essential characteristics and challenges of applying chatbots in the legal domain. Furthermore, this study presents a case study on the development of an efficient system for legal information extraction using semantic Question and Answer (QnA) techniques applied to German case law documents. Additionally, this work sheds light on existing real-world chatbot solutions in the legal domain. As a result, this thesis delves into the current state, evolution, and future trajectories of the aforementioned technologies.

In this thesis, a prototype is developed using LLM technology optimized with RAG. Then, an experimental setup is designed to evaluate the performance of the RAG-optimized LLM against three non-optimized popular LLM-powered AI technologies, to assess the scope of RAG in improving LLM-powered chatbots. The experiment includes formulation of multiple prompts reflecting real-life housing dispute scenarios, including common user errors. The prototype has been developed on the MS Azure platform, integrating the LLM Azure OpenAI 3.5 turbo and Azure AI search for the RAG approach. The evaluation is based on testing the chatbots' comprehension and response generation capabilities.

The results from the experiment are evaluated by a human legal expert. The expert's analysis focuses on the accuracy and completeness of the responses generated by the models and the prototype. This evaluation helps identify the prototype’s RAG capabilities to retrieve information from the source documents as well as it identifies the challenges, limitations, and improvement criteria for adopting AI-powered virtual assistants in the legal field.

In conclusion, this thesis identifies the opportunities and unveils the gaps and intricacies in AI-powered chatbot’s capabilities to retrieve data from sources, to understand complex user scenarios, and to provide tailored responses aiming to provide meaningful guidance for users seeking solutions in the legal arena.

Keywords
Large Language Model, Retrieval Augmented Generation (RAG), Housing Disputes, Virtual Assistants, Generative AI, Generative Pre-training Transformer (GPT)

Table of contents

Abbreviations 1
1 Introduction 2
1.1 Objectives 3
1.2 Scope 3
1.3 Research Questions 4
2 Literature Review 5
2.1 An overview of chatbot and virtual assistant technologies 5
2.1.1 Defining chatbots, virtual assistants, and legal virtual assistants 5
2.1.2 Categories of chatbot 6
2.1.3 Potential of chatbots or virtual assistants in enhancing legal services 10
2.1.4 Challenges and limitations associated with adoptions of chatbots 12
2.2 Existing AI-powered chatbots in the legal arena around the world 12
2.2.1 Juro legal AI assistant 13
2.2.2 Harvey AI legal AI assistant 13
2.2.3 CaseMine legal AI assistant 14
2.3 HOAS and the Helmi Chatbot: AI-powered Housing Assistance 14
2.4 SmartRent in the United Kingdom: AI-powered Assistance for Tenants 16
2.5 Collaborative system in QnA in German case law documents 17
2.5.1 Human-AI collaboration approach 17
2.5.2 The experiment and the metrics 19
2.6 Housing disputes domain in Finland 21
2.7 LLM and RAG Overview 22
3 Research Design 24
3.1 Research Approach 24
3.2 Scrum Iterative Development Framework 25
3.3 Investigation Methods 25
3.3.1 Structure of experimental setup 26
3.3.2 Qualitative Feedback for evaluating the experiment results 27
4 Implementation of the Prototype 29
4.1 Development platform: Azure services 29
4.2 System model 30
4.2.1 RAG Based Implementation Through Azure AI Search 31
4.2.2 Source Data Gathering and Data Extraction 32
4.2.3 RAG features of the prototype 33
5 Experiment: Testing popular LLMs and testing the prototype 35

5.1 Overview of the AI powered chatbots chosen for evaluation 35
5.1.1 ChatGpt 3.5 Model 35
5.1.2 Gemini 1.0 36
5.1.3 Perplexity AI 37
5.2 Experiment intentions 38
5.3 Execution and results 38
5.3.1 Qualitative analysis from feedback 41
6 Discussion 44
6.1 Research Question 1 44
6.2 Research Question 2 44
6.3 Research Question 3 45
6.4 Recommendations, Future Suggestions and opinions 46
7 Conclusion 48
References 49
Appendices 56
Appendix 1.1: Application Technologies 56
Appendix 1.2: Feedback Form 59
Appendix 1.3: Experimental Setup 61

Abbreviations

AI	Artificial Intelligence
API	Application programming interface
DL	Deep Learning
FAQ	Frequently Asked Questions
GPT	Generative Pre-Trained Transformers
IT	Information Technologies
LLM	Large Language Model
LVA	Legal Virtual Assistant
ML	Machine Learning
NLP	Natural Language Processing
NLG	Natural Language Generation
NLU	Natural Language Understanding
POC	Proof of Concept
QnA	Questions and Answers
RAG	Retrieval Augmented Generation
VA	Virtual Assistant

1 Introduction

With the rapid advancements in Artificial Intelligence (AI) technologies, the way professional activities are conducted has been drastically changing, and the legal profession is no exception. Smart virtual assistants and other AI tools are being integrated into legal practice, reshaping the industry. AI-powered legal assistants are evolving to act as a legal brain, providing legal advice and assisting with research, thereby saving time and reducing costs. Technologies such as Generative AI, AI assistants, Natural Language Processing (NLP), Large Language Models (LLM), machine learning, and predictive analytics are contributing to the development and improvement of chatbots and virtual assistants in the legal industry (Aslam, 2023). This integration has enabled streamlined legal processes, improved efficiency, and transformed the role of legal professionals.

Legal housing disputes in Finland may arise from various issues, such as tenancy agreements, property rights, landlord-tenant conflicts, and conflicts with neighbors (Kettunen & Ruonavaara, 2015). Given the complexity of housing-related legal matters, an efficient and accessible virtual assistant can assist both legal professionals and individuals involved in such disputes.

This thesis aims to explore how Large Language Models (LLMs) optimized by Retrieval-Augmented Generation (RAG) contribute to the effectiveness of Legal Virtual Assistants in understanding and responding to complex legal queries from users seeking solutions to housing disputes in Finland.

The primary problems, needs, and development tasks that the thesis aims to solve include:

Enhancing accessibility and efficiency in housing dispute resolution:

Understanding the intricacies of legal information and procedures surrounding housing disputes can be challenging for individuals without legal expertise. This thesis aims to explore how a user-friendly and effective digital solution, like an LLM-powered virtual assistant, can assist individuals in navigating the housing dispute resolution processes in Finland.

Streamlining legal information retrieval:

Retrieving accurate and up-to-date legal information related to Finnish housing laws can be a time-consuming and complex task. By leveraging LLM-powered virtual assistants optimized with RAG, users may have the opportunity to access relevant legal information that is contextually appropriate and accurate, significantly streamlining the information retrieval process.

Evaluating the effectiveness of RAG optimized LLM in legal virtual assistants:

This thesis will assess the performance of RAG optimized LLM in the context of Finnish housing laws and dispute resolution. The evaluation will focus on the virtual assistant's ability to understand user queries, extract relevant information, and generate contextually appropriate responses.

Contributing to the advancement of AI applications in Finnish housing law:

By analyzing the effectiveness of LLM powered virtual assistants in the specific domain of Finnish housing law, this thesis aims to contribute valuable insights and knowledge to the field. This knowledge can be used to foster the development of innovative AI solutions that address specific legal challenges within Finnish housing law.

This thesis is commissioned by the FAIR (Finnish AI Region) project.

1.1 Objectives

- Investigate the effectiveness of Large Language Models (LLMs) optimized by Retrieval Augmented Generation (RAG) in the context of Legal Virtual Assistants designed for handling housing disputes in Finland.
- Assessing the accuracy and comprehensiveness of LLM-based virtual assistant responses to user queries related to Finnish housing laws and regulations.
- Evaluating the ability of the LLM to provide relevant and contextually appropriate information regarding various housing dispute scenarios.
- Gain insights into the strengths and limitations of LLM and RAG technologies in the virtual assistant and legal domain.

1.2 Scope

The research will be limited to the application of RAG-optimized LLM-powered virtual assistant within the domain of housing disputes in Finland.

The focus will be on Finnish housing laws and regulations, excluding international or comparative legal frameworks.

The evaluation of the LLM's effectiveness will primarily involve:

- Accuracy and factual correctness of responses to user queries.
- Comprehensiveness and relevance of the information provided.

Limitations:

- The thesis does not address broader legal domains beyond housing disputes in Finland, maintaining a specific and focused scope.
- The thesis does not extend to the development of legal policies or legislative changes; it focuses solely on the technical aspects of implementing LLM and RAG in the virtual assistant for housing disputes in Finland.
- The thesis does not aim to replace legal professionals or their expertise but rather to enhance and support their work in the specific context of housing disputes.
- Ethical considerations related to potential bias or misinterpretations by the LLM are excluded.

1.3 Research Questions

Q1. How does the integration of Large Language Models (LLMs) optimized by Retrieval Augmented Generation (RAG) enhance the performance of a Legal Virtual Assistant in resolving housing disputes in Finland?

This question examines the overall efficacy and impact of implementing LLMs optimized by RAG in a legal virtual assistant specifically tailored for addressing housing disputes in the Finnish context. It explains how the combination of LLM and RAG technology can accurately interpret, analyze, and provide legal advice based on Finnish housing laws and regulations.

Q2. Does the Retrieval Augmented Generation (RAG) optimization significantly enhance the factual accuracy of the Large Language Model (LLM) responses when compared to a non-optimized LLM in the context of housing disputes in Finland?

This question investigates the specific impact of RAG optimization on the factual accuracy of LLM-generated responses within the domain of housing disputes in Finland, comparing the performance of RAG-optimized LLMs against non-optimized counterparts to determine whether the optimization leads to a measurable improvement in accuracy.

Q3: What are the key challenges and limitations encountered when employing LLMs optimized by RAG within the domain of housing disputes in Finland?

This question investigates the practical obstacles and shortcomings associated with the utilization of LLMs optimized by RAG in addressing housing disputes, highlighting areas for improvement and potential barriers to effectiveness.

2 Literature Review

This chapter summarizes the key findings from the literature review and outlines the implications for both research and practical applications. It provides a comprehensive overview of the relevant technologies, including virtual Assistants and chatbots, Large Language Models (LLMs) and their capabilities in understanding and generating human language, Retrieval Augmented Generation (RAG) and its potential to enhance LLM performance in question-answering tasks. It also provides a brief overview of the housing dispute domain and some of the common types of disputes that occur in Finland.

The chapter also explores some real-world applications of human-AI collaboration in the housing domain. It includes research on a collaborative system for question answering (QnA) in German legal documents, which demonstrates the effectiveness of AI-assisted legal research. Additionally, the chapter discusses existing AI-powered chatbots or virtual assistants that provide services in the legal industry worldwide. Additionally, this chapter includes the Helmi chatbot service provided by HOAS which showcases how AI virtual assistants can streamline housing inquiries. Another example included is SmartRent, a UK-based initiative utilizing AI assistants to enhance the tenant experience in the housing market.

2.1 An overview of chatbot and virtual assistant technologies

This chapter explores the landscape of chatbots and virtual assistants (VAs), their general architecture, functionalities, and their recent applications. Understanding these technologies lays the groundwork for the development of an AI-powered legal virtual assistant optimized for housing dispute resolution in Finland.

2.1.1 Defining chatbots, virtual assistants, and legal virtual assistants

While often used interchangeably, chatbots and VAs possess subtle distinctions.

Chatbots primarily focus on simulating conversation through text or voice commands. They are frequently rule-based or rely on pre-defined scripts to address specific tasks like answering frequently asked questions (FAQs) or providing basic customer support (Adamopoulou & Moussiades, 2020). Chatbots can be integrated seamlessly into websites, messaging platforms, and mobile applications.

VAs offer a broader range of functionalities exceeding mere conversation. VAs can access and process information from various sources, schedule appointments, manage tasks, and even control smart home devices (Pereira et al., 2023).

While there are some technical distinctions between these terms, for the purpose of this paper, we will use them interchangeably to refer to AI-powered conversational agents that can interact with users through text queries.

2.1.2 Categories of chatbot

Based on their usage, functionalities and purpose, there are several types of chatbots in use today, each designed to meet specific needs and functions across various industries. Here we explore some of the widely used chatbot categories in recent times.

Rule based chatbots: Rule-based chatbots operate on a foundation of pre-defined rules and decision trees. These chatbots function in a script-like manner, responding to specific user inputs with predetermined outputs. Their strength lies in handling straightforward and repetitive tasks, excelling in scenarios where complex reasoning or nuanced understanding is not required(Lee et al., 2023). Rule-based chatbots typically address common user needs by answering Frequently Asked Questions (FAQs) on topics like return policies and store hours. They can also facilitate basic customer service interactions such as initiating returns or checking order status. Additionally, these chatbots excel at streamlining simple booking and reservation tasks, allowing users to book appointments or reserve products efficiently (Abd-alrazaq et al., 2019; Adamopoulou & Moussiades, 2020). A prime example of this functionality can be seen on retail websites, where rule-based chatbots assist users in finding products by size and color, provide store hours, and address common return policy questions.

AI-powered chatbots: AI-powered chatbots represent a significant advancement over rule-based counterparts. These chatbots incorporate machine learning and NLP capabilities to understand and respond to user inputs in a more human-like manner. This enables them to respond to complex and varied interactions, continuously learning and improving their response accuracy over time.(Al-Hasan et al., 2024; Işıkdemir, 2024). GPTs**,** such as ChatGPT by OpenAI and Gemini by Google, represent a cutting-edge form of AI chatbot. These models are trained on massive datasets of text and code, enabling them to generate human-quality responses that are contextually relevant and grammatically sound(Al-Hasan et al., 2024). This sophistication makes them highly effective for a wide range of conversational tasks. In addition to handling a variety of intricate user queries, GPTs can provide responses that are coherent and contextually relevant. They can be fine-tuned to be used in specific domains and applications such as providing health care advice or

legal advice (Patil & Gudivada, 2024). Typical use cases for GPTs include creating content in a variety of creative text formats, responding to intricate customer support queries, serving as domain-specific virtual assistants, and more(Rivas & Zhao, 2023). While GPTs offer immense potential to revolutionize AI and chatbots, their capabilities are constrained by limitations that can significantly impact real-world applications. These limitations include a lack of common sense and reasoning, potential for bias and discrimination, limited ability to understand context across multiple conversations, and a lack of explainability and transparency in their outputs (Koubaa et al., 2023). Additionally, the potential for malicious use and the high computational cost of training and running these models raise concerns for responsible development and accessibility(Lecler et al., 2023).

Other types of chatbots: Beyond rule-based, AI-powered, and GPT chatbots, other varieties exist. Contextual chatbots personalize interactions by remembering past conversations. Voice-activated chatbots offer hands-free convenience through spoken commands(Adamopoulou & Moussiades, 2020). Social media platforms see engagement boosted by social media chatbots. Secure transactions and bookings are facilitated by transactional chatbots(Whang et al., 2022). Both customer service chatbots and support chatbots provide technical assistance and troubleshooting solutions, with customer service chatbots often focused on broader customer interactions. Each category caters to specific user needs and interaction styles.(Skjuve et al., 2021)

Chatbot Categories		
	Knowledge domain	Generic
		Open Domain
		Closed Domain
	Service provided	Interpersonal
		Intrapersonal
		Inter-agent
	Goals	Informative
		Chat based/Conversational
		Task based
	Response Generation Method	Rule based
		Retrieval based
		Generative
	Human-aid	Human-mediated
		Autonomous
	Permissions	Open-source
		Commercial
	Communication channel	Text
		Voice
		Image

Figure 1. Categories of chatbots (Adapted from Adamopoulou & Moussiades, 2020)

Technologies and architecture of chatbots

In recent years, chatbots and virtual assistants have undergone remarkable transformations by leveraging AI, DL and ML to personalize interactions and provide more sophisticated assistance(Aslam, 2023; Hamid et al., 2023). This transformation is fueled by advancements in NLP integrated into various advanced LLMs(Işıkdemir, 2024). These advancements allow chatbots and VAs to understand the nuances of user queries and generate responses that are more accurate, relevant, and tailored to the user's specific needs(*ChatGPT, Generative AI, LLM, NLP: How to Understand the New Era of Artificial Intelligence Already Impacting Businesses - ProQuest*, n.d.).

User Interface (UI): The UI is the medium through which users engage with the chatbot, whether through text or voice. Chatbots can be incorporated into different platforms, including messaging apps, websites, and voice assistants.

NLP: In the architecture of developing chatbots, NLP is a crucial component for understanding and interpreting user inputs, enabling chatbots to engage in meaningful and coherent conversations (Filonova, 2022). NLP performs a range of sophisticated tasks mentioned as follows, that transform the raw text into structured data the chatbot can process.

