



الجمهورية الجزائرية الديمقراطية الشعبية
The People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research
جامعة محمد بوضياف بالمسيلة
University Mohamed Boudiaf of M'sila



كلية الرياضيات والإعلام الآلي
Faculty of Mathematics and Informatics

قسم الإعلام الآلي
Department of Computer Science

Domain: Mathematics and Computer Science

Thesis Presented to Fulfill the Partial Requirement
for the **Master's Degree** in Computer Science

Specialty: *ISSE / SIGL*

Prepared By: Aymen Gasmi

Supervised By:

Said Gadri

Entitled

**Designing and developing an Intelligent Chatbot for Product Inquiry
Analysis in E-Stores Using NLP and LLMS Models**

Jury Members

Name and Surname	Function
Samir Akrouf	President
Said Gadri	Supervisor
Noureddine Amraoui	Examiner

Academic Year **2024/2025**

الاهداء

بسم الله الرحمن الرحيم، الحمد لله الذي بنعمته تتم الصالحات، والصلاة والسلام على سيدنا محمد، خاتم الأنبياء والمرسلين، وعلى آله وصحبه أجمعين.

أهدي هذا العمل المتواضع، ثمرة سنواتٍ من الكفاح والاجتهاد، إلى والدي العزيز، قدوتي ومعلمي الأول، الذي لم يدخر جهداً في دعمي وتشجيعي، فكان سنداً لا يميل، أطال الله في عمره وامتعه بالصحة والعافية. وإلى والدتي الحبيبة، نبع الحنان، ومصدر السكينة، التي غرست في نفسي الثبات، وكانت دوماً إلى جانبي بدعائها ودعمها، فجزاها الله عني خير الجزاء.

كما أهدي هذا التخرج إلى عائلتي الكريمة، إخوتي وأخواتي الأعزاء، الذين كانوا لي دوماً خير معين، وشركاء في كل نجاح، فلهم مني كل الحب والتقدير والامتنان.

التشكرات

اولا أحمد الله تعالى حمداً يليق بجلال وجهه وعظيم سلطانه، إذ وفقني ويسّر لي إتمام هذا العمل، وأعانني على تجاوز الصعوبات، فله الحمد في الأولى والآخرة، وله الشكر على ما أنعم وتفضل.

وفاءً وتقديراً، واعترافاً بالجميل، نتوجه بأسمى عبارات الشكر والعرفان إلى صاحب الفضل بعد الله، أستاذنا الفاضل الدكتور سعيد قادري، الذي كان لنا خير مُعين وناصح طيلة العام الدراسي، فجزاه الله عنا كل خير، وبارك في علمه وجهده.

كما نعبر عن بالغ شكرنا وامتناننا للأساتذة الأفاضل، الذين لم يدّخروا جهداً في توجيهنا ومساندتنا، فكانوا لنا مصدر إلهام، وركيزة من ركائز العلم والمعرفة.

ولا يفوتنا أن نتقدم بخالص الشكر والتقدير إلى السادة أعضاء لجنة المناقشة، على ما بذلوه من جهدٍ، وتوجيهاتٍ سديدة، فلکم منا كل الامتنان والتقدير.

Contents

- Dedication** **i**

- Acknowledgment** **ii**

- Table of Contents** **iii**

- List of Figures** **vi**

- General Introduction** **1**

- 1 AI and Chatbots to Improve Customer Support** **3**
 - 1 Introduction 4
 - 2 Online Store 4
 - 2.1 Definition 4
 - 2.2 Spread and Importance of E-Commerce Stores 4
 - 2.3 Type of E-stores 4
 - 2.4 Exemple E-stores in Algeria 5
 - 3 Customer support 7
 - 3.1 Definition 7
 - 3.2 Role of Customer Support 7
 - 3.3 Importance of customer support 7
 - 3.4 Customer Support Channels 9
 - 4 Enhancing customer support with AI Assistants(Chatbots) 9
 - 4.1 Example of International online store AI assistants 10
 - 5 Conclusion 12

- 2 Coupling NLP with ML and DL to build AI Agent** **13**
 - 1 Introduction 14
 - 2 NLP Overview 14
 - 2.1 Natural Language 14
 - 2.2 Natual Language Processing Definition 14
 - 2.3 Component Of Natural Language Processing 14
 - 3 NLP and AI 15

4	Natural Language Processing Technique	16
5	Natural Language Processing Tasks and Application	17
6	Machine Learning for NLP	18
6.1	Naive Bayes	18
6.2	Support Vector Machines (SVM)	18
6.3	Logistic Regression	18
7	Deep Learning for NLP	19
7.1	Recurrent Neural Networks (RNNs)	19
7.2	Convolutional Neural Networks (CNNs)	19
7.3	Transformers	20
8	Chatbots Definition	20
9	Chatbots Types	21
10	Chatbots benefits in buisness	23
10.1	For custommer	23
10.2	For businesses	24
11	Platform to create your chatbot	24
12	No code disadvantages	26
13	Conclusion	26
3	Optimizing Conversational AI with RAG and LLMs	28
1	Introduction	29
2	Technique used in our project	29
2.1	Large Language Model	29
2.2	Retrieval-Augmented Generation	30
3	Hardware used	31
4	Tool used	31
4.1	Programming language	31
4.2	Framework and Libraries	31
4.3	Software and API	32
5	Project Implementation	33
5.1	Knowledge Datasets Preparation	33
5.2	Code Implementation and Methodology	35
5.3	Chatbot Interface Explanation	39
5.4	Evaluation and Testing	42
6	Future Work and Enhancements	43
6.1	Interaction Logging and Analysis	43
6.2	Intent Detection Model	43
6.3	Real-Time Database Integration	44
6.4	Fine-Tuning of Language Models	44

6.5	Voice Interaction Capabilities	44
6.6	Alerting and Notification System	44
6.7	Dynamic Knowledge Base Updates	44
6.8	Administrative Knowledge Management Interface	44
7	Conclusion	45
	General Conclusion	46

List of Figures

Fig 1.1	Jumia platform	6
Fig 1.2	OuedKniss	6
Fig 1.3	Ali Express	11
Fig 1.4	Hostinger	11
Fig 2.1	How NLP, ML, and DL are related	16
Fig 2.2	RNN	19
Fig 2.3	CNN	19
Fig 2.4	Transformers	20
Fig 2.5	General chatbot architecture sequence diagram	21
Fig 2.6	Rule-based-chatbot	22
Fig 2.7	Ai-chatbot	22
Fig 2.8	Chatbot-Categories	23
Fig 2.9	Jotform	25
Fig 2.10	WotNot	25
Fig 2.11	CustomGPT	25
Fig 2.12	Microsoft Copilot Studio	26
Fig 3.1	RAG Technique	30
Fig 3.2	Dataset Used	35
Fig 3.3	Loading-data	36
Fig 3.4	Embeddings and create vector database	37
Fig 3.5	Api code	38
Fig 3.6	API Request and Response Result from the Chatbot Server	39
Fig 3.7	Chatbot Interface in the Online Store	39
Fig 3.8	English Chatbot Conversation Interface	40
Fig 3.9	Arabic Chatbot Conversation Interface	41
Fig 3.10	Multi language	42
Fig 3.11	Similarity Score Distribution	43

General Introduction

The world has witnessed a rapid technological transformation, driven by artificial intelligence (AI), which has contributed to transforming and developing various economic and social fields. AI has evolved from a conceptual research field into a tangible reality gradually embodied in daily life. It has also contributed to developing intelligent systems specialized in various fields, such as decision-making, data analysis, and interacting with users in ways that were not possible before. This phenomenon marks a new era in the interaction between humans and technology.