Tokenization breaks down text into individual words or phrases, facilitating further analysis. Part-of-speech tagging identifies the grammatical roles of these tokens, such as nouns, verbs, and adjectives, which helps in understanding sentence structure (Attigeri et al., 2024). Named entity recognition (NER) involves identifying and classifying proper nouns like names of people, organizations, or locations within the text (Lareyre et al., 2023). Sentiment analysis evaluates the emotional tone of the input, determining whether it expresses positive, negative, or neutral sentiments (Shankar & Parsana, 2022). Together, these NLP tasks enable chatbots to accurately comprehend and respond to user inputs, enhancing their ability to provide relevant and contextually appropriate interactions (Whang et al., 2022).

Nowadays, NLP is seamlessly integrated into LLMs like ChatGPT and Gemini. These models leverage advanced NLP techniques to understand, process, and generate human-like text (Attigeri et al., 2024). By combining the extensive capabilities of LLMs with sophisticated NLP tasks such as tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis (Işıkdemir, 2024), these models can engage in complex, contextually rich conversations and provide highly relevant responses. This integration enhances the chatbot's ability to handle a wide range of queries with improved accuracy and coherence, making interactions more natural and effective (Koga & Du, 2024).

Generating Response: The process of response generation in a chatbot is crucial for creating relevant and contextually appropriate replies to user inputs (Kulkarni et al., 2021). It begins with analyzing the user's message using NLP to understand the intent and extract key information. The chatbot then generates a response using predefined rules, templates, or advanced methods such as machine learning and neural networks (Gao et al., 2023). Advanced chatbots, particularly those powered by LLMs like ChatGPT, use sophisticated algorithms to produce human-like and coherent responses (Bratić et al., 2024). These models can dynamically construct replies by predicting the next word in a sequence and considering the entire conversation history to maintain context. This capability allows chatbots to offer informative, engaging, and contextually appropriate interactions, ultimately enhancing user experience and satisfaction (Eshbayev et al., 2022).

Backend Integration: To provide relevant information or carry out specific tasks, chatbots frequently need to interact with external resources like databases or APIs. Backend integration allows chatbots to access and modify data from these external systems as required (Vasileiou & Maglogiannis, 2022)

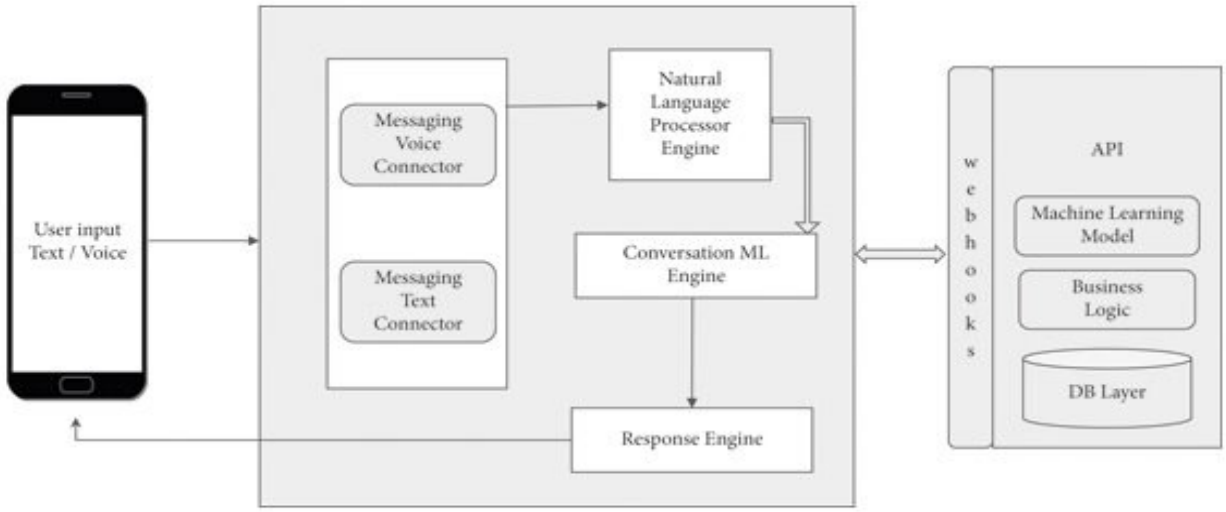

Figure 2. Advanced chatbot architecture (adapted from Vasileiou & Maglogiannis, 2022)

2.1.3 Potential of chatbots or virtual assistants in enhancing legal services

A virtual assistant may offer a diverse set of capabilities, including:

Information Retrieval: AI-powered chatbots have the ability to access and process vast amounts of data. They can efficiently retrieve relevant information based on user queries, saving time and effort (Khadija et al., 2023; Mathis, 2022). In the legal domain, a chatbot referred to as LVA can access and search through legal data, including case laws, statutes, regulations, and legal articles. This could be highly beneficial for legal professionals and individuals seeking specific legal knowledge.

Task Automation: VAs can automate repetitive tasks such as scheduling appointments, managing calendars, drafting basic legal documents, and handling routine client communication (Brunt-Work, 2023). This frees up valuable time for lawyers and legal professionals to focus on more complex legal matters and strategic tasks.

Understanding user queries: VAs are equipped with natural language processing (NLP) capabilities, enabling them to understand and respond to user queries in a natural and conversational manner (Shankar & Parsana, 2022). This allows users to interact with the legal system through intuitive language, making legal information more accessible and user-friendly.

Legal Research Assistance: LVAs can aid legal professionals in conducting legal research by identifying relevant legal precedents, analyzing legal documents, and summarizing key legal issues within a case.

Basic Legal Advice: An LVA can possess the ability to understand user queries about legal information, retrieve data from the knowledge base, and provide responses to legal inquiries. It can also answer common legal questions, direct users to relevant legal resources, and recommend seeking professional legal assistance when needed.

Document Drafting Support: LVAs with the help of artificial neural networks can assist in drafting simple legal documents like contracts, wills, and non-disclosure agreements with user input, streamlining the document creation process.

With the aforementioned capabilities of virtual assistants in the legal domain, they have the potential to significantly impact the nature and delivery of legal services:

Increased Access to Legal Information: LVAs can democratize access to legal knowledge, particularly for individuals who may not have the resources to consult a lawyer. This can empower individuals to better understand their legal rights and navigate the legal system.

Improved Efficiency and Productivity: By automating routine tasks and providing basic legal assistance, LVAs can significantly improve the efficiency and productivity of legal professionals, allowing them to focus on high-value legal services and strategic advice.

Cost Reduction: The automation of tasks and provision of basic legal assistance by LVAs can potentially reduce legal service costs for clients, making legal services more affordable for a wider range of individuals and businesses.

Figure 3. Potential application of LVA.

2.1.4 Challenges and limitations associated with adoptions of chatbots

While chatbots offer numerous advantages as mentioned in previous chapter, they also come with many limitations and challenges.

Limited understanding: Chatbots may struggle with understanding complex or ambiguous queries (Schwenke et al., 2023), especially in domains requiring an accurate understanding of the scenarios, such as law or healthcare. Additionally, maintaining context over extended conversations can be challenging, leading to inconsistencies or misunderstandings. Moreover, chatbots may lack empathy and emotional intelligence, hindering their effectiveness in sensitive or emotionally charged interactions (Murtarelli et al., 2021).

Data bias and hallucinations: Chatbots trained on biased data can perpetuate those biases in their responses. Data bias is a major concern for chatbots, as they learn from the information they're trained on. Biases in that data, whether from skewed samples, historical prejudices, or even the algorithms themselves, can lead chatbots to perpetuate stereotypes, discriminate against users, and deliver unfair treatment (Lin et al., 2023; Murtarelli et al., 2021). Providing inaccurate information, expressing nonsensical statements, or sharing false information is considered hallucinations committed by the LLM models (Żmihorski, 2023).

Security and Privacy: Chatbots that handle sensitive data raise concerns about data security and user privacy (Rivas & Zhao, 2023). Ensuring data privacy and security is crucial, especially when chatbots handle sensitive information.

Cost and investment: Creating and managing chatbots requires a significant investment of time, resources, and ongoing improvement to keep up with changing user needs and technological advancements (Schwenke et al., 2023). These limitations and challenges highlight the importance of thorough planning, continuous improvement, and careful use of chatbot technology to achieve specific business objectives while minimizing potential drawbacks (Lin et al., 2023; Patil & Gudivada, 2024).

2.2 Existing AI-powered chatbots in the legal arena around the world

This chapter dives into the world of AI assistants which are specifically designed for the legal domain. We'll explore how these AI-powered assistants are operating in the legal domain and transforming the domain, highlighting some of the most prominent players such as Juro, Harvey AI, and LawDroid.

2.2.1 Juro legal AI assistant

Juro, a legal technology platform leverages AI to streamline the entire contract management process. Unlike standalone legal AI chatbots, Juro integrates its AI assistant within a comprehensive contract automation platform. This allows users to not only draft, summarize, and review contracts with exceptional speed, but also handle the entire contract lifecycle: creation, negotiation, approval, signing, storage, and management (*7 Best Legal AI Chatbots for 2024*, n.d.).

Juro's key differentiator lies in its holistic approach. By embedding the AI assistant directly within the contract workflow, it provides contextually relevant and accurate responses. This eliminates the limitations of siloed AI solutions that lack access to an organization's specific practices and guidelines. Juro prioritizes user privacy as well. Data remains within the EEA (European Economic Area) for GPT interactions, and contracts or prompts are never used to train large language models (LLMs). This directly addresses security and privacy concerns that are increasingly important for in-house legal teams (*7 Best Legal AI Chatbots for 2024*, n.d.).

Juro's AI assistant tackles a major pain point in contract management: creating clear and concise summaries. Legal contracts are often lengthy documents filled with jargon, making it difficult for stakeholders outside the legal department to grasp key details. Juro's AI can automatically generate contract summaries in seconds, extracting the critical information and presenting it in a clear, digestible format (*Contract Summary: What It Is and How to Create One*, n.d.).

2.2.2 Harvey AI legal AI assistant

Founded in 2023 by legal and AI experts, Harvey AI prioritizes security and user trust. This is evidenced by their impressive growth. Harvey AI has a team exceeding 100 people, a tenfold revenue increase, and a $715 million valuation secured through $80 million in Series B funding. Harvey AI caters to legal, tax, and finance professionals, offering a secure platform for leveraging cutting-edge AI within their workflows (*Harvey | OpenAI*, n.d.).

The legal AI chatbot of Harvey AI is designed to empower law firms and consulting companies. Recognized as a leader in the legal AI market, Harvey AI partners with prestigious firms like Allen & Over and industry giants like PwC. Similar to Juro's legal AI assistant, Harvey AI leverages OpenAI's GPT technology and machine learning to automate routine legal tasks, including due diligence, litigation support, and legal document analysis(*7 Best Legal AI Chatbots for 2024*, n.d.).

Harvey AI distinguishes itself through its comprehensive training process. While its core is built upon OpenAI's GPT foundation, it undergoes further specialization in legal domains. This involves training on vast amounts of legal data encompassing case law, legal reference materials, and

industry best practices. Additionally, upon integration with a specific law firm, Harvey AI personalizes its capabilities by ingesting the firm's internal templates and work products. This mimics the onboarding process of a new legal professional, ensuring Harvey AI delivers outputs tailored to the firm's specific practices and areas of expertise(*Generative AI for Professional Services | Harvey*, n.d.; *Harvey AI: Legal Artificial Intelligence*, n.d.).

2.2.3 CaseMine legal AI assistant

CaseMine's journey began with CaseIQ, a pioneering tool utilizing extractive AI. CaseIQ excels at navigating vast legal databases, pinpointing relevant information through advanced natural language processing (NLP) techniques. It identifies key passages and highlights pivotal sections of judgments using the Importance Matrix, offering a visual guide to the most influential parts of legal documents. CiteText, another extractive AI tool, complements CaseIQ by distilling the interpretation of legal precedents. By extracting the essence of how courts have applied past judgments, CiteText streamlines research by showcasing authoritative legal applications (*Evolution of Legal AI from Extractive to Generative - The CaseMine Story*, n.d.).

Recognizing the limitations of extractive AI, CaseMine took a bold step forward with AMICUS, a game-changing generative AI tool launched in 2023. AMICUS transcends simple information retrieval, transforming into a comprehensive legal research assistant. It offers a variety of functionalities:

Conversational Research: AMICUS engages in conversations, not just returning results. It delves deeper, reasoning about the applicability of legal information and ensuring users receive the most relevant answers to complex legal questions.

Legal Document Drafting: AMICUS streamlines document creation by generating precise and compliant legal documents, minimizing human error and saving valuable time.

Summary Generation: AMICUS provides concise and accurate summaries of legal materials, surpassing traditional headnotes and ensuring crucial information is never overlooked. (*7 Best Legal AI Chatbots for 2024*, n.d.; *Evolution of Legal AI from Extractive to Generative - The CaseMine Story*, n.d.)

2.3 HOAS and the Helmi Chatbot: AI-powered Housing Assistance

HOAS, a non-profit organization dedicated to providing affordable student housing, faced a growing demand for customer service. Their existing live chat system, while popular, struggled to keep pace with the increasing volume of inquiries. Additionally, many student inquiries were

repetitive and could potentially be addressed through self-service options (*Hoas - Hoas*, n.d.; *Housing Chatbot Improves the Overall Customer Satisfaction | GetJenny*, n.d.).

In 2018, HOAS partnered with GetJenny to create Helmi, a Finnish and English speaking AI chatbot. Helmi seamlessly integrated with HOAS's existing chat system, allowing it to handle routine inquiries and direct more complex issues to human agents.

The implementation of Helmi resulted in immediate improvements. Within the first few months, HOAS met its goals for the chatbot project, including a notable increase in customer service satisfaction scores from 4.11 to 4.26 (on a scale of 1-5). Helmi's ability to provide instant, accurate responses helped alleviate the workload on human agents and improved the overall efficiency of customer service operations.

Helmi operates 24/7, offering round-the-clock assistance to users. This accessibility significantly improves convenience and eliminates time constraints associated with traditional housing searches. Helmi's user interface is designed to be user-friendly and intuitive, aiming to remove barriers often encountered in conventional housing searches. This simplifies the process for users of all technical backgrounds.

Streamlined Inquiries: Helmi assists users with various housing-related inquiries, including: Housing application procedures, Real-time updates on available properties, Lease agreement details, Rent payment information, and Maintenance request submission.

Improved Efficiency: By automating routine tasks and inquiries, Helmi frees up HOAS staff to focus on more complex issues and individual needs. (*Hoas - Hoas*, n.d.; *Housing Chatbot Improves the Overall Customer Satisfaction | GetJenny*, n.d.)

Limitations and Considerations:

While Helmi offers significant benefits, it's crucial to acknowledge its limitations:

Complexity Handling: Helmi's capabilities are primarily focused on addressing common and straightforward inquiries. When faced with complex or nuanced issues requiring human judgment or interpretation, the chatbot may struggle to provide adequate solutions.

Accent/Dialect Recognition: In some instances, Helmi may encounter difficulties understanding users with specific accents or dialects. This could potentially lead to misinterpretations or communication breakdowns.

Overall Impact:

Despite these limitations, Helmi streamlines the housing process for users while enabling HOAS to optimize its operational efficiency. (*Chatbot Use Cases: 25 Real-Life Examples*, n.d.; *Hoas - Hoas*, n.d.; *Housing Chatbot Improves the Overall Customer Satisfaction | GetJenny*, n.d.)

2.4 SmartRent in the United Kingdom: AI-powered Assistance for Tenants

Across Europe, various organizations are adopting AI-powered solutions to address housing challenges. In the United Kingdom, SmartRent offers a prime example of such an initiative.

SmartRent is a leading provider of AI-powered property management solutions in the UK. Their platform incorporates a virtual assistant that caters specifically to tenants' needs.

Similar to Helmi, SmartRent's virtual assistant operates 24/7, providing round-the-clock assistance to tenants with their housing inquiries. The platform facilitates convenient rent payments through various online channels, eliminating the need for manual processes or late payment penalties.

Tenants can easily submit maintenance requests through the platform, ensuring timely communication and resolution of any property issues.(*5 Ways SmartRent UK Is Revolutionising Smart Home Tech | SmartRent*, n.d.; *Leveraging Artificial Intelligence for Property Management - Kurby Real Estate AI*, n.d.; *SmartRent | Smart Home Solutions for Multifamily Communities*, n.d.; *SmartRent Delivers Seamless Property Management with Salesforce - Salesforce*, n.d.)

The virtual assistant acts as a central communication hub, allowing tenants to receive important updates and announcements from their landlords or property managers.

Tenants can access personalized information about their tenancy agreements, including lease details, payment history, and property-specific rules and regulations.

SmartRent's AI-powered solution also benefits landlords and property managers by:

Streamlining Communication: The platform automates routine communication tasks, freeing up time for property managers to focus on more complex issues.

Improved Efficiency: Online rent payments and maintenance request management enhance operational efficiency and reduce administrative burdens.

Data-driven Insights: The platform provides valuable data and analytics that can be used to optimize property management strategies and improve tenant satisfaction.

(*5 Ways SmartRent UK Is Revolutionising Smart Home Tech | SmartRent*, n.d.; *Leveraging Artificial Intelligence for Property Management - Kurby Real Estate AI*, n.d.; *SmartRent | Smart Home*

Solutions for Multifamily Communities, n.d.; *SmartRent Delivers Seamless Property Management with Salesforce - Salesforce*, n.d.)

Overall Impact:

SmartRent's AI-powered virtual assistant demonstrates the growing trend of utilizing AI to improve the tenant experience in the UK housing market. By offering 24/7 support, convenient payment options, and a centralized communication platform, SmartRent streamlines the housing experience for tenants while providing valuable tools for efficient property management.

2.5 Collaborative system in QnA in German case law documents

This section discusses the work of C. Hoppe et al. (2022) who developed a system for efficient legal information extraction through semantic Question and Answer (QnA) in German case law documents. Here's a breakdown of their approach, findings, and the significance of their research:

2.5.1 Human-AI collaboration approach

In order to extract information efficiently from digitally stored legal systems, Hoppe et al., 2022 adopted Human-AI collaboration approach. Human-AI collaboration is focused on achieving a common objective through extensive interaction between humans and AI. The challenge is, legal laypersons cannot effectively extract information from vast quantities of legal documents on their own. Therefore, in the system developed by (Hoppe et al., 2022), rather than manually searching through numerous documents, individuals use a graphical interface or user interface to interact with an AI that handles the document search and question-answering (QA) tasks, this is mentioned as the human layer in their work. The AI layer comprises a toolkit of interchangeable statistical and machine learning components designed to efficiently address search tasks. Thus, the system is structured into two interconnected and continuously interacting layers: the human layer (front end) and the AI layer (back end).

The Human Layer: To facilitate the collaborative interaction between humans and the QA system developed by (Hoppe et al., 2022) in this work, users can interact in two primary ways.

1. **Programming interface:** Users can enhance the knowledge base by inserting new legal documents. In the programming interface, these documents are automatically pre-processed, embedded, indexed, and saved in the database, thus no programming skills are required.
2. **Graphical Human-AI Interface:** Users can submit search queries, which can be either individual keywords or specific legal questions through the graphical user interface. Queries may include legal-specific terms and depending on the search term, various components of

the AI toolchain (described in the AI layer section) are used to find the best-matching documents. Thus, the top results are prepared for the user to directly extract the requested information. Moreover, In an evaluation loop, the presented documents can be optionally rated to improve the quality of future responses.

The AI layer: When a search term is submitted through the human layer, the AI layer begins retrieving relevant information.

For the document retrieval, it uses statistical methods and deep sentence transformer models to transform queries into vectors of dimension d = 768. Then it computes cosine similarity to find the most relevant passages. Hence, it retrieves the top k = 10 relevant passages related to the search query.

If keywords are used by the user, a BERT (Bidirectional Encoder Representations from Transformers) model which is trained for the named entity recognition checks for legal entities.

(Note: The BERT model is a state-of-the-art language representation model developed by Google. It leverages a transformer architecture to understand the context of words in a sentence by looking at both preceding and following words, enabling more accurate natural language processing tasks (Hoppe et al., 2022).)

For specific legal questions, an ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacement Accurately) model extracts exact answers from the relevant passages which retrieves the top k = 5 answers most relevant to the posed question. The AI layer also ranks and presents the relevant passages or answers on the graphical human-AI interface.

(Note: ELECTRA is a pre-trained language model introduced in 2020 by researchers at Google Research. It utilizes a novel technique called "replaced token detection" during the training process (*More Efficient NLP Model Pre-Training with ELECTRA*, n.d.). This technique involves masking a portion of the input text and replacing it with another random word. The model then learns to predict the original masked word based on the surrounding context, but crucially, it does not predict the replaced word itself. This approach helps the model become more robust against adversarial attacks and better at understanding natural language (Gargiulo et al., 2022). GELECTRA is a variant of ELECTRA specifically adapted for the German language. It uses the same principles as ELECTRA to enhance language understanding and processing tasks in German, providing similar efficiency and performance benefits for German-language applications (Chan, 2020).)

For the document pre-processing and Indexing which is done through the programming interface, involves removing HTML elements, converting Unicode symbols, and splitting long documents into 200-word passages.

Document indexing Stores plain text, metadata, and deep vector representations of passages in the database. Vectors enable semantic search and question answering. The process completes when all texts, metadata, and passage vectors are successfully indexed.

2.5.2 The experiment and the metrics

The experiments conducted aimed to evaluate the performance of the system using various underlying models. (Hoppe et al., 2022), tested several pre-trained models alongside a self-trained reader model and introduced a self-annotated dataset called LegalQuAD specifically for QA tasks in German case law documents.

Creation of the LegalQuAD data set

To assess and fine-tune the retriever and reader models for QA, (Hoppe et al., 2022), needed annotated datasets with question-answer pairs. Notably, no such dataset existed for QA in German legal documents. To address this gap, (Hoppe et al., 2022) developed the LegalQuAD dataset, which consists of 226 question-answer pairs derived from German case law documents across various legal fields, formatted similarly to the SQuAD dataset. The data annotation was performed by trained lawyers familiar with NLP, who crafted specific questions and highlighted corresponding answers from provided legal texts. To ensure diversity, questions varied in complexity and were rephrased with synonyms to minimize lexical overlap. Additionally, all questions were made self-sufficient, ensuring they could be answered solely based on the text provided.

Model training and evaluation metrics

(Hoppe et al., 2022) evaluated different combinations of the entire QA workflow on their annotated LegalQuAD dataset. Several publicly available models were tested against this dataset, including the retrieval methods BM25 and MFAQ, combined with the reader models GELECTRA-base-GermanQuAD and GELECTRA-large-GermanQuAD. Additionally, (Hoppe et al., 2022) fine-tuned their own reader model, GELECTRA-large-GermanQuAD-LegalQuAD, using 200 random question-answer pairs from LegalQuAD. This fine-tuning was conducted over two epochs with a learning rate of 1e-5, using Adam optimizer, a batch size of 10, and a maximum sequence length of 256 tokens.

To evaluate the models, (Hoppe et al., 2022) used two primary metrics as follows.

The first one is Exact Match (EM) which measures the proportion of documents where the predicted answer span exactly matches the correct answer span. This metric is highly precise and stringent; for example, a predicted answer "§ 15 BGB." would score zero if the correct answer was "In § 15 BGB." due to the exact match requirement.

The second one is F1-score which measures the ratio of overlapping words between the labeled and predicted answer spans. This metric is more lenient than EM and provides a score closer to human judgment regarding the similarity of answers. These metrics helped (Hoppe et al., 2022) to determine the accuracy and effectiveness of the models in extracting the correct information from the legal documents.

Results:

The experiments showcased the effectiveness of a collaborative question-answering system for German legal documents. (Hoppe et al., 2022)'s approach, combining retriever models MFAQ and BM25 with a self-trained reader GELECTRA-large-GermanQuAD-LegalQuAD, outperformed pre-trained models in terms of EM and F1-scores. This suggests that fine-tuning models specifically for legal language significantly improves performance.

Their workflow proved functional, achieving valuable results compared to state-of-the-art methods. Human-AI collaboration in preprocessing, indexing, and querying legal documents was shown to be manageable for both legal laypersons and lawyers, without requiring programming skills. Notably, the fine-tuned models exhibited substantial performance improvements, indicating the difficulty of generalizing legal language for pre-trained models. This research offers promise for extracting precise answers from extensive legal documents in the future, even with limited labeled data.

Synthesis of the work:

The study introduces a collaborative question-answering system tailored for German legal documents. Their approach combines retriever models (MFAQ and BM25) with a self-trained reader (GELECTRA-large-GermanQuAD-LegalQuAD), demonstrating superior performance over pre-trained models in terms of EM and F1-scores. This underscores the importance of fine-tuning models for the intricacies of legal language.

As the field progresses towards AI-based methods in information retrieval (IR), the necessity for collaborative systems for AI-supported question answering and semantic search in legal domains becomes apparent. Integrating humans into the process chain is crucial, ensuring legal information accessibility to both laypersons and professionals through a human-AI interface. These findings

can enhance existing information retrieval systems, making legal documents more transparent and searchable for all users, thereby contributing to a smarter society.

2.6 Housing disputes domain in Finland

We provide a brief overview of housing dispute domain in Finland so that we better describe the scope of work for the developed prototype.

The housing sector in Finland plays a crucial role in the country's social and economic landscape and it can have many legal issues occurred. However, navigating housing disputes can be a complex and challenging process for both individuals and legal professionals. This chapter identifies common challenges associated with resolving them, and it explores the potential of AI-powered virtual assistants in addressing these challenges.

Common types of housing disputes in Finland:

In Finland, housing disputes frequently stem from a variety of issues. One of the most common sources of conflict is rent arrears and non-payment, where tenants may face potential eviction proceedings due to unpaid rent. Maintenance and repairs also often lead to disagreements, as tenants and landlords dispute over who is responsible for maintaining and repairing rental properties. Additionally, noise disturbances and nuisance behavior by neighbors can create significant challenges for tenants, leading to tensions within housing communities. Another major area of dispute is the termination of tenancy agreements, where the validity or fairness of termination notices can become contentious and typically necessitates legal guidance. These issues collectively contribute to the complexity of housing disputes in Finland, affecting both tenants and landlords (Kettunen & Ruonavaara, 2015).

Challenges in resolving housing disputes:

Individuals and legal professionals navigating housing disputes in Finland encounter several challenges. Retrieving relevant information is often time-consuming and requires legal expertise to identify the applicable legal provisions and regulations for a specific dispute. Generating appropriate responses, such as formulating effective legal arguments and drafting legal documents, can be difficult without proper legal training and experience. Additionally, access to legal representation poses a significant barrier, as the cost can be prohibitive, especially for those with limited financial resources. These challenges collectively complicate the resolution of housing disputes in Finland.

Potential of AI-powered virtual assistants in housing disputes:

AI-powered virtual assistants (VAs) have the potential to address several challenges associated with housing disputes in Finland. These VAs can streamline information retrieval by providing easy access to relevant legal information, regulations, and case law specific to housing disputes (Pereira et al., 2023). They can assist in response generation by helping individuals draft basic legal documents, such as letters to landlords or applications for legal aid, using templates and guided prompts. By offering insights into legal rights and options, VAs empower individuals to participate more effectively in the dispute resolution process (Quali-Bot, the Virtual Assistant That Also Helps in Legal Claims Issues, 2024). Furthermore, they provide a cost-effective alternative to legal representation, especially beneficial for those with limited financial resources.

2.7 LLM and RAG Overview

Large Language Models (LLMs) have emerged as a powerful force in the field of Artificial Intelligence (AI), particularly within the realm of Natural Language Processing (NLP) (Bratić et al., 2024). These advanced AI systems, often referred to as chatbots, are known for their ability to understand and generate human-like language, making them capable of performing tasks such as question answering, essay writing, creative content generation, and even code writing (Teubner et al., 2023). Prominent examples of LLMs include OpenAI's ChatGPT, particularly its latest iteration GPT-4 Vision, Google's Bard AI (now Gemini), and Microsoft's Bing Chat (Miao et al., 2024). Their rapid development has facilitated their adoption across various domains, including business, academia, and even the legal sector.

LLMs are essentially a subset of deep neural networks built on the transformer architecture. Their remarkable skills in understanding, generating, and modifying natural language stem from their training on massive amounts of text data, often encompassing billions of words (Ullah et al., 2024). This training process utilizes a "next word prediction" approach, where the model learns to predict the next word in a sequence given a set of preceding words. Through this iterative process, LLMs gain the ability to grasp the nuances of human language and identify the intricate statistical patterns within it (Teubner et al., 2023).

As (Min et al., 2024) describes that LLMs serve as the foundational models for NLP and Natural Language Generation (NLG) tasks. After undergoing pre-training on vast datasets, these models are further refined through techniques like zero/one/few-shot learning and in-context learning, enabling them to effectively manage the complexities and interconnectedness inherent in language(Işıkdemir, 2024).

Despite their impressive capabilities, LLMs face certain challenges. One such issue is the "hallucination problem," where the model might fabricate stories or facts that appear plausible but lack factual grounding, often due to limitations in the data it was trained on (Kang et al., 2024; Ott et al., 2023). This can lead to the generation of inaccurate information presented in a seemingly credible manner.

Retrieval-Augmented Generation (RAG) emerges as a promising solution to address these limitations (Gao et al., 2024). RAG integrates knowledge from external databases, enhancing the accuracy and credibility of the generated output, particularly for knowledge-intensive tasks. It also facilitates continuous knowledge updates and the incorporation of domain-specific information (Alan et al., 2024). By synergistically combining the inherent knowledge of LLMs with the vast dynamic repositories of external databases, RAG effectively reduces the risk of generating factually incorrect content. This integration has led to widespread adoption of RAG, establishing it as a key technology in advancing chatbots and enhancing the suitability of LLMs for real-world applications (Miao et al., 2024).

Alternative solutions have been proposed to address the challenges faced by LLMs, such as directly incorporating all information into prompts or fine-tuning models with fresh data (Li et al., 2024). However, the former approach is often impractical, while the latter incurs significant financial costs. The RAG approach presents a viable alternative, as it retrieves relevant information from stored databases when required and provides it to the LLM. This approach utilizes these references to generate more accurate and reliable responses by equipping LLMs with pertinent questions and related reference resources beforehand (Li et al., 2024).

The implementation of the Proof of Concept (POC) for this thesis work leverages the RAG model as its core method. By storing information, utilizing databases, and enhancing the generation process through the inclusion of relevant reference materials, the RAG model ultimately contributes to improved answer quality and reliability (Gao et al., 2024).

3 Research Design

This chapter presents Methodological procedure which includes the research approach, development Method, implementation procedures, data collection methods and its reasons, and data analysis methodologies that were adopted to assure validity and reliability of the findings.

3.1 Research Approach

Constructive Research approach: This thesis work, titled "Assessing the Efficacy of Large Language Model (LLM) optimized by Retrieval Augmented Generation (RAG) in Legal Virtual Assistant for the domain of Housing Disputes in Finland," adopts the constructive research approach. This methodology emphasizes the development and implementation of practical solutions to specific problems or challenges. In this context, it involves building a functional legal virtual assistant (VA) specifically designed to address housing disputes within the Finnish legal system.

The constructive research approach offers several advantages for this project. Firstly, it facilitates the creation of a tangible tool, the legal VA, that can be directly applied in real-world scenarios. This allows for the evaluation of its effectiveness in assisting individuals navigating housing disputes within the Finnish legal framework. Secondly, by actively building and implementing the legal VA, the research delves into the practicalities of optimizing LLMs with RAG for legal information retrieval tasks. This hands-on experience provides valuable insights into the real-world challenges and opportunities associated with this approach. Finally, the constructive approach inherently emphasizes problem-solving, ensuring that the research directly contributes to finding a practical solution for a specific challenge within the housing dispute domain in Finland.

Therefore, the constructive research approach aligns perfectly with the goals of this thesis, which include, developing a functional legal VA capable of assisting individuals with housing disputes in Finland, evaluating the efficacy of LLMs optimized with RAG in the context of legal information retrieval for housing disputes and To gain practical experience in optimizing LLMs with RAG for legal tasks, contributing to the broader understanding of this approach within the legal domain.

3.2 Scrum Iterative Development Framework

This thesis work involves developing a prototype of an AI-powered virtual assistant which will be able to handle basic housing disputes issues in Finland. To implement this development project, we planned and organized our work with the Scrum Iterative development framework.

Scrum is an iterative and incremental development framework used for managing complex software and product development projects (Alavandhar & Ņikiforova, 2017). It provides a structured yet flexible approach to product development, emphasizing collaboration, transparency, and adaptability. Two key components of Scrum are "Sprints" and "Product Backlog." (Pries & Quigley, 2011)

Sprints: Scrum operates in fixed-length iterations called sprints, typically lasting 2 to 4 weeks. By breaking the work into smaller, manageable increments and iterating over short time frames, Scrum aims to deliver value quickly and respond effectively to changing requirements or feedback. The idea of sprints in Scrum is to create a rhythm of regular, focused work that allows for adaptability and continuous improvement (Alavandhar & Ņikiforova, 2017).