One of the most prominent of these developments is the increasing reliance on chatbots, or "intelligent agents," or advanced systems that perform tasks requiring contextual analysis and understanding.[1] Chatbots are among the most widely used of these systems, due to their ability to communicate with humans. This technological development has found widespread application in the field of digital customer support, where it has emerged as a primary mechanism for providing technical assistance, answering inquiries, guiding users, and improving the interaction experience with electronic systems.

With the increasing use of digital services, customer service is one of the most important services that ensure user satisfaction and continued engagement with digital platforms. In this field, chatbots have proven effective in meeting users' needs around the clock without the need for constant human supervision. However, challenges remain regarding chatbots' ability to understand user inquiries and provide appropriate responses within accurate and understandable contexts. Many modern chatbots rely on artificial intelligence technologies, such as natural language processing (NLP), machine learning, generative artificial intelligence, and large language models. This has led to the emergence of advanced frameworks such as Retrieval Augmentation Generation (RAG), which combines the advantages of information retrieval and text generation to enhance their ability to understand user intent and interact with them in a human-like manner.

Manual customer service requires significant time, effort, and labor, and is relatively expensive. Organizations face challenges in providing effective customer service 24/7. The persistent lack of human support leads to a gap in user experience, especially on digital platforms that rely on instant responses.

As AI moves from research to practical applications, this thesis will explain the techniques of Natural Language Processing (NLP), Large Language Models (LLM), and Retrieval Augmentation Generation (RAG), their relationship to AI, and how to develop and evaluate highly efficient chatbots that contribute to improving customer service, maintaining customer engage-

ment on digital platforms, and enhancing customer support. We will also discuss how these bots can be developed to understand user inquiries and provide appropriate responses within precise and understandable contexts.

Our manuscript is organized as follows:

- **In the general introduction**, we introduce the research area, highlighting the importance of enhancing e-commerce platforms by integrating intelligent chatbots. We discuss the increasing reliance on digital services and the need for effective and automated customer support solutions to improve user experience and business performance.
- **Chapter 1**: This chapter covers e-commerce types, their digital service delivery, customer support methods, and how chatbots improve operational efficiency through instant responses.
- **Chapter 2**: This chapter introduces chatbots, their roles in business, and key technologies used to build them, including rule-based systems, machine learning, and NLP. It also highlights popular chatbot examples.
- **Chapter 3**: This chapter covers the design and implementation of the proposed chatbot, detailing the development steps, tools used, and final results, along with suggestions for future improvements.
- Finally, the general conclusion summarizes the main findings of the study, discusses the effectiveness of integrating chatbots into e-commerce, and identifies future research directions for improving automated customer support systems.

This study contributes to the ongoing development of intelligent support systems in digital commerce, demonstrating the practical application of AI in real-world customer interaction scenarios.

Chapter 1

AI and Chatbots to Improve Customer Support

1 Introduction

In this chapter, we discuss the classifications and types of e-commerce stores, with a particular focus on the ways these services are delivered digitally. Additionally, we analyze the various forms of customer support used in e-commerce stores, including manual intervention and automated assistance. We then explore the role of the chat-bot in improving the operational efficiency of e-commerce stores by facilitating immediate responses to customer inquiries, easing the burden on support staff, and providing a smoother and more professional user experience.

2 Online Store

2.1 Definition

The term online store is, the most simply put, the store of services or products on the global network. According to the official definition, an online store is an integral part of a process known as online shopping. Online shopping involves a process in which visitors can buy certain products or services offered within an online store.[2].

2.2 Spread and Importance of E-Commerce Stores

E-commerce has spread in recent years and witnessed remarkable development. Online shopping has become like traditional shopping, allowing customers to purchase products and services from anywhere, at any time, without having to visit traditional stores.

The importance of e-commerce lies in its ability to save consumers time and effort, while providing a variety of options at competitive prices. E-commerce stores also allow business owners to reach a wider audience, both locally and globally, leading to increased sales and accelerated business growth.

E-commerce stores also provide a flexible marketing environment, enabling companies to analyze customer behavior and improve user experience through analytics and artificial intelligence. Additionally, modern e-commerce stores increasingly integrate automated customer support tools, such as chatbots, which help provide immediate and accurate responses 24/7.

With all these advantages, e-commerce stores have become an essential part of the digital economy and a vital component of any business strategy aimed at growth and innovation in today's competitive market. [3]

2.3 Type of E-stores

Below are the main types of eCommerce models, each with its own use case and application [4]:

-
- **Business-to-Consumer (B2C)**
A business sells products or services directly to individual customers.
Example: Buying clothes from an online store like Amazon.
 - **Business-to-Business (B2B)**
A business sells products or services to another business.
Example: A company selling software to other companies.
 - **Business-to-Government (B2G)**
A business provides goods or services to a government agency.
Example: A tech company building a system for a government department.
 - **Consumer-to-Consumer (C2C)**
Consumers sell products or services to other consumers.
Example: Selling used items on platforms like eBay.
 - **Consumer-to-Business (C2B)**
An individual offers products or services to a business.
Example: A freelancer designing a logo for a company.
 - **Consumer-to-Government (C2G)**
An individual provides services or makes payments to the government online.
Example: Paying taxes or fines through a government website.
 - **Business-to-Business-to-Consumer (B2B2C)**
A business sells to another business, which then sells to the final consumer.
Example: A wholesaler sells to a retailer, and the retailer sells to the customer.
 - **Business-to-Administration (B2A)**
A business delivers products or services to public administration or government bodies.
Example: A company providing cybersecurity services to a government office.

2.4 Exemple E-stores in Algeria

Jumia is primarily a Business-to-Consumer (B2C) online marketplace. It connects businesses and sellers (both local and international) directly with individual consumers, offering a wide range of products for sale



Figure 1.1: Jumia platform

Ouedkniss is an Algerian B2C and C2C online platform where individuals and businesses can buy and sell a wide range of products and services.



Figure 1.2: OuedKniss

3 Customer support

3.1 Definition

Customer Support pertains to a systematic collection of methodologies and protocols aimed at facilitating assistance for clientele who necessitate specialized support. Such individuals may possess inquiries regarding your merchandise or enterprise, or they may encounter complications related to a product or service.

Customer support can come from all sorts of places—like in person, over the phone, through email, or even via live chat. The main idea is to be there for your customers when they need you, no matter how they reach out. [5]

3.2 Role of Customer Support

The primary goal of customer support is to help customers when they encounter problems with your company's products or services. However, it's also important that your support team be friendly and make customers feel valued. If customers receive poor service, they're likely to negatively impact your entire brand.[6]

Today, customer support is a fundamental pillar that determines the quality of service provided to consumers or clients by competing companies and organizations. Gone are the days when it was merely about technical fixes or simple queries. It has transformed into an integrated experience representing the human, interactive interface between the customer and the company. Every time a user interacts with the support team, a mental image is created of whether the company is professional about their work and is genuinely concerned for the customer, thus becoming an essential element of corporate growth and success strategies.

With the promise of rapid technological advancement and the many options now available to the user, the onus is on companies to provide highly efficient support that meets growing consumer expectations, not only with regard to response time but with the quality of the solutions provided, accompanied by an absolutely professional level of understanding of the user's needs. The customers are no longer judging the technical quality of a product or service on the grounds of its specification or price. Instead, support can now prove one of the highest ranking factors that ultimately influence buying decisions, customer loyalty, and repeated uses of the product or the service.

3.3 Importance of customer support

Customer support is important as it hinges on the crucial role played by support agents in promptly and efficiently addressing customer concerns, thereby influencing customer satisfaction. This, in turn, has a ripple effect on customer retention, customer lifetime value, and the overall brand reputation.[7]

- **Positive brand reputation**

Establishing a positive brand image in your industry is challenging, as it can significantly impact customer perception. Excellent customer support enhances your reputation. It demonstrates your commitment to customer satisfaction.

Your brand's image is a precious asset that can significantly impact customer perception and trust. Great customer support enhances this image by conveying a commitment to quality and customer satisfaction. When customers receive attentive and effective support, they associate these positive experiences with your brand. This, in turn, fosters a sense of trust and reliability.

A positive brand image not only attracts new customers but also encourages repeat business from existing ones. It builds a reputation that can withstand challenges and setbacks, helping your business weather storms and maintain a strong market presence. In a world where reputation and trust are paramount, prioritizing customer support is an investment in shaping a positive and enduring brand image.

- **Improved customer retention**

Maintaining existing customers is more cost-effective than acquiring new ones, and customer retention is key to a sustainable business model.

A customer support system directly impacts your retention rate by fostering long-term relationships. When your customer support team provides quick and effective solutions, customers feel heard and valued. This leaves a positive impression on them.

As a result, you can expect repeat business, larger ticket sizes, and increased overall sales

When customers receive timely responses and solutions, they remain loyal to your business and actively promote your brand through word-of-mouth

- **Better competitive advantage**

In today's competitive marketplace, customers have numerous options to replace your product or services. A single negative experience with customer support can cause customers to permanently switch to a competitor.

When customers are choosing among different options, their decision is influenced not only by the quality of the product or services but also by the level of customer support provided. Having exceptional customer support conveys a powerful message that your company is committed to going the extra mile to meet customer needs. This can turn the tables and influence potential customers to choose your brand over others.

- **Increased sales and revenue**

Outstanding customer support enhances both customer satisfaction and retention. Loyal customers are more likely to repeat transactions, increasing their lifetime value. Providing

them with an exceptional experience and gaining their trust makes them inclined to return for your products or services. Thus, investing in customer support can be a strategic and economical approach to boost sales and revenue.

3.4 Customer Support Channels

Organizations use a range of customer support channels to accommodate diverse preferences and needs. These include phone support for intricate issues, email for comprehensive queries, live chat for instant help, and self-service resources for those who prefer to find solutions independently.[8]

- **Phone support**

This is the most traditional and preferred way for customers to speak directly with a support agent. Phone support is best for addressing complex issues that need detailed assistance or a personal touch.

- **Email support**

Email is a professional and versatile channel for customers to inquire or raise concerns. This platform allows seamless communication while enabling both customers and support teams to keep a comprehensive record of their interactions for future reference.

- **Live chat**

Live chat provides customers with real-time support and assistance. It is a convenient channel for customers to get in touch with a support agent and experience immediate assistance. Live chat typically offers faster first-response times, improving the overall customer experience.

- **Social media support**

Customers can use social media platforms like Twitter and Facebook to voice concerns or ask questions. Businesses have social media support teams that constantly monitor these channels to provide timely responses.

- **Self-service options** FAQs, knowledge bases and AI-powered chatbots help customers resolve issues independently.

4 Enhancing customer support with AI Assistants(Chatbots)

With the continued expansion of artificial intelligence technologies, the benefits of adopting chatbots in the customer service environment are increasing. These systems enable organizations to provide immediate support around the clock, regardless of business hours or time zones, reducing wait times and enhancing customer satisfaction and loyalty.

Chatbots contribute to reducing operational costs, as they are able to handle a large volume of recurring inquiries, provide initial support, and distribute requests, allowing human support staff to focus on complex cases.

Furthermore, these systems enhance the efficiency of support teams by automating repetitive tasks, gathering initial information, and automatically directing cases to the appropriate departments. For example, the cosmetics brand Lush used an AI agent to respond to general inquiries, freeing up employees to focus on complex issues. This resulted in an average saving of 5 minutes per support ticket, equivalent to approximately 360 hours per month.

Chatbots offer a high level of personalization, as they can connect to back-end systems and retrieve data such as purchase history and user preferences, enabling them to provide personalized recommendations and support services.

These systems also contribute to improving the customer experience thanks to their ability to interact in multiple languages, enabling them to support a global, multicultural customer base. Omnichannel support allows customers to interact with bots across their preferred channels, such as live chat, messaging apps, and social media, enhancing accessibility and the overall user experience.[9]

Recent advances in generative AI technologies indicate that chatbots are transforming from simple tools that provide links to support articles into autonomous, intelligent agents capable of understanding complex conversations and interacting in a manner that mimics the performance of human agents. These systems enable integration with back-end enterprise systems to provide personalized support based on each user's context. Depending on the level of automation required, these intelligent agents can handle more than 80% of customer inquiries independently, with the ability to seamlessly escalate complex cases to relevant staff when needed.[10]

4.1 Example of International online store AI assistants

These are the most famous international companies that use professional and advanced chatbots to provide excellent and good customer service, and I had experience with them.

AliExpress, a leading global e-commerce platform, uses an intelligent chatbot system to streamline customer service. This AI-powered assistant enables users to receive instant assistance with order tracking, product inquiries, and other transaction-related issues through live chat. The chatbot interface provides 24/7 multilingual support, efficiently handling common requests and escalating complex cases when needed, significantly improving the shopping experience with efficient and always-available assistance.products for sale



Figure 1.3: Ali Express

A platform for selling web hosting services and providing you with a domain name. This platform provides you with a smart robot called Kodee, which allows you to view the services you've purchased and guides you through purchasing or working on your website.