Product Backlog: A product backlog is a prioritized list of features, enhancements, bug fixes, and other work items that need to be addressed in a product. It is a key component of agile development methodologies, particularly Scrum (Pries & Quigley, 2011). The product backlog serves as a dynamic document that evolves and reflects the overall vision and goals of the product.

3.3 Investigation Methods

This thesis employs an experimental setup to assess the efficacy of three popular LLMs, along with our prototype, a RAG-based legal VA, specifically for housing disputes in Finland. The experimental setup allows for controlled testing of the LLMs' performance under specific conditions. By presenting a defined set of prompts and scenarios, we can directly observe and evaluate the responses generated by different LLMs and our prototype. The detailed structure of the experimental setup is described in chapter 3.3.1.

The results from the experiment are evaluated through qualitative feedback provided by a human expert specialized in legal perspectives. This feedback helps in assessing the accuracy, completeness, and clarity of the responses generated by each model. The details of the qualitative feedback are described in chapter 3.3.2.

3.3.1 Structure of experimental setup

Targeted issues

Housing disputes encompass a wide range of issues that can arise between landlords and tenants, neighbors, homeowners associations, and other stakeholders. Due to limitations of time and resources, we have focused on two specific issues within the housing dispute domain:

1. **Unable to Pay Rent:** This encompasses various situations where individuals face difficulty fulfilling their rent obligations.
2. **Damage to Property:** This covers issues related to property damage, including the identification of responsibility, repair processes, and potential compensation.

Legal assistance scenarios

For each targeted issue, we have identified five distinct legal assistance scenarios. These scenarios represent common situations that individuals might encounter within the Finnish housing dispute legal framework. The scenarios were selected based on a study of commonly occurring housing dispute issues in Finland, and through consultation with the commissioning party of this thesis.

For example, scenarios related to "Unable to Pay Rent" includes:

- Seeking for rent assistance programs.
- Negotiating rent reduction with the landlord.

Prompt generation

Within each scenario, 5 different user queries/prompts are formulated. These prompts are designed to simulate user inquiries, incorporating variations in phrasing, grammar, and potential spelling mistakes to reflect the diverse ways users might interact with a legal virtual assistant.

Total prompts

Following this structure, a total of 50 prompts (5 prompts per scenario x 5 scenarios x 2 issues) are generated for evaluation.

Generating responses from different virtual assistant platforms

The generated 50 prompts are presented to four different virtual assistants: ChatGPT, Gemini, Perplexity, and the prototype which was developed as a proof of concept (POC). This allows for a qualitative analysis of their performance in handling legal information retrieval and generation tasks within the housing dispute domain.

Details of these scenarios, prompts and responses are provided in the Appendix, where an .xlsx file includes all relevant information.

3.3.2 Qualitative Feedback for evaluating the experiment results

In this section, we detail the methodology used to gather qualitative feedback for evaluating the experiment results. The qualitative feedback is provided by a single human expert with experience in legal matters related to housing disputes. This feedback is crucial in assessing the efficacy of the different AI language models and the RAG-based prototype as legal virtual assistants.

Structure of the Qualitative Feedback Form

Feedback is collected from the human expert through a feedback form.

The feedback form is structured to evaluate the performance of AI language models (LLMs) and the prototype for RAG-based LVA in addressing housing dispute queries.

A human expert with legal expertise in housing dispute domain fills out the form by reviewing the results from the experiment. It comprises sections designed to assess two aspects of the models' responses, including accuracy and completeness. Each section is accompanied by specific criteria to guide the evaluator in providing qualitative feedback.

The feedback form is designed to comprehensively assess how well the models provide legal advice for housing disputes. It starts by outlining the evaluation's purpose and then collects the evaluator's information. The core of the form focuses on predefined criteria for evaluating the models' performance. These criteria consist of two main categories: correctness and completeness. Correctness is broken down into factual accuracy, source reliability, and internal consistency. Completeness is evaluated based on coverage of key points, detail sufficiency, and information relevance. Finally, the form concludes with an open-ended section allowing the evaluator to provide feedback on strengths and weaknesses and recommend improvements. This structured format ensures a systematic and thorough assessment of the models' effectiveness.

Please refer to the Appendix section for the feedback form.

Methods of data analysis

The responses generated by each virtual assistant for each prompt are documented in an Excel spreadsheet. This spreadsheet containing the documented performance results from each of the LLMs including our POC is evaluated by a human expert in the legal field. The human expert evaluates the LLMs' and the POC's performance by filling up a qualitative feedback form.

The following data analysis methods will be employed:

Quantitative analysis:

The accuracy scores for each virtual assistant across all prompts will be calculated to provide a quantitative comparison of their performance, marking as accurate or not accurate, complete or incomplete.

For the RAG-based system, it will be calculated that how many times it successfully retrieves data from the sources.

Qualitative analysis: Error analysis based on qualitative feedback

Based on the feedback, types of errors made by each virtual assistant will be identified and categorized (e.g., factual inaccuracies, misinterpretations, irrelevant information). This analysis will provide insights into the specific strengths and weaknesses of each model.

By combining these quantitative and qualitative analyses, we will draw conclusions regarding the effectiveness of the LLM and the prototype, RAG-optimized language models in legal information retrieval tasks within the context of housing disputes, we will also identify and discuss limitations, challenges, implications, future research directions.

4 Implementation of the Prototype

4.1 Development platform: Azure services

The prototype is developed on the Azure platform, a cloud-based solution designed to simplify the process of building modern applications. Microsoft's Azure service provides the flexibility to either host applications entirely in the cloud or extend on-premises applications with Azure services. This flexibility ensures that applications are scalable, reliable, and maintainable. Azure supports a wide range of popular programming languages, including Python, JavaScript, Java, .NET, and Go, and offers a comprehensive SDK library along with extensive support in development tools like VS Code, Visual Studio, IntelliJ, and Eclipse.(*Azure OpenAI Service Models - Azure OpenAI | Microsoft Learn*, n.d.)

Azure can host the entire application stack, encompassing web applications, APIs, databases, and storage services. It also allows existing on-premises applications to incorporate Azure services to enhance their capabilities. For instance, applications can utilize Azure Blob Storage for cloud file storage, Azure Key Vault for secure storage of application secrets, or Azure AI Search for adding full-text search capabilities. These services are fully managed by Azure and can be seamlessly integrated into existing applications without altering the current architecture or deployment model. Additionally, Azure offers various container-based services to support application modernization.

Azure Functions facilitate the creation of solutions for event-driven workflows, such as responding to HTTP requests, handling file uploads in Blob storage, or processing queue events. Azure OpenAI Service provides REST API access to powerful language models, including GPT-4 and GPT-3.5-Turbo, which are available for general use. These models can be adapted for specific tasks like content generation, summarization, image understanding, semantic search, and natural language to code translation. Users can access the service through REST APIs, Python SDK, or Azure's web-based interface in Azure OpenAI Studio.(*Azure OpenAI Service Models - Azure OpenAI | Microsoft Learn*, n.d.)

Azure AI Search is a fully managed cloud search service that enables information retrieval over user-owned content (*Introduction to Azure AI Search - Azure AI Search | Microsoft Learn*, n.d.). It hosts search services that manage indexes, indexers, data sources, skillsets, and synonym maps. A search index provides persistent storage of search documents, which are structured data loaded from external sources and made searchable. An indexer automates the process by reading data in native formats, serializing it into JSON, and optionally applying AI enrichment through skillsets for enhanced indexing (*Azure OpenAI Service Models - Azure OpenAI | Microsoft Learn*, n.d.).

Overall, Azure offers a comprehensive suite of cloud services catering to various application development and deployment needs. Its support for popular programming languages, flexible hosting options, and integration with AI services like Azure OpenAI and Azure AI Search make it a powerful platform for building modern, scalable applications.

4.2 System model

Azure OpenAI and Azure AI search integration

The POC of the virtual assistant was built on an open-source sample app provided by Microsoft Azure for the Retrieval-Augmented Generation pattern running in Azure, using Azure AI Search for retrieval and Azure OpenAI large language models to power ChatGPT-style and Q&A experiences.

To implement the system model, it was installed in a local environment. For a local setup, installation of the required tools are as follows: Azure Developer CLI, Python (3.9 or higher), Node.js 14+, Git, and PowerShell 7+ (for Windows users). Then ensured Python and pip are in the PATH on Windows, and python version can be run from the console. Then the following command is run:

azd init -t azure-search-openai-demo

Above command will initialize a git repository and the program will be cloned from the git repository.

Link to the git repository is: https://github.com/Azure-Samples/azure-search-openai-demo?tab=readme-ov-file#azure-deployment

Deploying: To provision Azure resources and deploy the application code, first log in to an Azure account with “azd auth login”. Create a new azd environment using “azd env new”, entering a name for the resource group. This creates a new folder in the .azure folder and sets it as the active environment for future azd commands. Optionally, customize the deployment by setting environment variables to use existing resources, enable optional features (such as auth or vision), or deploy to free tiers. Finally, run “azd up” to provision Azure resources and deploy the sample application, including building the search index based on the files in the ./data folder.

This sample app demonstrates a few approaches for creating ChatGPT-like experiences over the personalized datasets using the Retrieval Augmented Generation pattern. It uses Azure OpenAI Service to access a GPT model (gpt-35-turbo), and Azure AI Search for data indexing and retrieval.

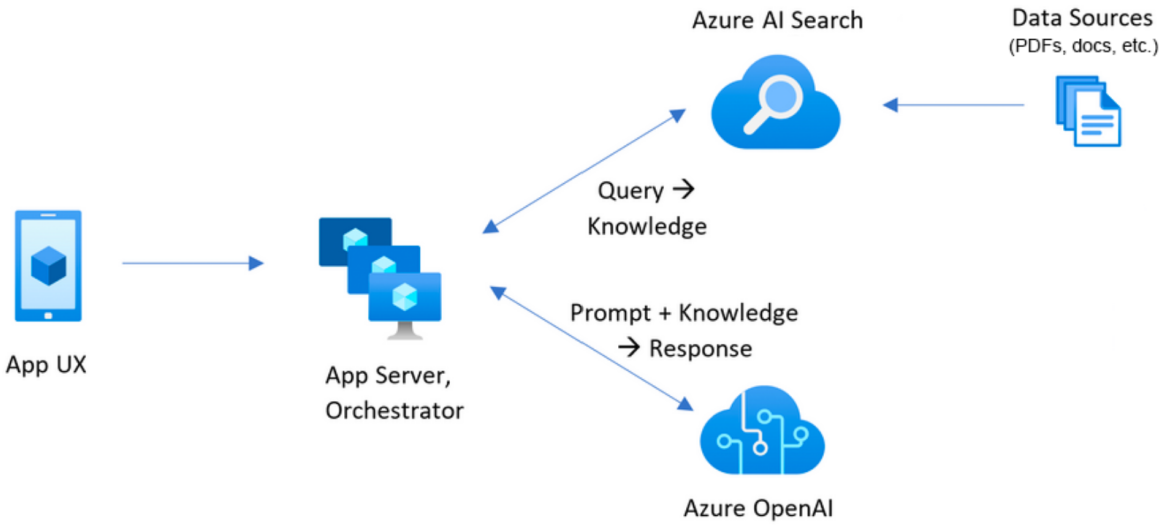

Figure 4. Approach for integrating Azure open AI and Azure AI search services. Adapted from (*GitHub - Azure-Samples/Azure-Search-Openai-Demo: A Sample App for the Retrieval-Augmented Generation Pattern Running in Azure, Using Azure AI Search for Retrieval and Azure OpenAI Large Language Models to Power ChatGPT-Style and Q&A Experiences.*, n.d.)

4.2.1 RAG Based Implementation Through Azure AI Search

The RAG model (explained in Chapter 2), is used to find and generate answers based on data relevant to specific queries or subjects. The process begins with obtaining and extracting source data, typically stored in formats like .pdf and .doc/.docx files, including documents on case law, legislation, and parliamentary discussions. The next step is chunk generation, where the source data is divided into smaller pieces, or chunks. These chunks are then embedded, meaning they are converted into numerical vector representations. This stage involves mapping words or phrases to vectors, using libraries such as OpenAI or GPT-3.5 Turbo.

Creating a vector store is crucial for implementing a RAG-based system, as it allows for the efficient retrieval of relevant passages or documents from a knowledge base. Vector queries are then run to fetch these relevant chunks. During the integration of prompts and search results, the system integrates relevant data and search results based on the prompt questions. The relevant chunks are retrieved from the vector database in accordance with the prompt, which helps in searching for contextually relevant information. These chunks are sent to a large language model (LLM) to aid in the response creation process.

Finally, answer generation uses the retrieved information as a basis to generate response text. At this stage, the type, length, and linguistic style of the generated text can be specified. An LLM, such as Azure OpenAI's GPT-3.5-turbo model, utilizes the similarity search module in Azure AI search to retrieve relevant documents and generate responses.

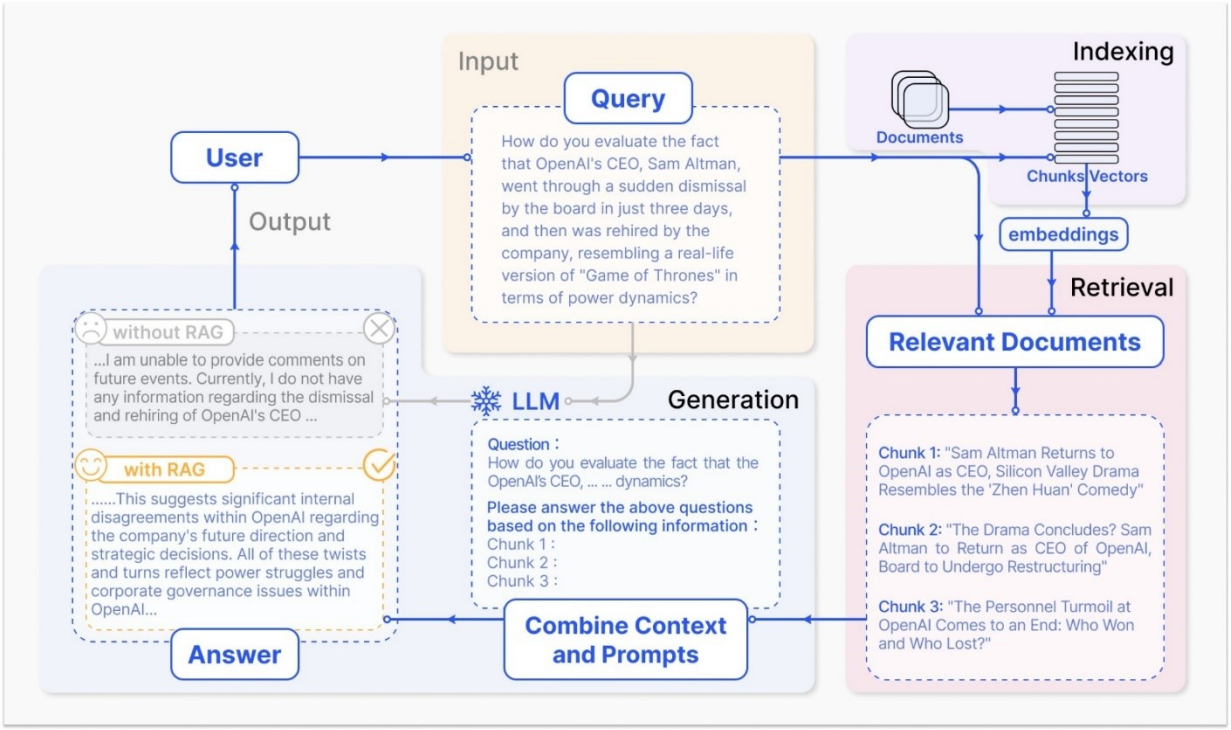

Figure 5. A representative instance of the RAG process applied to question answering, adapted from (Gao et al., 2023)

4.2.2 Source Data Gathering and Data Extraction

We collected the source data from Finlex Finnish Act on Residential Leases en19950481.pdf (finlex.fi) as pdf files. Source: https://www.finlex.fi/en/

Act on the Letting of Residential Apartments 481/1995 - Up-to-date legislation - FINLEX ® and Finnish Consumer Protection Act Kuluttajansuojalaki.engl.lop.doc (finlex.fi) were collected from the webpage and converted to PDF format.

Then Integrated vectorization adds data chunking and text-to-vector embedding to skills in indexer-based indexing. It also adds text-to-vector conversions to queries.