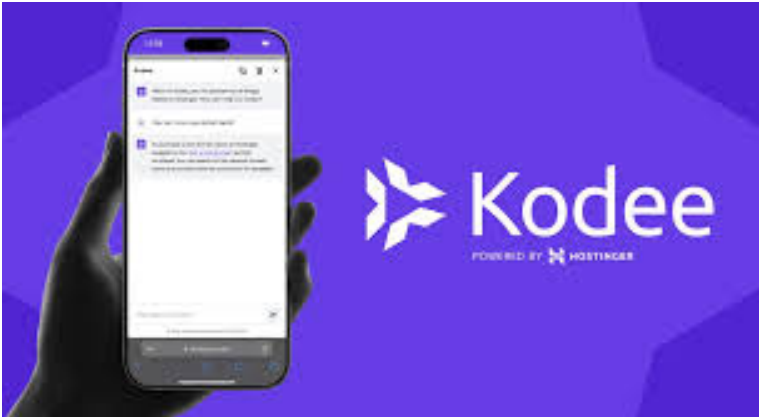


Figure 1.4: Hostinger

5 Conclusion

With the rapid growth of e-commerce platforms and online service marketplaces, improving how we interact with customers has become essential. Today, e-commerce is a core part of the digital economy, allowing users to buy products and services anytime and from anywhere, while saving time, offering a variety of options, and providing competitive prices. To enhance the user experience and maintain service quality, it's important to integrate chatbots into the system. These smart tools provide instant support 24/7, without being limited by working hours or time zones. Customer support is no longer just a basic function—it now plays a big role in influencing purchase decisions and building customer loyalty. Using chatbots helps reduce waiting times, ease the pressure on support teams, and improve overall operational efficiency. In the end, this leads to higher customer satisfaction, stronger trust, and better sales and long-term growth.

Chapter 2

Coupling NLP with ML and DL to build AI Agent

1 Introduction

This chapter explores the field of artificial intelligence (AI) and its main branch, natural language processing (NLP). NLP is defined as a branch of AI that enables computers and digital devices to recognize, understand, and generate human language. Modern NLP relies heavily on machine learning (ML) and deep learning (DL) techniques to achieve advanced capabilities. Deep learning, particularly using models such as neural networks and transformers, has revolutionized the field of AI and significantly improved NLP applications. We will cover the core components of NLP: natural language understanding (NLU) for interpreting meaning, and natural language generation (NLG) for generating text, along with key processing techniques. The chapter will then focus on the various tasks and applications of NLP.

2 NLP Overview

2.1 Natural Language

Natural language is the language people use every day, such as English, Arabic, etc. This language evolves naturally over time within societies and is often rich, complex, and sometimes even mysterious, influenced by culture and history. On the other hand, programming languages such as Python, C++, and Java were deliberately designed with strict rules and are not natural languages, but rather a language used for communication between humans and computers. This distinct difference between human natural languages and programming languages is fundamental to the field of natural language processing (NLP), which aims to help computers understand and process human language in a way that feels natural to us.[11]

2.2 Natural Language Processing Definition

Natural language processing allows computers and digital devices to recognize and understand text and speech and generate human-like textual output by combining computational linguistics (rule-based modeling of human language), statistical modeling, machine learning, and deep learning. IN addition, natural language processing is increasingly used in solving enterprise problems, simplifying and automating business processes, enhancing employee productivity, and simplifying work processes. [12]

2.3 Component Of Natural Language Processing

Natural language processing (NLP) consists of tow basic components: natural language understanding (NLU),and natural language generation (NLG).

- **NLU** : Natural Language Understanding (NLU) decodes the meaning of text/speech through

syntactic and semantic analysis, and builds ontologies to map relationships between words – enabling machines to interpret, analyze, and understand language like humans.

- **NLG** : Natural Language Generation (NLG) transforms structured data into human-like text or speech, enabling computers to “write.” It powers text summarization, dynamic reporting, and conversational AI using advanced techniques like neural networks and transformers.

NLP enables machines to understand (NLU) and generate (NLG) human language, powering intelligent communication.[13]

3 NLP and AI

Artificial intelligence (AI) is a field of computer science that focuses on developing intelligent systems that mimic human intelligence. Founded in the 1950s, early AI relied on logic and rule-based systems. Machine learning (ML), a branch of AI, allows systems to learn from data without explicit programming, while deep learning (DL), a specialized branch of machine learning, uses neural networks inspired by the human brain. Natural language processing (NLP), machine learning (ML), and deep learning (DL) are interrelated and distinct subfields of AI. Early NLP systems relied on rules, but modern NLP leverages machine learning and deep learning (DL) for more advanced applications. Machine learning operates through three main paradigms:

- Supervised learning (learning from labeled data, such as spam detection),
- Unsupervised learning (finding patterns in unlabeled data, such as topic modeling),
- Reinforcement learning (learning by trial and error, less common in natural language processing (NLP) but common in robotics and gaming).

Deep learning has revolutionized the field of artificial intelligence, improving applications such as speech recognition, machine translation, and natural language processing (NLP). While NLP increasingly relies on machine learning and deep learning techniques, challenges remain in areas such as reinforcement learning and semi-supervised methods. This book explores these methods in more depth to develop practical solutions for NLP.[14]

This diagram illustrates the relationship between artificial intelligence (AI) and natural language processing (NLP), explaining that NLP is a branch of AI that overlaps with machine learning (ML) and deep learning (DL). It also demonstrates how NLP draws on techniques from both machine learning and deep learning within the broader scope of artificial intelligence (AI).

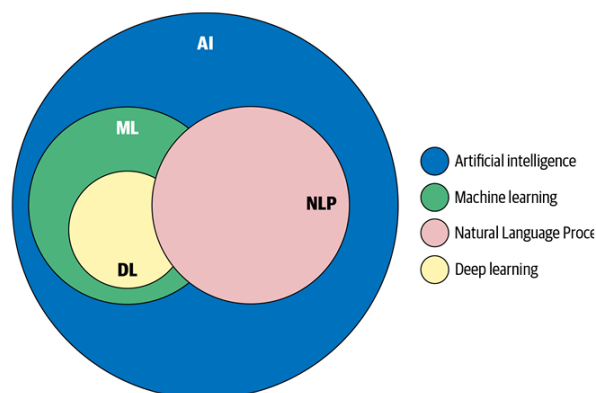


Figure 2.1: How NLP, ML, and DL are related

4 Natural Language Processing Technique

Natural language processing (NLP) encompasses many techniques, depending on the task being performed. this is a simple technique used in NLP :

- **Tokenization** It is the process of dividing a sentence into a number of parts (words) and is called a token, or separating each word separately, to deal with it and know its type and so on. There are two types of it: Word tokenizer, which separates the words, and sentence tokenizer, which separates the sentences, each sentence separately.

Example : In the previous chapter, we saw a quick overview of what is NLP, what are some of the common applications and challenges in NLP, and an introduction to different tasks in NLP.

['In', 'the', 'previous', 'chapter', ',', 'we', 'saw', 'a', 'quick', 'overview', 'of', 'what', 'is', 'NLP', ',', 'what', 'are', 'some', 'of', 'the', 'common', 'applications', 'and', 'challenges', 'in', 'NLP', ',', 'and', 'an', 'introduction', 'to', 'different', 'tasks', 'in', 'NLP', '.']

- **Speech of Part :** is the grammatical determination of the type of word, whether it is a verb, a noun, or an adjective, based on the context of the word and the sentence in which it is located, and not the word itself. It is based on the fact that the meaning of any word is not in itself, but in its content and context, and according to the words surrounding it. Therefore, it interprets each word according to its content, meaning, and context, and classifies it among many sections.
- **Stemming :** It relies on the idea of returning a word to its root and removing all additions, whether at the beginning or end. For example, words like "play" are all repeated (playing,

player, plays, played). This tool is extremely useful for determining word meanings, as well as for combining all words with the same root into one word. If words that all revolve around the word "win" are repeated in multiple forms, their total occurrences should be calculated after they are traced back to their original root.

- **Word embedding :** Word Embedding is a technique in Natural Language Processing (NLP) that represents words as dense, low-dimensional numeric vectors. These embeddings capture semantic (meaning-based) and syntactic (grammar-based) relationships, ensuring that words with similar meanings have similar vector representations. Unlike traditional methods like Bag of Words (BoW) or TF-IDF, which generate high-dimensional, sparse vectors, word embeddings provide compact and meaningful representations ideal for machine learning models.

5 Natural Language Processing Tasks and Application

The use of natural language processing tasks varies from one field to another. Each field has a specific application depending on the project. These are a set of basic tasks that are frequently used in various fields. Some examples include [11]:

- **Language modeling :** Natural language processing can be used to predict the next words or sequence of words. This technology is also essential in speech recognition, sentiment analysis, and correcting spelling errors.
- **Text classification:** Text classification is used to organize text data and documents into pre-defined categories and labels. Natural language processing (NLP) algorithms analyze the text content and classify it into relevant categories based on underlying topics or themes. The most important uses are: email filtering and information retrieval, which facilitates efficient information management and retrieval.[12]
- **Information retrieval :** Information retrieval (IR) is defined as the process of accessing and retrieving the most appropriate information from text based on a particular query given by the user. The goal of information retrieval is to provide a user with documents that fulfill the user's information need. An information retrieval system searches a collection of natural language documents with the goal of retrieving exactly the set of documents that matches a user's question, indicating the existence and location of documents that might contain the required information. The documents that satisfy the user's requirement are called relevant documents. IR is a powerful technology for finding information.
- **Conversational agent :** Natural language processing is used in developing chatbots through their ability to understand and interact with users in natural language, providing answers relevant to the question asked. Among its most important uses are customer support and virtual shopping assistance.[13]

-
- **Information Extraction and Question Answering Systems** : Information extraction uses NLP techniques to extract structured data from unstructured text, enabling tasks like entity recognition and question-answering.

6 Machine Learning for NLP

Machine learning (ML) techniques are used in natural language processing (NLP) to analyze and understand text data and natural language, just as they are used with images or speech. These techniques include supervised learning techniques, such as classification and regression, and NLP tasks—for example, classifying news articles by topic (sports and politics) or predicting stock prices based on articles or emails. Unsupervised techniques, such as clustering, are also used to group similar documents without predefined labels[14]. This is the most machine learning Algorithms used in the development of natural language processing system :

6.1 Naive Bayes

A simple probabilistic classifier based on Bayes' theorem, it assumes feature independence to calculate class probabilities. It is widely used in text classification tasks (such as spam filtering and sentiment analysis) due to its speed and accuracy on small data, often impressive accuracy. As Raschka (2014) notes, "Naive Bayes classifiers... are known for creating simple, high-performance models, especially in the fields of document classification." [15]

6.2 Support Vector Machines (SVM)

A margin-based linear classifier that finds the hyperplane separating classes with maximum margin. SVMs are effective in high dimensional text spaces and have been widely applied to NLP classification (e.g. document categorization, sentiment or topic classification). In early work, Joachims (1998) demonstrated that “SVMs achieve substantial improvements over the currently best performing methods and behave robustly” on text categorization. SVMs often outperform other methods on NLP benchmarks and do not require extensive feature tuning [16].

6.3 Logistic Regression

A discriminative linear classifier that models the probability of class membership (via the sigmoid or softmax function). In NLP it is a standard baseline for supervised classification tasks (e.g. sentiment, topic, or POS tagging), providing well-calibrated probability estimates. As Jurafsky and Martin (2025) explain, “in natural language processing, logistic regression is the baseline supervised machine learning algorithm for classification, It often rivals SVMs in performance and serves as the basic building block for more complex models (e.g. neural networks) [17].

7 Deep Learning for NLP

7.1 Recurrent Neural Networks (RNNs)

Neural networks designed for sequential data, RNNs process one token at a time while maintaining a hidden “memory” state. This makes them well suited for language modeling, machine translation, speech recognition, and any task where order matters. RNNs (especially LSTM or GRU variants) “can use their internal memory to process and work on arbitrary sequences of inputs,” making them powerful for NLP sequence tasks. They have been “successfully used for sequence labeling and sequence prediction tasks, such as ... language modeling, machine translation” in NLP [18].

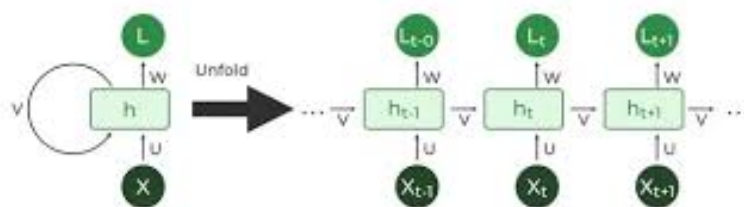


Figure 2.2: RNN

7.2 Convolutional Neural Networks (CNNs)

CNNs apply convolutional filters over word embeddings or character sequences to capture local n-gram features. Although originally developed for images, CNNs have proven very effective in NLP. For example, Yoon Kim (2014) showed that CNNs “have subsequently been shown to be effective for NLP and have achieved excellent results in semantic parsing, search query retrieval, sentence modeling, and other traditional NLP tasks” [19].