4.2.3 RAG features of the prototype

This research leverages the RAG functionality from the ‘azure-search-openai-demo’ project. The POC (Proof of Concept) directly clones this functionality for testing the prototype.

The "azure-search-openai-demo" application uses a Retrieval-Augmented Generation (RAG) approach, which involves enhancing the capabilities of a language model by incorporating relevant information retrieved from a large dataset. Here are the details of the RAG approach, (Fig. 8) as indicated in the demo application:

RAG Approach: Multiple Approaches

This indicates that the demo application employs several methods to integrate retrieval and generation processes. Multiple approach of RAG can be implemented using different strategies, such as:

Pre-retrieval: Information is retrieved before generating the response. The language model uses this information as context to generate more accurate and relevant responses.

Post-retrieval: The language model generates a response first, and then relevant information is retrieved to validate, correct, or enhance the generated response.

Iterative Retrieval-Generation: The model iteratively alternates between retrieving information and generating text. This approach allows the model to refine its responses based on continuously updated retrieval results.

Key Features in the Context of RAG

Vector support: The demo supports vector-based retrieval, which uses embeddings to find semantically relevant information. This method enhances the retrieval process by identifying contextually similar documents or data points.

Data Ingestion: The demo can ingest data in various formats, making it flexible in terms of the types of information it can incorporate into the retrieval process. This includes text, PDFs, images, etc.

Auth + ACL: Authentication and Access Control Lists (ACL) support ensures that the retrieved information adheres to security and privacy guidelines, which is crucial for applications dealing with sensitive data.

Limitations in the current setup

Persistent chat history: The demo does not support persistent chat history, meaning it cannot maintain context across different user sessions unless the browser tab remains open.

User feedback: The lack of a user feedback mechanism means the model cannot learn from user interactions to improve future responses directly within the demo setup.

In summary, the demo application leverages a RAG approach using multiple strategies to enhance the capabilities of its language model, integrating robust retrieval mechanisms (like vector support) and extensive data ingestion capabilities. However, it currently lacks persistent chat history and user feedback features, which could further refine its performance and user experience.

Feature	azure-search-openai-demo
RAG approach	Multiple approaches
Vector support	✅ Yes
Data ingestion	✅ Yes (Many formats)
Persistent chat history	❌ No (browser tab only)
User feedback	❌ No
GPT-4-vision	✅ Yes
Auth + ACL	✅ Yes
User upload	✅ Yes

Figure 6. RAG features of the azure-search-openai-demo application which is adopted as the development ground for POC of the Prototype. Screenshot collected from https://github.com/Azure-Samples/azure-search-openai-demo/blob/main/docs/other_samples.md (*Azure-Search-Openai-Demo/Docs/Other_samples.Md at Main · Azure-Samples/Azure-Search-Openai-Demo · GitHub*, n.d.)

5 Experiment: Testing popular LLMs and testing the prototype

This chapter provides an objective analysis of the experiments conducted to assess the performance of the POC along with the three LLMs. First we discuss the overview of the three LLMs and make a comparison of the four Gen-AI models, An in-depth examination of the setup, datasets, and performance metrics is provided. Perform comparative prompt and response generation on both RAG optimized Azure OpenAI LLM and non-RAG GenAI environment, such as ChatGPT, compare both the results and evaluate. The chapter's primary objective is to prepare the reader for accurate interpretation of results following an understanding of the research process.

5.1 Overview of the AI powered chatbots chosen for evaluation

In this experiment, we have selected three popular AI language models to test as a legal virtual assistant available today: ChatGPT 3.5, Gemini , and Perplexicity. We also test our prototype based on RAG. All the models are based on similar technologies, LLM. However, their capabilities differ as they use different language models and trained on different knowledge bases on different period of time. Thus, we have four different virtual assistants to evaluate their performance. The performance is evaluated from the perspective of a human expert. The human expert reviews the results provided by each of the models based on the 50 user queries related to housing dispute as mentioned in previous chapter.

Here is a overview of the LLMs to better understand their capabilities and capacities as we evaluate their performance later.

5.1.1 ChatGpt 3.5 Model

ChatGPT 3.5, developed by OpenAI, is a significant advancement in the field of artificial intelligence, particularly in natural language processing. This model is based on the Generative Pre-trained Transformer (GPT) architecture, a sophisticated neural network design that excels at understanding and generating human-like text. As an improved version of the GPT-3 model, ChatGPT 3.5 uses deep learning techniques to deliver impressive performance in various applications.(*What Is ChatGPT?* | *OpenAI Help Center*, n.d.)

One of the model's key capabilities is its ability to comprehend and generate coherent text across a diverse array of topics (*Generative AI - ChatGPT-3.5*, n.d.). It can engage in multi-turn conversations, making it highly suitable for chatbots, virtual assistants, and customer service tools. Additionally, ChatGPT 3.5 is adept at completing text prompts, assisting with tasks such as drafting emails, writing articles, and even generating code (*Generative AI - ChatGPT-3.5*, n.d.; Ray, 2023). Its abilities extend to basic language translation, creative content generation, and providing information

based on its extensive training data up to September 2021 (*What Is ChatGPT?* | *OpenAI Help Center*, n.d.).

The strengths of ChatGPT 3.5 are evident in its versatility, fluency, and adaptability. It can handle a wide range of natural language processing tasks, producing text that is often indistinguishable from human writing. The model's ability to adjust its tone and style based on input makes it highly adaptable for different contexts and user needs. Furthermore, its efficient processing speed enables real-time applications (*ChatGPT0-3.5: An Overview and Limitations* | *Blocshop*, n.d.; Ray, 2023).

Despite its strengths, the model has several weaknesses. Its knowledge is limited to information available up to September 2021, meaning it lacks awareness of subsequent events and developments (*What Is ChatGPT?* | *OpenAI Help Center*, n.d.). Additionally, ChatGPT 3.5 can exhibit biases present in its training data, potentially leading to biased or inappropriate responses. While it generates text that sounds plausible, it may not always be factually accurate and can sometimes produce incorrect information or fabricate details. The quality of its output is also highly dependent on the clarity and specificity of the input it receives (*ChatGPT0-3.5: An Overview and Limitations* | *Blocshop*, n.d.; *What Is ChatGPT?* | *OpenAI Help Center*, n.d.; Ray, 2023).

Note: Our prototype is also built on this GPT 3.5 model family’s variation GPT-3.5 Turbo (*Azure OpenAI Service Models - Azure OpenAI* | *Microsoft Learn*, n.d.).

5.1.2 Gemini 1.0

Gemini 1.0 LLM, developed by Google AI, is a sophisticated tool in the field of natural language processing. This model is built using a transformer-based neural network architecture, similar to other advanced LLMs, which allows it to handle a variety of complex language tasks with impressive proficiency. The specific date Gemini 1.0 was last trained on has not been publicly disclosed by Google.

Gemini 1.0 offers several key capabilities similar to other advanced language models (LLM). It excels in text generation, supports text translation across a broad range of languages, and is adept at answering questions in an informative manner. Gemini also has code generation capabilities for various programming languages. However, there are some limitations associated with the free version of Gemini 1.0. Access may be restricted, and users might encounter quotas on usage, limiting the extent to which they can leverage the model's capabilities. Like other AI models, Gemini can also exhibit biases based on the data it was trained on, which could affect the quality and objectivity of its outputs. Additionally, running large language models requires significant computational resources, which might be constrained in the free tier.

5.1.3 Perplexity AI

Perplexity AI is not a single large language model (LLM) itself, but a company that serves as a gateway. They provide access to various cutting-edge LLMs through their API. Although the specific models powering their free version remain undisclosed, they likely utilize similar technologies and offer comparable functionalities such as text generation, translation, question answering, and summarization (*Perplexity AI Key Features*, n.d.). There may be slight variations in advantages and disadvantages depending on the underlying LLM, but the overall experience should be familiar to those who have used other large language models. (*Perplexity AI Key Features*, n.d.; *Perplexity AI: What You Need to Know and How to Use It | by Entrustech Inc | Medium*, n.d.)

POC, the prototype uses the OpenAI GPT 3.5 Turbo, however, it is enhanced by the Azure AI search and can retrieve the available source data that was added to it's database, related to Finnish housing acts and case laws. Details of data sources are provided in the Appendix section.

Table 1. Comparison of the AI chatbots in experiment.

Feature	ChatGPT 3.5	Gemini	Perplexity	Prototype (Azure OpenAI 3.5 Turbo + RAG)
Provider	OpenAI	Google AI	Perplexity AI	Microsoft Azure OpenAI
LLM Model	GPT-3.5	Transformer-based model	Uses API's from multiple LLM	OpenAI GPT-3.5 Turbo
Optimization Technique	None	None	None	Retrieval-Augmented Generation (RAG) by Azure AI search
Focus	Chat and general text completion	Likely similar to ChatGPT 3.5	Similar to ChatGPT 3.5	Legal domain (Housing Disputes in Finland), GPT 3.5
Technology	Deep Learning	Deep Learning	Deep Learning	Deep Learning + Information Retrieval
Capabilities	Text generation + Machine translation +Question answering (limited)	Text generation + Machine translation +Question answering (limited)	Text generation + Machine translation +Question answering (limited)	Text generation + Access to few Finnish legal resources through RAG _ Question answering focused on housing disputes
Strengths	High fluency and coherence in generated text,Performs well on general tasks	High fluency and coherence in generated text, creativity, comparatively newer data provided than GPT 3.5	Fluency and Coherence	Information Retrieval Capabilities for added source data.
Weaknesses	Limited factual accuracy +Not optimized for specific domains	Limited factual accuracy +Not optimized for specific domains	Limited factual accuracy Not optimized for specific domains	May require more training data for specific legal nuances
Availability	Free version	Free version	Free version	Requires Azure subscription

5.2 Experiment intentions

As described in the previous chapter, a total of 50 questions/prompts are formulated to reflect real-life dispute scenarios commonly occurred. The prompts include human generated errors, misspellings, and grammatical mistakes generated from a non-English speaker perspective. Questions patterns are verified by a Legal professional so that they reflect the commonly occurred dispute scenarios in housing domain in Finland.

Intentions behind the experiments

This experiment is aimed to shed light on the real-world performance of the RAG-optimized GPT-3.5 turbo LLM model by testing it in a housing dispute scenario specific to Finland. The evaluation employs both quantitative metrics and qualitative user feedback (detailed in the previous chapter) to comprehensively assess the LVM's and other LLM's effectiveness in generating response. Additionally, a comparative analysis using baseline metrics and existing models will be conducted to understand the advantages of the RAG-optimized GPT-3.5 turbo LLM as a Legal Virtual Assistant.

5.3 Execution and results

Each of the 50 prompts are submitted to the POC's user interface as well as submitted to the other models at the same time. Responses provided by all the four chatbot's are recorded, copied to their corresponding column and question in the excel sheet of experimental set up.

Quantitative results:

Except for our prototype, each of the other three chatbots provided 50 responses, each offering some kind of solution related to the query, regardless of its accuracy. Our prototype, however, suggested solutions to 46 queries and failed to provide solutions for 4 queries, citing a lack of information from the provided sources. The four instances where the prototype could not generate a solution were all related to the dispute type "cannot pay the rent." Specifically, three of these instances involved scenarios where the tenant was seeking unemployment benefits, and the other involved a medical emergency preventing the tenant from paying rent.

In terms of reference generation, the ChatGPT 3.5 model did not provide any references or website links for any of the 50 instances. On the other hand, Gemini and Perplexicity included references and web links in their responses, while the prototype provided references for 46 of the instances where it generated responses with solutions. In this quantitative analysis, we are not

taking into account the accuracy of the responses or the relevance and validity of the references. These aspects will be discussed in a later chapter.

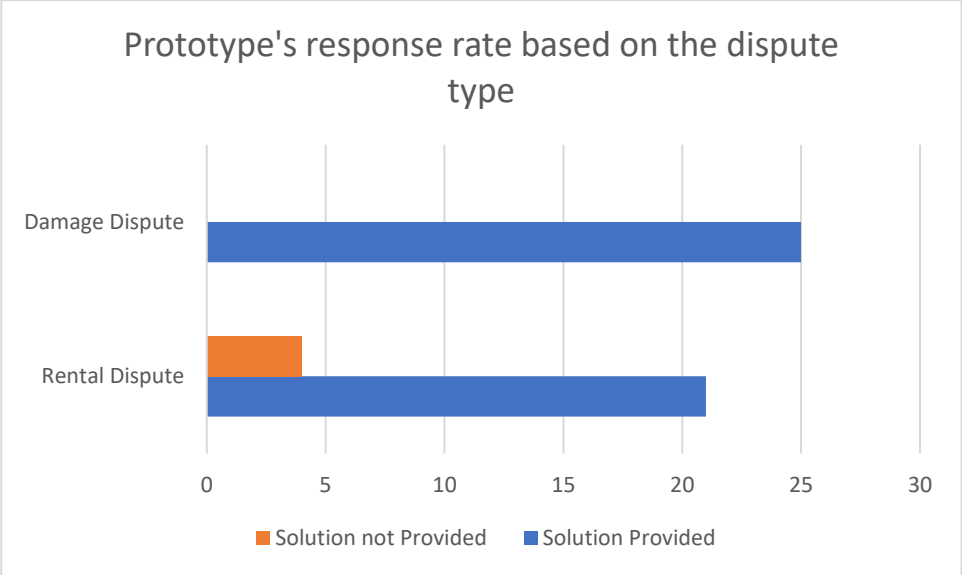

Figure 7. Prototype’s response rate based on the dispute type

Quantitatively, the prototype's ability to provide solutions to 46 out of 50 queries indicates a success rate of 92%. However, this still lags behind the other models, which achieved a 100% success rate in generating solutions. The prototype's inability to respond to 4 queries due to a lack of information highlights a potential area for improvement in its knowledge base or the integration of external data sources. This limitation might also suggest inefficiencies in retrieving data, as solutions were provided for very similar reformulations of the same query. The fact that the prototype did not provide information outside its sources may indicate an effort to ensure the validity of the information it offers, adhering strictly to the available data.

Reference and Link Provision:

ChatGPT 3.5's lack of references or website links could imply either a different design philosophy prioritizing direct answers or limitations in its capability to provide sourced information.

Gemini, Perplexicity, and the prototype's inclusion of references and links enhance the credibility and utility of their responses, making them more robust for users seeking verifiable information.

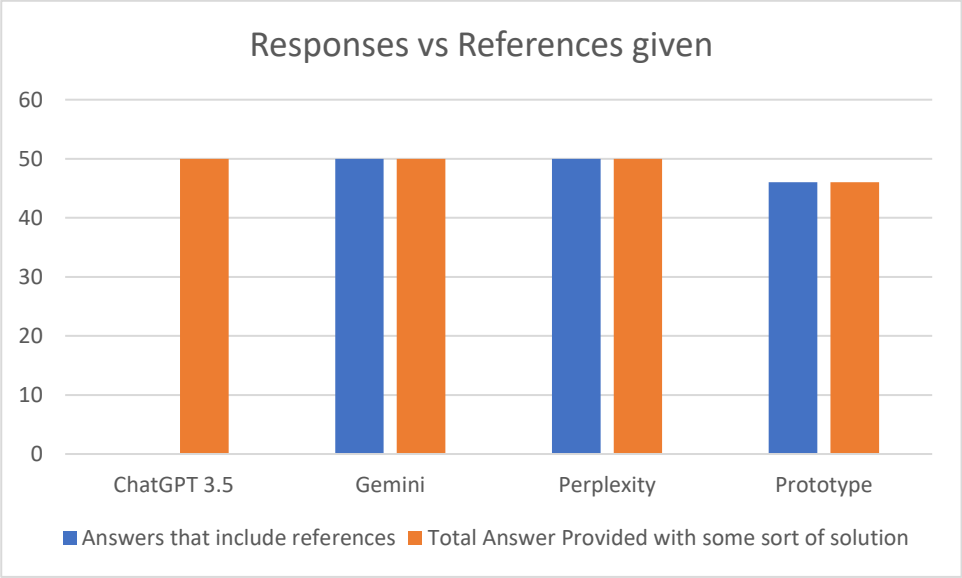

Figure 8. Responses vs references given.

Comparative Strengths:

Prototype's performance in providing references for 46 instances closely aligns with the prototype's ability to generate solutions, though prototype offers better support through its references. In contrast, Gemini and Perplexicity distinguish themselves by delivering comprehensive responses accompanied by supporting references, making them more reliable and preferable for users who require documented sources.