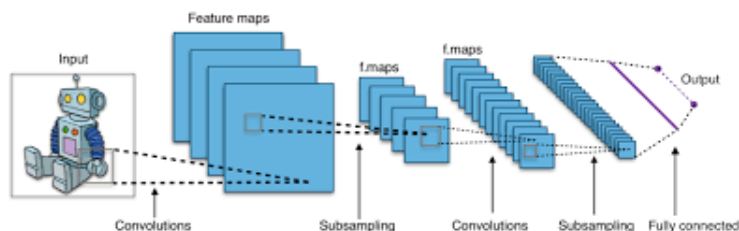


Figure 2.3: CNN

7.3 Transformers

A purely attention-based architecture that replaces recurrence with self-attention. Transformers model global dependencies by allowing every token to attend to every other token in the sequence. Vaswani et al. (2017) introduced the Transformer, noting it “eschews recurrence and instead relies entirely on an attention mechanism to draw global dependencies between input and output”. Transformers enable highly parallel computation and have become the foundation of modern NLP (e.g. BERT, GPT). They achieved state-of-the-art results in machine translation and many other NLP tasks [20].

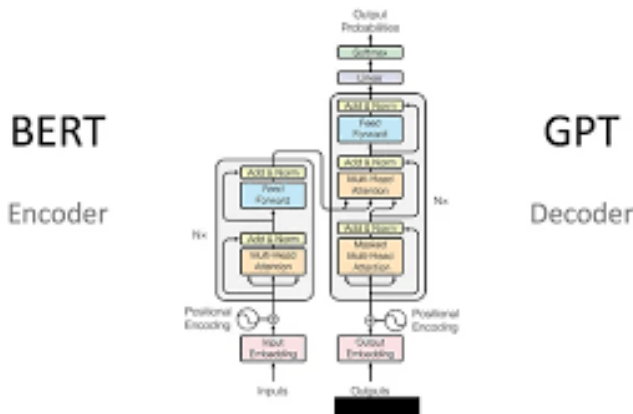


Figure 2.4: Transformers

8 Chatbots Definition

A chatbot (originally chatterbot) is a software program or application designed to simulate or engage in conversation with human users. These interactions typically occur through text or voice interfaces/channels, often as mobile applications or website plugins, allowing users to interact with the application or provide on-site services. Chatbots are a well-known example of Artificial Intelligence (AI) and automation, though not all of them are equipped with AI. Their purpose is to answer questions, complete tasks, automate simple interactions, and provide services such as customer support or information retrieval. Chatbots are designed to allow humans to interact with digital devices as if they were communicating with a real person. They can be general (responding to all questions) or specific (trained on domain-specific data). While simpler chatbots may rely on Natural Language Processing (NLP) to analyze, understand, and respond to queries, the sources indicate that modern or advanced chatbots increasingly use NLP, Natural Language Understanding (NLU), and Machine Learning (ML) techniques to understand user intent and context. Somewhat advanced chatbots are distinguished by their ability to analyze and understand natural language inputs, while more advanced Generative AI chatbots can generate new content and classify text/inputs.

.This figure represent general chatbot architecture sequence diagram :

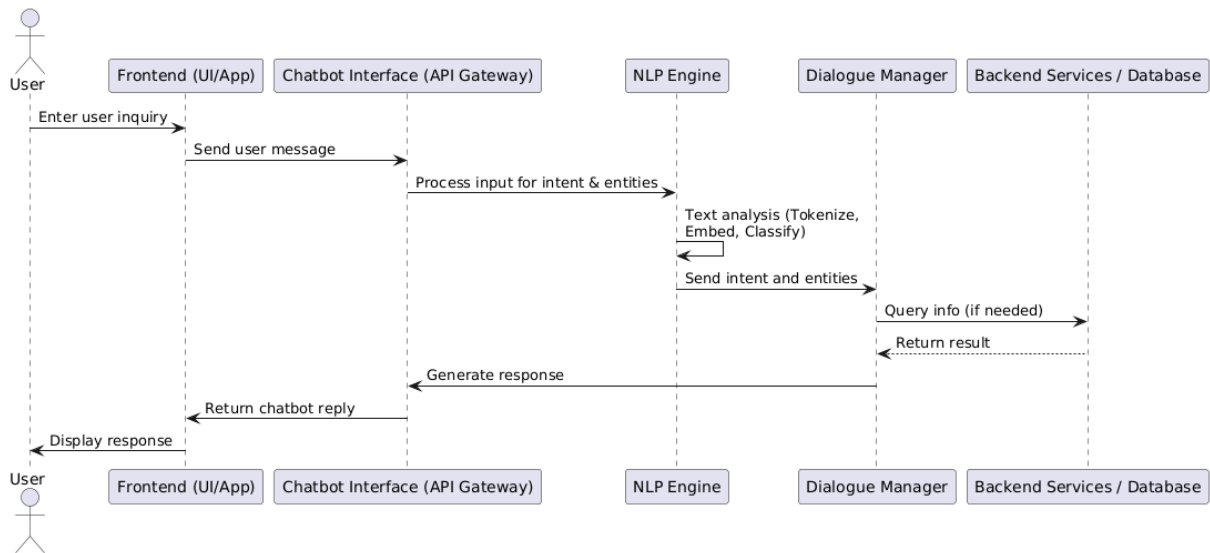


Figure 2.5: General chatbot architecture sequence diagram

9 Chatbots Types

Chatbot software can be classified based on various criteria, but one of the most prominent distinctions is rooted in their underlying technology and how they process input and generate responses. This fundamental classification helps highlight their capabilities and limitations [21]:

- Menu or Button-Based Chatbots :** These are the simplest type. They interact with users via the selection of pre-determined options or buttons presented in a menu, essentially via a decision tree. They can work effectively for transactional applications or answering repetitive easy questions, but do not meet advanced requests because they lack a free text input field. They might be slower to acquire needs if users have to click through many choices.
- Rule-Based Chatbots:** Building on the decision tree concept, these chatbots operate based on a manual set of predefined rules, scripts, or if/then logic. They recognize keywords or the lexical form of input text to select responses from a fixed, humanly hand-coded database of rules. They are relatively easy and fast to set up and customize. However, they are not robust to spelling or grammatical mistakes, struggle with complex queries, typically don't learn from interactions, lack personalization, and can get stuck when they don't understand the user's request, leading to frustration. They are often called Decision-Tree based chatbots and are suitable for simple campaigns or basic support .

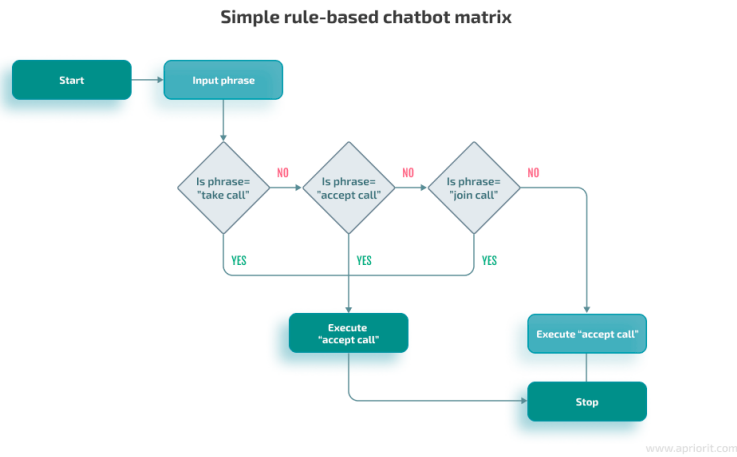


Figure 2.6: Rule-based-chatbot

- AI-Powered Chatbots (NLP-based / Conversational AI):** These chatbots leverage machine learning, Natural Language Processing (NLP), and Natural Language Understanding (NLU) to analyze and understand the intent, context, and meaning of natural language inputs. This allows them to understand questions regardless of phrasing, provide quick and accurate responses, and offer a top-notch conversational experience. They can continuously learn from user interactions, improving accuracy over time and reducing manual intervention. They provide personalized and contextual responses, maintain more natural interactions, handle a wider range of conversations, and can ask clarifying questions when unsure. They carry a risk of not understanding inputs but learn from these instances. They are also known as contextual chatbots or virtual agents. They may require more initial time and resources for setup and training.