5.3.1 Qualitative analysis from feedback

General Impressions

As we collected feedback from a human expert, general impression of the responses from ChatGPT, Gemini, Perplexity, and the prototype is notably critical. He points out that none of the models provided meaningful advice or guidance. Although the responses might appear impressive at first glance, a deeper examination reveals that they lack significant value and practical impact. The advice is basic and common sense, failing to offer users concrete steps to resolve their specific legal issues. With a simple internet search, better information and direction can be found as per the opinion of the human expert. This issue is compounded by inconsistencies and adjustments in response formatting for similar queries in different words, which contribute to the responses' lack of reliability.

Evaluation on Correctness of the Models

Factual Accuracy: Expert finds it impossible to measure the accuracy of the responses as all models consistently provide misleading information. Each model's responses vary for similar queries, suggesting a fundamental issue in factual consistency. For example, where a tenant does not have the money to pay the rent, one of the LLMs is suggesting paying for due rent on top of current month’s rent which absolutely does not make any sense.

Source Reliability: The reliability of sources is another major concern. ChatGPT 3.5 does not provide any references, while Gemini and Perplexity offer some, though these are often unreliable. For instance, Perplexity cited French data for Finnish housing queries and used Reddit comments, which are not credible sources at all. Prototype predominantly relied on its database of PDF files, which were often irrelevant or misinterpreted, leading to misleading advice. For example, Prototype erroneously referenced an event from 2007 unrelated to the user's current situation, highlighting a critical flaw in understanding and contextualizing information.

We analyzed the data and identified 6 instances where prototype completely misinterpreted the source data of case laws. For example (Misinterpretation 4, prompt number 19 from the .xlsx file for experimental setup), a source data illustrates a legal case for damage dispute happened in 2010. The prototype is retrieving this incident and providing solution as if the issue is related to the user who submitted the prompt.

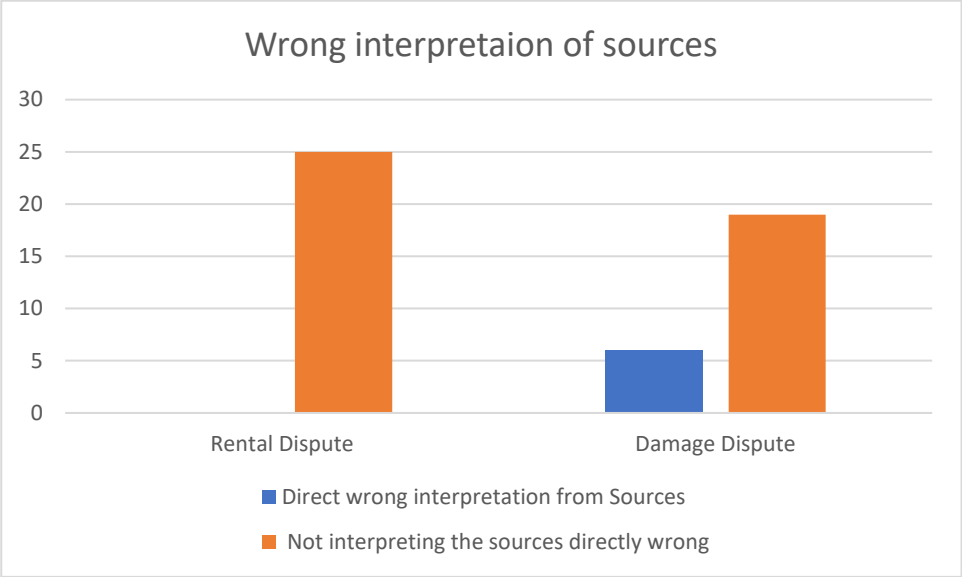

Figure 9. RAG-based prototype interprets wrongly for damage disputes.

Internal Consistency: As per the expert Judgement, Internal consistency exist within multiple responses and it is problematic across all models. They provide different answers to the same reformulated questions, with less variation in ChatGPT and Gemini compared to Perplexity and the prototype. This inconsistency undermines the reliability of the solution provided.

Evaluation on completeness of the Models

The responses generally lack depth and fail to cover all relevant aspects of the queries. The advice given is basic and does not guide the user on how to solve their problem with housing related matters at all. There are some generalized steps such as, “Communicate with the land lord” are really basic and does not help much.

The responses lack sufficient detail and certainty. On closer inspection, they appear empty and do not provide actionable solutions. This lack of detail fails to meet users' needs for precise and impactful legal advice. For instance, the right to housing support (Kela) is a legal right in Finland, but none of the models clearly communicated this. Gemini provided a link to Kela but did not emphasize its legal significance.

The relevance of the information is another area of concern. The answers, while sometimes touching on relevant points like health hazard inspections, do not emphasize their legal importance.

It is evident that none of the models provided meaningful or reliable advice, despite initially appearing impressive. The lack of concrete steps to resolve legal issues, coupled with inconsistencies in response formatting and factual inaccuracies, highlights significant shortcomings in the models' performance. Moreover, concerns regarding source reliability and the completeness, detail sufficiency, and relevance of information further underscore the limitations of current AI-powered chatbots in providing comprehensive legal guidance. This evaluation serves as a sobering reminder of the complexities involved in developing AI systems capable of offering reliable and actionable legal advice, emphasizing the need for continued research and refinement in this area.

6 Discussion

6.1 Research Question 1

How does the integration of LLMs optimized by RAG enhance the performance of a Legal Virtual Assistant in resolving housing disputes in Finland?

Current research hasn't provided a definitive answer on the effectiveness of RAG-based LVA in resolving housing disputes in Finland.

The main limitation lies in the result of the experiment. It didn't compare the performance of the RAG-based LVA prototype to models without RAG. This makes it impossible to isolate the impact of RAG technology and assess its true contribution.

Although, the qualitative data suggests that the prototype retrieved information from data sources and generated responses, the extent of its effectiveness and the potential limitations of RAG remain unclear. Factors like the amount of source data and fine-tuning of the model require further investigation.

Furthermore, all the LLMs evaluated, including the RAG prototype, struggled with understanding the legal aspects of housing disputes from user queries. This highlights broader issues with LLMs in legal domains, such as NLP techniques in understanding complex legal settings. Furthermore, there have been doubts regarding the factual accuracy and completeness of responses provided by LLMs.

To conclude the answer, the research doesn't provide enough evidence to definitively say whether RAG-optimized LLMs enhance LVAs for housing disputes. It's possible that even with successful data retrieval by RAG, the LLM itself might not be able to interpret the legal information and generate accurate responses. Further research is needed, potentially focusing on evaluating RAG in isolation from LLMs to understand its true potential in legal contexts.

6.2 Research Question 2

Does the RAG optimization significantly enhance the factual accuracy of the Large Language Model (LLM) responses when compared to a non-optimized LLM in the context of housing disputes in Finland?

Based on the analysis conducted in this study, the effectiveness of Retrieval Augmented Generation (RAG) optimization in enhancing the factual accuracy of Large Language Model (LLM) responses in the context of housing disputes in Finland appears to be limited. While RAG

optimization theoretically allows LLMs to dynamically retrieve relevant external information to augment their responses, the practical implementation of RAG in our study did not demonstrate a significant improvement in factual accuracy when compared to a non-optimized LLM. Both RAG-optimized and non-optimized LLMs struggled with accuracy issues, providing misleading information, misinterpreting user queries, and failing to retrieve relevant legal data effectively. Therefore, while RAG optimization holds promise for improving factual accuracy in LLM responses, its actual impact may be constrained by various challenges, including data source limitations, complexity of legal language, and implementation challenges. Further research and development efforts are needed to address these limitations and fully realize the potential of RAG optimization in enhancing the factual accuracy of LLM responses in the context of housing disputes in Finland.

6.3 Research Question 3

What are the key challenges and limitations encountered when employing LLMs optimized by RAG within the domain of housing disputes in Finland?

This study exploring the use of Large Language Models (LLMs) optimized by Retrieval-Augmented Generation (RAG) for Finnish housing disputes revealed several challenges. The key issue identified was ensuring the accuracy and reliability of LLM responses. Despite RAG's integration, responses often lacked factual grounding due to limitations in the LLM models themselves. This undermines the trustworthiness of their legal advice and reduces their usefulness. The research also highlights the broader limitations of current LLM technology in the legal domain.

Legal language's inherent complexity poses another hurdle. LLMs struggle to interpret and respond in a way that captures the intricacies of Finnish housing law, potentially leading to oversimplified or incorrect advice. LLMs trained on generalized datasets lack the depth and specificity required for legal use, often providing inadequate and superficial responses. Legal advice necessitates a high level of detail and precision, which current LLM capabilities may not adequately fulfill. Moreover, there is a misalignment between the broad training data and the specific legal context of queries, resulting in difficulties prioritizing legally significant information. Capturing the nuances of legal rights and obligations is challenging for LLMs, further limiting their effectiveness in conveying actionable legal guidance.

The effectiveness of RAG heavily depends on the quality and relevance of its data sources. In this study's context, comprehensive and up-to-date Finnish legal databases were limited. Training LLMs on vast amounts of data is another challenge, further restricted by language barriers between Finnish and English legal documents.

Implementing RAG effectively requires advanced techniques and extensive fine-tuning to ensure it retrieves and utilizes relevant information accurately. The study suggests that current implementations lacked sufficient fine-tuning, especially for legal applications. This may have limited the model's ability to handle specific legal terminology, contexts, and jurisdictional nuances.

Finally, using RAG through advanced LLMs can be expensive. The computational resources and infrastructure needed, such as high-performance servers or cloud computing, contribute significantly to the cost. Ongoing maintenance, support, and customization add to the financial burden, requiring organizations to budget for software updates, technical support, and integration expenses.

6.4 Recommendations, Future Suggestions and opinions

To truly leverage the benefits of RAG, it is crucial not only to retrieve relevant data but also to interpret and utilize this data effectively, thereby improving both Natural Language Processing (NLP) and Natural Language Generation (NLG) capabilities. Continuous learning and fine-tuning of LLMs for specific domains, such as Finnish housing law, are essential steps forward.

Focusing on the nuances of the Finnish legal language is imperative. Enhancements in the system through techniques such as prompt engineering can also significantly improve performance. Incorporating expert opinions and user feedback into the development process ensures that the models align better with real-world requirements and user expectations. Investing in more advanced LLMs, such as the latest versions available, will provide a stronger foundation for further improvements. Continuous training with updated and comprehensive legal data will help maintain the relevance and accuracy of the models.

RAG's capability to retrieve specific information from targeted documents needs to be refined further. Models should be equipped to ask counter-questions, enabling a deeper understanding of the user’s legal situation. This interactive capability will allow the LLMs to gather more context-specific information, leading to more precise and reliable legal advice.

RAG’s integration with LLMs necessitates a multifaceted strategy involving continuous improvement in NLP and NLG, domain-specific fine-tuning, advanced techniques, and robust infrastructure investments. These steps are vital to harness the full potential of LLMs in addressing complex legal queries effectively.

The complicated nature of legal services presents significant challenges for LLM technologies, which are not yet capable of fully comprehending user queries. Legal consultation requires a detailed understanding of all applicable legislation, case laws, and the specific circumstances of each case. It is not merely about generating responses based on user queries but involves a deep, nuanced understanding of legal contexts and implications.

The limitation of having only one expert viewpoint poses a challenge in establishing a universally applicable approach to providing solutions to users' legal queries and guiding them in the right direction. Additionally, the prototype's inability to fully utilize Retrieval-Augmented Generation (RAG) due to constraints of time, resources, and expertise further complicates the matter. From the perspective of a developer, I faced difficulties in determining the optimal extent to which the prototype reads from data sources and utilizes the external knowledge base of Large Language Models (LLMs). Moreover, LLMs' current limitations in distinguishing between individual case laws, extracting insights from past cases, and understanding the context and situation of specific users highlight the extensive progress required before AI-powered chatbots can autonomously provide comprehensive legal solutions. Nevertheless, this study serves as a foundation for identifying improvement criteria and underscores the necessity of combining multiple techniques and technologies to enhance the accuracy and completeness of legal AI assistants. Taking incremental steps, such as targeting specific areas of legal work procedures, can pave the way for AI to offer quick, easy, and effective information retrieval from large databases, ultimately facilitating the integration of traditional legal practices with AI advancements. By addressing these challenges, AI has the potential to evolve and provide more tailored and accurate guidance or insights to users seeking legal advice.

7 Conclusion

This thesis aimed to evaluate the feasibility of integrating Retrieval-Augmented Generation (RAG) with Large Language Models (LLMs) to create an AI-based assistant capable of resolving housing disputes in Finland. However, through a comprehensive experimental setup, we concluded that comparing the performance of our RAG-optimized prototype against other LLMs was not viable. This was due to all the LLM models' inability to provide accurate legal advice, thus failing to meet the criteria for correctness, completeness, and practical applicability. The study revealed that while the integration of RAG was intended to enhance factual accuracy and relevance, the actual improvement observed was marginal. Both RAG-optimized and non-optimized LLMs struggled to offer significantly better legal advice, often providing misleading or irrelevant information that undermined their utility.

Several limitations were identified, including the generalized nature of the training data, which lacked the specific legal context necessary for detailed and accurate legal advice. There was a significant misalignment between the training data of the LLM models and the specific legal scenarios encountered in housing disputes, leading to inadequate and sometimes misleading responses. Additionally, the technical implementation of RAG did not significantly improve the models' ability to retrieve and utilize relevant legal information effectively, a challenge exacerbated by the complexity of legal language and the specific nuances of Finnish law. Resource constraints also played a role, as the computational requirements for fine-tuning and running advanced LLMs with RAG capabilities were substantial, making it a costly endeavor.

Future research should focus on acquiring more comprehensive and domain-specific legal datasets, potentially through collaborations with legal institutions to access proprietary data. Advanced fine-tuning techniques and prompt engineering can help align the models more closely with the legal context of queries. Investigating the integration of multimodal data, such as text, images, and audio, can provide a richer contextual understanding and improve the accuracy of legal responses. Conducting user-centric studies to gather feedback from legal professionals and end-users will help identify practical usability issues and guide iterative improvements. Exploring the ethical and legal implications of deploying AI-driven legal assistants is crucial to ensuring data privacy, mitigating biases, and maintaining regulatory compliance.

In conclusion, while the integration of RAG with LLMs shows promise, significant challenges remain in achieving the level of accuracy and specificity required for practical legal use. Addressing these limitations through targeted research and development can pave the way for more effective and reliable legal virtual assistants, ultimately enhancing access to legal resources and support for individuals facing housing disputes in Finland.