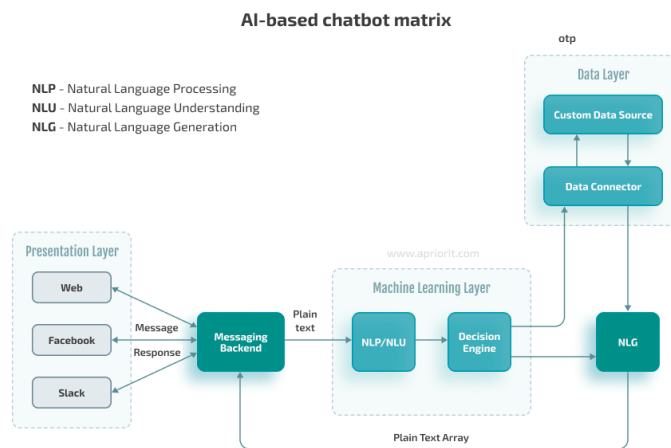


Figure 2.7: Ai-chatbot

Finally, chatbots can be classified according to several key criteria, including their domain of knowledge, the type of service they provide, their intended goals, the methods used to process input and generate responses, the level of human assistance involved, and how they are built.

When classified by domain of knowledge, chatbots are characterized by the range of information they can access or the volume of data they have been trained on. Open-domain chatbots are capable of engaging in general conversations across a wide range of topics, while closed-domain chatbots are designed to operate within a specific domain of knowledge and may struggle to handle queries outside of that domain.

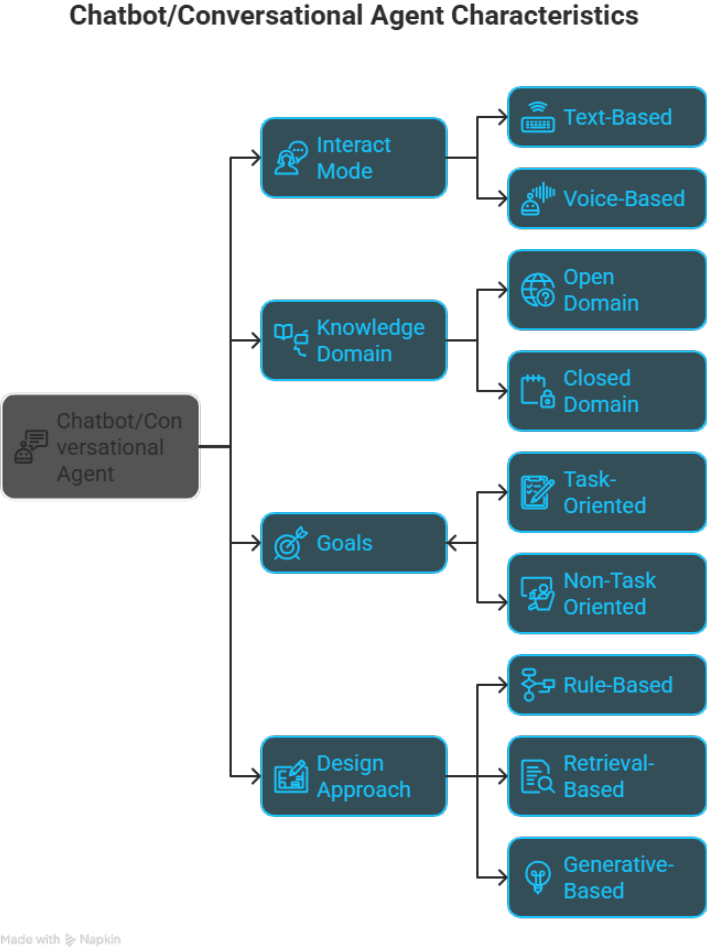


Figure 2.8: Chatbot-Categories

10 Chatbots benefits in buisness

Chatbots offer numerous benefits for both businesses and customers, transforming interactions and improving efficiency :

10.1 For customer

- **24/7 customer service** Chatbots are always available, eliminating wait times and providing instant support even outside business hours or across different time zones .

-
- **Deliver multilingual support** Deliver multilingual support Modern chatbots can communicate in multiple languages
 - **Enhanced Customer Experience** Through fast communication, personalization, and rapid response, chatbots help build positive relationships and increase customer satisfaction.

10.2 For businesses

- **Task Automation :** Chatbots can automate routine and repetitive tasks, such as answering frequently asked questions, providing product information, processing orders, tracking deliveries, handling returns, or scheduling appointments
- **Cost Reduction/Effectiveness** Implementing chatbots can be more cost-effective than hiring and maintaining a large support staff, as bots can handle a high volume of queries simultaneously
- **Lower Employee Churn** By automating repetitive tasks, deflecting tickets, and streamlining workflows, chatbots reduce the workload on human employees, allowing them to focus on more complex and strategic tasks, which can lead to improved employee experience and satisfaction
- **Increased customer retention** Ongoing dialogue, personalized offers, and helpful tips delivered by digital systems can increase customer loyalty and retention rates

11 Platform to create your chatbot

An AI agent builder is a software platform designed to help users create, deploy, and manage AI agents. These builders provide the necessary framework, infrastructure, and components, often making it possible to build AI agents without starting from scratch and without needing extensive coding or development knowledge. The goal is to make it faster and easier to implement agentic systems within workflows, and this the most popular no code platform to build your AI Agent :

- **Jotform AI Chatbot Builder :** explicitly stated as requiring no coding. You can transform existing forms or build new ones from scratch or using templates, then train the AI with documents or URLs and customize it using a visual Agent Builder. It's designed to streamline and automate form-related processes for businesses.