References

5 Ways SmartRent UK is Revolutionising Smart Home Tech | SmartRent. (n.d.). Retrieved May 17, 2024, from https://smartrent.com/news/5-ways-smartrent-uk-is-revolutionising-smart-home-tech/

7 best legal AI chatbots for 2024. (n.d.). Retrieved May 16, 2024, from https://juro.com/learn/legal-ai-chatbot#

Abd-alrazaq, A. A., Alajlani, M., Alalwan, A. A., Bewick, B. M., Gardner, P., & Househ, M. (2019). An overview of the features of chatbots in mental health: A scoping review. *International Journal of Medical Informatics*, *132*, 103978. https://doi.org/10.1016/J.IJMEDINF.2019.103978

Adamopoulou, E., & Moussiades, L. (2020). Chatbots: History, technology, and applications. *Machine Learning with Applications*, *2*, 100006-. https://doi.org/10.1016/j.mlwa.2020.100006

Alan, A. Y., Aydın, Ö., & Karaarslan, E. (2024). A RAG-based Question Answering System Proposal for Understanding Islam: MufassirQAS LLM. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4707470

Alavandhar, J. V., & Ņikiforova, O. (2017). Several Ideas on Integration of SCRUM Practices within Microsoft Solutions Framework. *Applied Computer Systems*, *21*(1), 71–79. https://doi.org/10.1515/acss-2017-0010

Al-Hasan, T. M., Sayed, A. N., Bensaali, F., Himeur, Y., Varlamis, I., & Dimitrakopoulos, G. (2024). From Traditional Recommender Systems to GPT-Based Chatbots: A Survey of Recent Developments and Future Directions. *Big Data and Cognitive Computing*, *8*(4), 36-. https://doi.org/10.3390/bdcc8040036

Aslam, F. (2023). The Impact of Artificial Intelligence on Chatbot Technology: A Study on the Current Advancements and Leading Innovations. *European Journal of Technology*, *7*(3), 62–72. https://doi.org/10.47672/EJT.1561

Attigeri, G., Agrawal, A., & Kolekar, S. V. (2024). Advanced NLP Models for Technical University Information Chatbots: Development and Comparative Analysis. *IEEE Access*, *12*, 29633–29647. https://doi.org/10.1109/ACCESS.2024.3368382

Azure OpenAI Service models - Azure OpenAI | Microsoft Learn. (n.d.). Retrieved May 20, 2024, from https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/models

azure-search-openai-demo/docs/other_samples.md at main · Azure-Samples/azure-search-openai-demo · GitHub. (n.d.). Retrieved May 17, 2024, from https://github.com/Azure-Samples/azure-search-openai-demo/blob/main/docs/other_samples.md

Bratić, D., Šapina, M., Jurečić, D., & Žiljak Gršić, J. (2024). Centralized Database Access: Transformer Framework and LLM/Chatbot Integration-Based Hybrid Model. *Applied System Innovation*, *7*(1), 17-. https://doi.org/10.3390/asi7010017

Chan, B. (2020). German's Next Language Model. *ArXiv.Org*. https://doi.org/10.48550/arxiv.2010.10906

Chatbot use cases: 25 real-life examples. (n.d.). Retrieved May 17, 2024, from https://leaddesk.com/blog/chatbot-use-cases-25-real-life-examples/

ChatGPT, Generative AI, LLM, NLP: how to understand the new era of artificial intelligence already impacting businesses - ProQuest. (n.d.). Retrieved May 16, 2024, from https://www.proquest.com/docview/2791107911/citation/15D2E2C3EBA149B7PQ/1?accountid=27436&sourcetype=Wire%20Feeds

ChatGPT0-3.5: An Overview and Limitations | Blocshop. (n.d.). Retrieved May 20, 2024, from https://www.blocshop.io/blog/chatgpt3-5-limitations

Contract summary: what it is and how to create one. (n.d.). Retrieved May 16, 2024, from https://juro.com/learn/contract-summary#

Eshbayev, O. A., Mirzaliev, S. M., Rozikov, R. U., Kuzikulova, D. M., & Shakirova, G. A. (2022). NLP and ML based approach of increasing the efficiency of environmental management operations and engineering practices. *IOP Conference Series. Earth and Environmental Science*, *1045*(1), 12058-. https://doi.org/10.1088/1755-1315/1045/1/012058

Evolution of Legal AI from Extractive to Generative - The CaseMine Story. (n.d.). Retrieved May 17, 2024, from https://www.barandbench.com/news/evolution-of-legal-ai-from-extractive-to-generative-the-casemine-story

Filonova, E. (2022). *Evaluation of Natural Language Processing and Machine Learning Tools for the Automation of the Customer Service Task*.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023). *Retrieval-Augmented Generation for Large Language Models: A Survey*. http://arxiv.org/abs/2312.10997

Gargiulo, F., Minutolo, A., Guarasci, R., Damiano, E., De Pietro, G., Fujita, H., & Esposito, M. (2022). An ELECTRA-Based Model for Neural Coreference Resolution. *IEEE Access*, *10*, 75144–75157. https://doi.org/10.1109/ACCESS.2022.3189956

Generative AI - ChatGPT-3.5. (n.d.). Retrieved May 20, 2024, from https://www.w3schools.com/gen_ai/gen_ai_chatgpt-3-5.php

Generative AI for Professional Services | *Harvey*. (n.d.). Retrieved May 17, 2024, from https://www.harvey.ai/

GitHub - Azure-Samples/azure-search-openai-demo: A sample app for the Retrieval-Augmented Generation pattern running in Azure, using Azure AI Search for retrieval and Azure OpenAI large language models to power ChatGPT-style and Q&A experiences. (n.d.). Retrieved May 17, 2024, from https://github.com/Azure-Samples/azure-search-openai-demo

Hamid, A. A., Nurul, S., & Kamal, H. (2023). PKS IVAStar - An Overview of Chatbot Development. *Borneo Engineering & Advanced Multidisciplinary International Journal*, *2*(Special Issue (TECHON 2023)), 122–127. https://beam.pmu.edu.my/index.php/beam/article/view/110

Harvey | *OpenAI*. (n.d.). Retrieved May 17, 2024, from https://openai.com/index/harvey/

Harvey AI: Legal Artificial Intelligence. (n.d.). Retrieved May 17, 2024, from https://www.clio.com/blog/harvey-ai-legal/

Hoas - Hoas. (n.d.). Retrieved May 17, 2024, from https://hoas.fi/en/hoas/

Hoppe, C., Migenda, N., Pelkmann, D., Hötte, D., & Schenck, W. (2022). Collaborative System for Question Answering in German Case Law Documents. *IFIP Advances in Information and Communication Technology*, *662 IFIP*, 303–312. https://doi.org/10.1007/978-3-031-14844-6_24

Housing Chatbot improves the overall customer satisfaction | *GetJenny*. (n.d.). Retrieved May 17, 2024, from https://www.getjenny.com/housing-chatbot-improves-the-overall-customer-satisfaction

Introduction to Azure AI Search - Azure AI Search | *Microsoft Learn*. (n.d.). Retrieved May 20, 2024, from https://learn.microsoft.com/en-us/azure/search/search-what-is-azure-search

Işıkdemir, Y. E. (2024). NLP TRANSFORMERS: ANALYSIS OF LLMS AND TRADITIONAL APPROACHES FOR ENHANCED TEXT SUMMARIZATION. *Eskişehir Osmangazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi*, *32*(1). https://doi.org/10.31796/ogummf.1303569

Kang, M., Gürel, N. M., Yu, N., Song, D., & Li, B. (2024). *C-RAG: Certified Generation Risks for Retrieval-Augmented Language Models*. http://arxiv.org/abs/2402.03181

Kettunen, H., & Ruonavaara, H. (2015). Discoursing deregulation: the case of the Finnish rental housing market. *International Journal of Housing Policy*, *15*(2), 187–204. https://doi.org/10.1080/14616718.2014.990774

Khadija, M. A., Aziz, A., & Nurharjadmo, W. (2023). Automating Information Retrieval from Faculty Guidelines: Designing a PDF-Driven Chatbot powered by OpenAI ChatGPT. *Proceedings - 2023 10th International Conference on Computer, Control, Informatics and Its Applications: Exploring the Power of Data: Leveraging Information to Drive Digital Innovation, IC3INA 2023*, 394–399. https://doi.org/10.1109/IC3INA60834.2023.10285808

Koga, S., & Du, W. (2024). Integrating AI in medicine: Lessons from Chat-GPT's limitations in medical imaging. *Digestive and Liver Disease*. https://doi.org/10.1016/J.DLD.2024.02.014

Koubaa, A., Qureshi, B., Ammar, A., Khan, Z., Boulila, W., & Ghouti, L. (2023). Humans are still better than ChatGPT: Case of the IEEEXtreme competition. *Heliyon*, *9*(11), e21624–e21624. https://doi.org/10.1016/j.heliyon.2023.e21624

Kulkarni, A., Shivananda, A., & Kulkarni, A. (2021). Natural Language Processing Projects: Build Next-Generation NLP Applications Using AI Techniques. *Natural Language Processing Projects: Build Next-Generation NLP Applications Using AI Techniques*, 1–317. https://doi.org/10.1007/978-1-4842-7386-9

Lareyre, F., Nasr, B., Chaudhuri, A., Di Lorenzo, G., Carlier, M., & Raffort, J. (2023). Comprehensive Review of Natural Language Processing (NLP) in Vascular Surgery. In *EJVES Vascular Forum* (Vol. 60, pp. 57–63). Elsevier Ltd. https://doi.org/10.1016/j.ejvsvf.2023.09.002

Lecler, A., Duron, L., & Soyer, P. (2023). Revolutionizing radiology with GPT-based models: Current applications, future possibilities and limitations of ChatGPT. *Diagnostic and Interventional Imaging*, *104*(6), 269–274. https://doi.org/10.1016/J.DIII.2023.02.003

Lee, J., An, T., Chu, H. E., Hong, H. G., & Martin, S. N. (2023). Improving Science Conceptual Understanding and Attitudes in Elementary Science Classes through the Development and Application of a Rule-Based AI Chatbot. *Asia-Pacific Science Education*, *9*(2), 365–412. https://doi.org/10.1163/23641177-bja10070

Leveraging Artificial Intelligence for Property Management - Kurby Real Estate AI. (n.d.). Retrieved May 17, 2024, from https://blog.kurby.ai/leveraging-artificial-intelligence-for-property-management/

Li, J., Yuan, Y., & Zhang, Z. (2024). *Enhancing LLM Factual Accuracy with RAG to Counter Hallucinations: A Case Study on Domain-Specific Queries in Private Knowledge-Bases*. https://arxiv.org/abs/2403.10446v1

Lin, C. C., Huang, A. Y. Q., & Yang, S. J. H. (2023). A Review of AI-Driven Conversational Chatbots Implementation Methodologies and Challenges (1999–2022). *Sustainability (Basel, Switzerland)*, *15*(5), 4012-. https://doi.org/10.3390/su15054012

Mathis, B. (2022). Extracting Proceedings Data from Court Cases with Machine Learning. *Stats*, *5*(4), 1305–1320. https://doi.org/10.3390/stats5040079

Miao, J., Thongprayoon, C., Suppadungsuk, S., Garcia Valencia, O. A., & Cheungpasitporn, W. (2024). Integrating Retrieval-Augmented Generation with Large Language Models in Nephrology: Advancing Practical Applications. In *Medicina (Lithuania)* (Vol. 60, Issue 3). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/medicina60030445

Min, D., Hu, N., Jin, R., Lin, N., Chen, J., Chen, Y., Li, Y., Qi, G., Li, Y., Li, N., & Wang, Q. (2024). *Exploring the Impact of Table-to-Text Methods on Augmenting LLM-based Question Answering with Domain Hybrid Data*. http://arxiv.org/abs/2402.12869

More Efficient NLP Model Pre-training with ELECTRA. (n.d.). Retrieved May 17, 2024, from https://research.google/blog/more-efficient-nlp-model-pre-training-with-electra/

Murtarelli, G., Gregory, A., & Romenti, S. (2021). A conversation-based perspective for shaping ethical human–machine interactions: The particular challenge of chatbots. *Journal of Business Research*, *129*, 927–935. https://doi.org/10.1016/J.JBUSRES.2020.09.018

Ott, S., Hebenstreit, K., Liévin, V., Hother, C. E., Moradi, M., Mayrhauser, M., Praas, R., Winther, O., & Samwald, M. (2023). ThoughtSource: A central hub for large language model reasoning data. *Scientific Data 2023 10:1*, *10*(1), 1–12. https://doi.org/10.1038/s41597-023-02433-3

Patil, R., & Gudivada, V. (2024). A Review of Current Trends, Techniques, and Challenges in Large Language Models (LLMs). *Applied Sciences*, *14*(5), 2074. https://doi.org/10.3390/app14052074

Pereira, R., Lima, C., Pinto, T., & Reis, A. (2023). Virtual Assistants in Industry 4.0: A Systematic Literature Review. *Electronics 2023, Vol. 12, Page 4096*, *12*(19), 4096. https://doi.org/10.3390/ELECTRONICS12194096

Perplexity AI Key Features. (n.d.). Retrieved May 20, 2024, from https://www.perplexity.ai/page/Perplexity-AI-Key-AQEBigvaS9qAYqE12rkRsg

Perplexity AI: What You Need to Know and How to Use It | by Entrustech Inc | Medium. (n.d.). Retrieved May 20, 2024, from https://medium.com/@entrustech/perplexity-ai-what-you-need-to-know-and-how-to-use-it-82ee6ce1fbd

Pries, K. H., & Quigley, J. M. (2011). *Scrum project management*. CRC Press. http://books.google.com/books?id=Of6JC-1DHIoC&pgis=1

Quali-bot, the virtual assistant that also helps in legal claims issues. (2024). https://www.proquest.com/wire-feeds/quali-bot-virtual-assistant-that-also-helps-

Ray, P. P. (2023). ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, *3*, 121–154. https://doi.org/10.1016/J.IOTCPS.2023.04.003

Rivas, P., & Zhao, L. (2023). Marketing with ChatGPT: Navigating the Ethical Terrain of GPT-Based Chatbot Technology. *AI (Basel)*, *4*(2), 375–384. https://doi.org/10.3390/ai4020019

Schwenke, N., Söbke, H., & Kraft, E. (2023). Potentials and Challenges of Chatbot-Supported Thesis Writing: An Autoethnography. *Trends in Higher Education*, *2*(4), 611–635. https://doi.org/10.3390/higheredu2040037

Shankar, V., & Parsana, S. (2022). An overview and empirical comparison of natural language processing (NLP) models and an introduction to and empirical application of autoencoder models in marketing. *Journal of the Academy of Marketing Science*, *50*(6), 1324–1350. https://doi.org/10.1007/s11747-022-00840-3

Skjuve, M., Følstad, A., Fostervold, K. I., & Brandtzaeg, P. B. (2021). My Chatbot Companion - a Study of Human-Chatbot Relationships. *International Journal of Human-Computer Studies*, *149*, 102601-. https://doi.org/10.1016/j.ijhcs.2021.102601

SmartRent | Smart Home Solutions for Multifamily Communities. (n.d.). Retrieved May 17, 2024, from https://smartrent.com/

SmartRent Delivers Seamless Property Management with Salesforce - Salesforce. (n.d.). Retrieved May 17, 2024, from https://www.salesforce.com/uk/resources/customer-stories/smartrent-delivers-seamless-prop-mgmt/

Ullah, E., Parwani, A., Baig, M. M., & Singh, R. (2024). Challenges and barriers of using large language models (LLM) such as ChatGPT for diagnostic medicine with a focus on digital pathology - a recent scoping review. *Diagnostic Pathology*, *19*(1), 43–43. https://doi.org/10.1186/s13000-024-01464-7

Vasileiou, M. V., & Maglogiannis, I. G. (2022). The Health ChatBots in Telemedicine: Intelligent Dialog System for Remote Support. *Journal of Healthcare Engineering*, *2022*. https://doi.org/10.1155/2022/4876512

Whang, J. Bin, Song, J. H., Lee, J. H., & Choi, B. (2022). Interacting with Chatbots: Message type and consumers' control. *Journal of Business Research*, *153*, 309–318. https://doi.org/10.1016/J.JBUSRES.2022.08.012

What is ChatGPT? | *OpenAI Help Center*. (n.d.). Retrieved May 20, 2024, from https://help.openai.com/en/articles/6783457-what-is-chatgpt

Żmihorski, M. (2023). The hallucinating chatbot 'ChatGPT' poorly estimates real bird commonness. *Biological Conservation*, *288*, 110371. https://doi.org/10.1016/J.BIOCON.2023.110371

Appendices

Appendix 1.1: Application Technologies

Basic Libraries and Setting Azure OpenAI

In the open-source demo application for this POC, the backend approach of the Python code (refer to the figure 07) imports necessary libraries for a project involving Azure and OpenAI integration. It begins by importing essential modules like OS, ABC (Abstract Base Classes), dataclass, and typing for type hints. Then, it imports specific components from aiohttp for asynchronous HTTP requests and from Azure SDK for Python for integrating with Azure's search service. The code also imports specific models for Azure search documents such as QueryCaptionResult and VectorizedQuery. Furthermore, it imports AsyncOpenAI for asynchronous interactions with the OpenAI API. Additionally, the code imports custom modules like Authentication Helper for handling authentication and no newlines from the text module. This setup enables the Python program to leverage Azure's search capabilities (Vector query used for RAG approach) and OpenAI's language processing features (LLM) efficiently while maintaining readability and modularity.

```python
import os
from abc import ABC
from dataclasses import dataclass
from typing import Any, AsyncGenerator, Awaitable, Callable, List, Optional, Union, cast
from urllib.parse import urljoin

import aiohttp
from azure.search.documents.aio import SearchClient
from azure.search.documents.models import (
    QueryCaptionResult,
    QueryType,
    VectorizedQuery,
    VectorQuery,
)
from openai import AsyncOpenAI

from core.authentication import AuthenticationHelper
from text import nonewlines


@dataclass
class Document:
    id: Optional[str]
    content: Optional[str]
    embedding: Optional[List[float]]
    image_embedding: Optional[List[float]]
    category: Optional[str]
    sourcepage: Optional[str]
    sourcefile: Optional[str]
    oids: Optional[List[str]]
    groups: Optional[List[str]]
    captions: List[QueryCaptionResult]
    score: Optional[float] = None
    reranker_score: Optional[float] = None

    def serialize_for_results(self) -> dict[str, Any]:
        return {
```

Figure 10. Back-end approach of demo application- using Python code: importing libraries and integrating Azure OpenAi (LLM) and Vector Search query for RAG method.

The demo uses a multi-step approach that first uses OpenAI to turn the user's question into a search query, then uses Azure AI Search to retrieve relevant documents, and then sends the conversation history, original user question, and search results to OpenAI to generate a response.

```
from typing import Any, Awaitable, Callable, Coroutine, Optional, Union

from azure.search.documents.aio import SearchClient
from azure.storage.blob.aio import ContainerClient
from openai import AsyncOpenAI, AsyncStream
from openai.types.chat import (
    ChatCompletion,
    ChatCompletionChunk,
    ChatCompletionContentPartImageParam,
    ChatCompletionContentPartParam,
)

from approaches.approach import ThoughtStep
from approaches.chatapproach import ChatApproach
from core.authentication import AuthenticationHelper
from core.imageshelper import fetch_image
from core.modelhelper import get_token_limit
```

Figure 11. Back-end approach of demo application- using Python code: importing libraries for Azure AI search client and integrating with OpenAi model for RAG method.

The ChatCompletion class is likely returned by methods or functions provided by the AsyncOpenAI class, which is used for asynchronous interactions with OpenAI's API. This completion object is crucial for processing the generated response within the application, whether it's for displaying to a user, further processing, or any other required actions based on the conversation context.

Application technologies:

The POC, which clones the azure-search-openai-demo application utilizes a variety of modern technologies across its frontend, backend, database, and deployment infrastructure. Here's a detailed description of each component:

Frontend: React

React is a popular JavaScript library for building user interfaces, particularly single-page applications where real-time updates and dynamic interaction are key.

Backend: Python (Quart)

Python is a versatile, high-level programming language known for its readability and broad library support. Quart is an asynchronous web framework for Python, built on the popular Flask framework, but designed to support asynchronous operations.

Vector DB: Azure AI Search

Azure AI Search, the cloud search service that provides powerful and sophisticated search capabilities for application development.

Deployment: Azure Developer CLI (azd)

Azure Developer CLI (azd) is a command-line interface tool provided by Microsoft Azure, designed to streamline the development, deployment, and management of applications on Azure.

Tech	azure-search-openai-demo
Frontend	React
Backend	Python (Quart)
Vector DB	Azure AI Search
Deployment	Azure Developer CLI (azd)

Figure 10: Technologies used in azure-search-openai-demo which is adopted as the development ground for POC of the Prototype. Screenshot collected from https://github.com/Azure-Samples/azure-search-openai-demo/blob/main/docs/other_samples.md (*Azure-Search-Openai-Demo/Docs/Other_samples.Md at Main · Azure-Samples/Azure-Search-Openai-Demo · GitHub*, n.d.)

Appendix 1.2: Feedback Form

Feedback%20form.docx

Feedback Form for evaluating Large Language Models and RAG-based Legal virtual assistant

The purpose of the evaluation is to systematically assess the effectiveness of AI language models and a RAG-based legal virtual assistant in providing accurate, comprehensive, and actionable legal advice for housing disputes.

Evaluating LLM Model's Responses

General Impressions: Please provide a general overview of your impression of the responses fr each Model. Include any standout features or immediate concerns.
Neither ChatGPT, Gemini, Perplexity or the prototype provided any meaningful advice or guidan the questions provided. Their answers lack detail and certainty. While the answers may look impressive on first glance, once you really read them you realise that they are empty and have n impact on the situation. The advice and guidance are actually very basic and common sense, h it does not give the user any indication of how to use it to solve the problem for which the user h engaged with them for a solution. it is impossible to verify the accuracy thereof. As a human being one would follow the guidance provided step-by-step, so it does make a difference, a substantial difference in reponsess, whe ChatGPT and Gemini makes slight adjustments in both the formation (numbering) and number answers provided.

Evaluation Criteria

1. **Correctness**

Sub-Criteria	Comments
Factual Accuracy (Please evaluate the truthfulness and accuracy of the information provided.)	It is impossible to verify accuracy of each of the responses gene all the bots as almost all of them contain misleading informatio for similar queries all of them provides different responses. All the models are inaccurate and giving mileading information consistently, in all the responses.
Provide any examples for Factual Accuracy	Prototype provides this response to user query: "The landlord d start investigating the matter until August 2007" This is what hap in 2007, not the landlord of the user. But the bot is not able to de the difference.
Source Reliability (Please assess the credibility of the	**ChatGpt 3.5:** No sources provided at all by ChatGpT 3.5.

Appendix 1.3: Experimental Setup

Experimental%20Set
up.xlsx

Dispute Category	Dispute Scenario	Prompt Number	Prompt	ChatGPT Response	Accuracy ChatGPT and Remarks	Gemini Response	Accuracy Gemini and Remarks	Perplexity Response	Accuracy Perplexity and Remarks	Prototype POC Response	Accuracy Prototype and Remarks	Comment	Providing Reference
	Scenario 1: Landlord Neglects Roof Repair: A large section of the roof above the living room has developed a significant leak. Large, dark water stains are spreading across the living room ceiling, particularly concentrated near the center. Peeling paint and warped drywall are evident around the edges of the stains. The tenant has notified the landlord about the leak in writing email with a detailed description and attached pictures of the damage. The leak has been present for over two weeks, and the landlord has not responded to the notification or taken any action to address the problem.		leaky roof, no matter how many emails I've sent or pics I've taken. It's been more than 3 weeks and no response. It's getting worse, and I'm freaking out. Can ya help me figure out what I gotta do? My house in Porvoo, Finland.	when it affects your living conditions. In Finland, tenants have rights protected by law, and landlords are obligated to maintain the property in good condition. Here's what you can do: 1. **Document Everything**: Keep records of all communication with your landlord regarding the leaky roof, including emails, texts, or any other form of communication. Also, document the dates when you first reported the issue and any subsequent attempts to contact them. 2. **Contact the Landlord Again**: Send another formal written notice to your landlord, preferably via registered mail, detailing the problem with the leaky roof, the potential damage it's causing, and your expectation for repairs within a reasonable timeframe. Make sure to keep a copy of this notice for your records. 3. **Contact Local Authorities**: If your landlord still fails to respond or take action, you can reach out to the local housing authority or municipal authorities in Porvoo. They can provide guidance on your rights as a tenant and may intervene on your behalf. 4. **Seek Legal Advice**: Consider consulting with a lawyer or legal advisor who specializes in landlord-tenant disputes. They can review your situation and provide guidance on how to proceed legally, including the possibility of withholding rent or taking legal action against your landlord for breach of contract. 5. **Temporary Fixes**: In the meantime, if the leak is causing significant damage or posing a safety hazard, you may need to take	accurate, Not complete. Providing Steps. But with no value. No reference provided.	and an unresponsive landlord. Here are some steps you can take to get the issue resolved: 1. **Review your lease agreement:** The first step is to understand your rights and responsibilities as a tenant, and your landlord's responsibilities regarding repairs. Your lease agreement should outline maintenance procedures and timelines for repairs in case of leaks. 2. **Document everything:** Keep copies of all email communication with your landlord regarding the leak. If you've sent pictures, ensure you have dates and timestamps for them. If possible, document any conversations you've had with your landlord about the leak. 3. **Contact relevant authorities:** In Finland, the Consumer Protection Ombudsman (Kuluttajavirasto) can advise tenants on their rights and how to proceed in disputes with landlords. You can contact them through their website https://www.kkv.fi/ or by calling +029 566 611 (available weekdays from 9 am to 4 pm). Here are some additional resources that you might find helpful: * The Finnish Environmental Protection Agency (Ympäristöministeriö) also has information on tenant rights and responsibilities on their website https://ym.fi/. You can try searching their website for keywords like "vuokralainen"		efforts, you have rights as a tenant in Finland to ensure your living conditions are safe and habitable. Here's what you can do: 1. **Contact the Housing Ombudsman**: In Finland, the Housing Ombudsman (Asuntoneuvos) can provide guidance and assistance in resolving housing-related disputes, including issues with landlords not maintaining the property[3]. 2. **Request a Housing Inspection**: You can contact the local municipal housing authority in Porvoo to request an inspection of the property due to the leaky roof. They can assess the situation and enforce necessary repairs if the landlord fails to act[3]. 3. **Review Your Rental Agreement**: Check your rental agreement for clauses related to maintenance responsibilities. Landlords in Finland are typically obligated to maintain the property in good condition, including addressing issues like a leaky roof[3]. 4. **Seek Legal Advice**: If informal attempts to resolve the issue fail, consider seeking legal advice from a tenant association or a lawyer specializing in housing law. They can provide guidance on your rights and legal options[3]. 5. **Document Everything**: Keep records of all communication with your landlord, including emails and photos of the issue. This documentation can be crucial if legal action is needed to compel the landlord to make repairs[3].		facts regarding your situation: 1. The tenant informed the landlord about the water leak in the roof on May 16, 2006, and several times after that. The landlord was aware of the leaks since 2004 1. 2. The damp-damaged room could mainly be used as a storage room due to the earthen cellar smell and allergic symptoms caused by being there 1. 3. The landlord did not start investigating the matter until August 2007 1. 4. The tenant had the apartment inspected in February 2008, which revealed that the leaks were causing harm according to the Health Protection Act 2. 5. After the complaint to the Consumer Disputes Board was initiated, the landlord refunded the rent and repaired the defect, but the refund did not cover the harm caused 2. 6. The tenant demanded an explanation after learning about the water leaks in the apartment. The landlord offered another apartment instead 3. Based on these facts, it appears that the landlord has been aware of the roof leak issue for a significant period of time and has not taken appropriate action to address it. You may consider filing a		etation 3	
	Scenario 2: Accidental Tenant Damage: While rearranging furniture, The tenant accidentally breaks the window in the living room, leaving a gaping hole in the window, shattering the window pane, damaged the window frame and adjacent wall. Immediately, the tenant contacted landlord and informed about the damage. However, tenant soon becomes anxious about the fact that the repairs are extensive and likely to incur substantial costs. Now, the tenant seeks legal attention to ensure s/he is not unfairly burdened with fines or penalties beyond what is reasonable for accidental damage.	6	So I broke a window in my apartment. Totally by accident, but it's bad, like the whole pane is shattered and the frame looks messed up too. I called my landlord and everything, but now I'm worried they're gonna try to make me pay a lot to fix it. Is that even allowed if it wasn't on purpose? My house located in Turku, Finland. What do you advise?	Accidents happen, and it's good that you've already informed your landlord about the situation. In Finland, the responsibility for repairs and maintenance often depends on the terms outlined in your lease agreement and the specific circumstances of the damage. Generally, landlords are responsible for repairing structural damage, such as broken windows, unless it can be proven that the damage was caused intentionally or due to negligence on the part of the tenant. Here's what you can do: 1. **Review Your Lease Agreement:** Check your lease agreement to see if it mentions anything about tenant responsibilities for damages. It should outline the landlord's obligations for repairs and maintenance. 2. **Document the Incident:** Keep records of your communication with the landlord, including the date and time you notified them about the broken window. Take photos of the damage as evidence, and if possible, write down details about how the accident occurred. 3. **Understand Tenant Rights:** In Finland, tenants have rights protected by law. Landlords cannot unreasonably withhold repairs or charge tenants for damages that are considered part of normal wear and tear or were caused by factors beyond the tenant's control. 4. **Negotiate Responsibly:** If your landlord suggests that you should contribute to the cost of repairs, discuss the matter calmly and refer to your lease agreement and Finnish rental laws. If you feel the landlord's demands are unreasonable or in violation of your rights, you may seek advice from a tenant association or legal counsel.	Overall not accurate, Not complete. Providing Steps. But with no value. No reference provided.	In Finland, tenants are generally liable for damages caused by negligence to their rental property https://www.globalpropertyguide.com/europe/finland/landlord-and-tenant. However, accidents do happen, and the law considers the extent of negligence involved. In your case, accidentally breaking a window is unlikely to be considered negligence on your part. Here's what you can do: * **Review your tenancy agreement:** Your tenancy agreement might have specific clauses about damage repair responsibilities. Reread the agreement to see if there's anything mentioned regarding window repairs. * **Contact your landlord again:** Let your landlord know that you are not refusing to take responsibility for the repair, but that you believe the extent of the repairs should not fall entirely on you since it was an accident. Be polite and open to discussing a solution. * **Contact your renters insurance:** If you have renters insurance, it might cover the cost of repairs for accidental damage. Contact your insurance company and inquire about filing a claim. It's important to remember that Finnish law uses the concept of "reasonable wear and tear" [https://www.globalpropertyguide.com/europe/finland/landlord-and-tenant](https://www.globalpropertyguide.com/europe/finland/land		As a tenant in Finland, you are responsible for any damage caused to the rental apartment, even if it was accidental[1][4]. However, you are not liable for normal wear and tear that occurs from regular use of the apartment[2]. In your case, a broken window and damaged frame would likely be considered damage beyond normal wear and tear. Your landlord is allowed to charge you for the repairs, but they must provide invoices showing the actual costs[1]. The landlord cannot simply keep your security deposit or charge an arbitrary amount. To avoid disputes, I recommend: 1. **Thoroughly document the damage** by taking photos and videos. This will help prove the extent of the damage was not excessive. 2. **Obtain repair estimates from licensed contractors** and share them with your landlord. This shows you are being reasonable. 3. **Offer to pay for the repairs in installments** if the total cost is high. Most landlords are willing to work with tenants in good faith. 4. **As a last resort, you can challenge the landlord's charges** in		According to the Finnish Act on Residential Leases, if the apartment's deficient condition is not due to your negligence or carelessness, you have the right to be exempted from paying rent or have the rent reasonably reduced during the period when the apartment cannot be used or is not in the required condition 1. However, the right to be exempted from paying rent or have the rent reduced does not exist before the landlord receives notice of the apartment's deficient condition 1. You have a duty to immediately notify your landlord of any damage to the apartment or any deficiency that the landlord is responsible for repairing 2. If the repair or alteration work is carried out due to damage for which you are liable, the landlord may require you to pay for the repairs 2. The landlord must be able to prove their claim about the exceptional wear and tear of the apartment 3. It is recommended that you communicate with your landlord and discuss the situation. Citations: 1. Finnish Act on Residential Leases.pdf#page=8 2. Act on renting a residential apartment 481, 1995 - Current			
	Scenario 2: Accidental Tenant Damage: While rearranging furniture, The tenant accidentally breaks the window in the living room, leaving a gaping hole in the window, shattering the window pane, damaged the window frame and adjacent wall. Immediately, the tenant contacted landlord and informed about the damage. However, tenant soon becomes anxious about the fact that the repairs are extensive and likely to incur substantial costs. Now, the tenant seeks legal attention to ensure s/he is not unfairly burdened with fines or penalties beyond what is reasonable for accidental damage.	7	By accident I shattered the living room window today. There's a giant hole now and it looks terrible. I messaged my landlord straight away to let them know what happened, but now I'm worried about the repairs. They seem really expensive and I don't want to get stuck paying a fortune for something that was an accident. Is there anything I can do to make sure that I don't need to pay? I live in Helsinki.	Sorry to hear about the accident with your window! It's good that you've already informed your landlord about what happened. Here are some steps you can take to ensure you don't end up paying for the repairs: 1. **Review Your Lease Agreement**: Check your lease agreement to see if there are any clauses related to damages caused by tenants. Some leases specify that tenants are responsible for certain types of damages, while others may require the landlord to cover repairs. 2. **Communicate with Your Landlord**: Keep the lines of communication open with your landlord. Explain that it was an accident and express your willingness to cooperate in resolving the issue. Sometimes, landlords may be understanding and willing to cover the cost of repairs, especially for accidental damages. 3. **Check Your Renter's Insurance**: If you have renter's insurance, review your policy to see if it covers accidental damages to the property. Depending on your coverage, your insurance may help cover the cost of repairs. 4. **Get Multiple Repair Quotes**: If your landlord expects you to arrange for the repairs, consider getting multiple quotes from different repair companies. This can help ensure that you're not overcharged for the repairs. 5. **Negotiate with Your Landlord**: If the repair costs are high and	Overall not accurate, Not complete. Providing Steps. But with no value. No reference provided.	In Helsinki, tenant law protects you from being responsible for normal wear and tear, but accidental damage might be a different story. Here's what you can do: * **Review your lease agreement:** Look for any clauses related to repairs and tenant responsibility for damages. * **Check Finnish tenant law:** The Finnish Consumer Protection Act ([LÄHDE Finnish Consumer Protection Act]) outlines tenant rights and responsibilities. You can find resources online or consult a tenant rights organization (https://www.vuokralaiset.fi/en/) for free legal advice. * **Reach an agreement with your landlord:** Since it was an accident, negotiate with your landlord. Explain that you informed them immediately and were willing to work with them on a solution. Here are some possibilities to consider during the discussion: * **Replacement vs. Repair:** Replacing a whole window can be expensive. See if repairing the broken pane is an option. * **Sharing the Cost:** You might be able to share the cost of repairs with the landlord, especially if the window was older or nearing the end of its lifespan.							