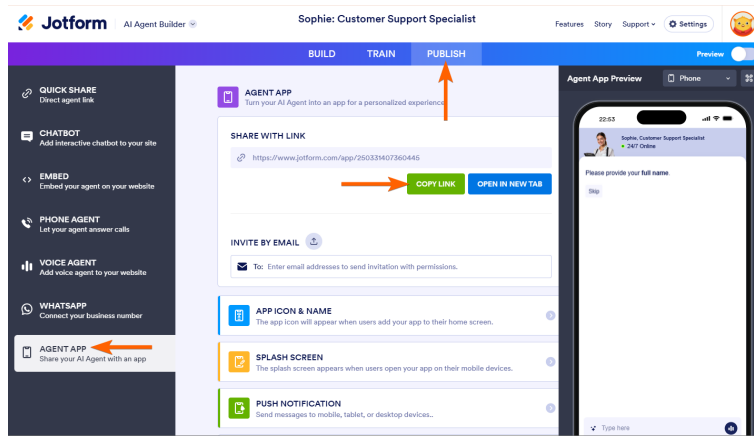


Figure 2.9: Jotform

- **WotNot:** This is a no-code AI agent builder that features a drag-and-drop interface that doesn't require any coding. It's specifically aimed at businesses looking to create chatbots for lead generation and customer support on their websites



Figure 2.10: WotNot

- **CustomGPT.ai:** This platform is a no-code AI chatbot platform built for businesses, emphasizing seamless knowledge integration and automation. Key features include "No-Code, Fully Customizable," making it simple to set up while being adaptable. It supports a vast number of data formats for accurate, context-aware responses .

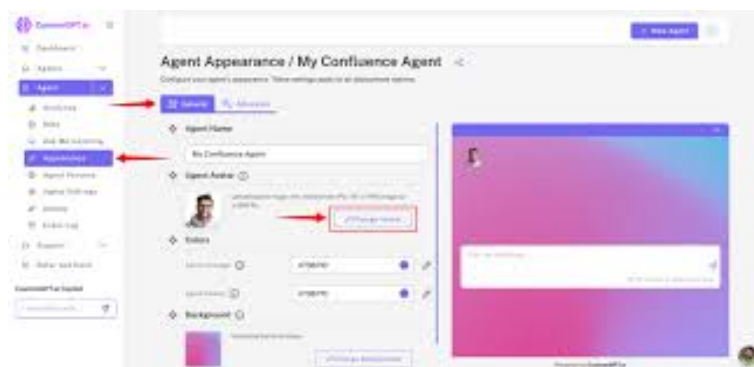


Figure 2.11: CustomGPT

- **Microsoft Copilot Studio:** This platform offers a no-code and low-code development environment for organizations to build chatbots and AI agents using generative AI. It provides an easy-to-use interface, even without coding experience.

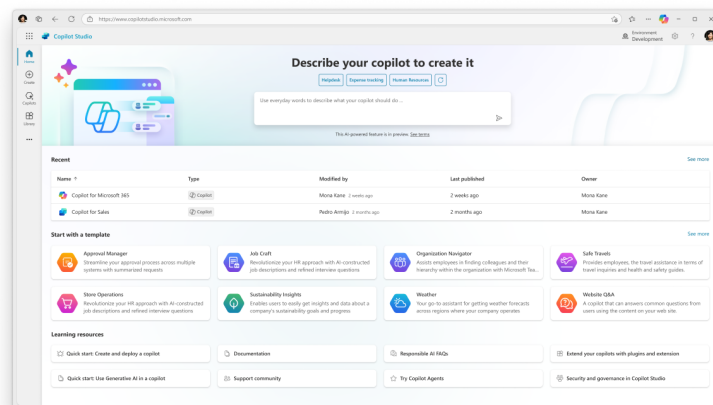


Figure 2.12: Microsoft Copilot Studio

12 No code disadvantages

Although these platforms facilitate the process of building a smart agent to interact with customers, they have drawbacks, the most important of which we will mention below:

- **Privacy and Security Concerns** AI agents often handle sensitive business data, so it's crucial to choose platforms with strong privacy controls, role-based access .
- **Technical Barriers (Despite Being "No-Code"):** While these platforms aim for ease of use, some may still require a level of technical understanding .
- **Limited Flexibility and Complexity Handling :** Some platforms are better suited for simpler tasks, and handling complex questions or workflows can be a challenge.
- **Training Data and Model Challenges:** Training AI assistants effectively requires preparing, sourcing, and minimizing bias in datasets. Complex tasks often require thousands or millions of data points. AI assistants can also be prone to issues like bias from unbalanced data, the "black box" problem (difficulty understanding why decisions were made), overfitting (struggling with new scenarios), and unintended actions if human oversight is insufficient.

13 Conclusion

In conclusion, this chapter has demonstrated that artificial intelligence (AI) provides the framework for developing intelligent systems, with natural language processing (NLP) being a vital

branch focused on enabling computers to process and understand human language. Modern NLP capabilities have been greatly enhanced by machine learning (ML) and deep learning (DL), using advanced models such as transformers. The basic processes of natural language understanding (NLU) and natural language generation (NLG), supported by various NLP techniques, are fundamental to their applications. A prime example of the impact of NLP is a chatbot, designed to mimic human conversation. While there are various types of chatbots, AI-powered chatbots utilize natural language processing (NLP), natural language processing (NLU), and machine learning (ML) to achieve more sophisticated understanding and interaction. Chatbots offer significant benefits to both businesses and users, including 24/7 service and automation. The emergence of no-code platforms has made chatbot creation much easier. Despite these advantages, challenges remain related to security, handling complexity, and training data.

Chapter 3

Optimizing Conversational AI with RAG and LLMs

1 Introduction

Software development projects exhibit considerable variation concerning their scope, complexity, duration, and application domain. Generally, a software project can be delineated as a systematic and coordinated endeavor directed towards the attainment of a precisely articulated objective, utilizing suitable technologies and methodologies. In this chapter, we elucidate the practical realization of our proposed system—an artificial intelligence-enhanced customer support application specifically engineered for the Android platform. The development procedure entailed a sequence of methodical phases, commencing with the delineation of hardware and software prerequisites, followed by the judicious selection of pertinent programming languages, development tools, and machine learning methodologies that are congruent with the nature of the task at hand. This chapter furnishes a thorough examination of the technical environment within which the system was constructed, along with the Natural Language Processing (NLP) techniques employed to analyze and address user inquiries within an e-commerce framework. Additionally, we provide illustrative screenshots and outputs from the system to effectively demonstrate the operational capabilities and features of the implemented application.

2 Technique used in our project

In the context of our research, we implement the Retrieval-Augmented Generation (RAG) methodology. RAG represents a sophisticated framework that combines the advantages of information retrieval with natural language generation. This methodology facilitates the system's capability to extract pertinent information from a knowledge repository and employ it to formulate responses that are not only more accurate but also contextually relevant. Such a strategy markedly enhances the efficacy of our chatbot by equipping it to deliver precise and informative responses grounded in actual data.

2.1 Large Language Model

Large Language Models (LLMs) represent advanced artificial intelligence frameworks predicated on transformer architectures that assimilate knowledge from extensive repositories of textual data. These models employ self-attention mechanisms to comprehend linguistic patterns and produce responses that closely resemble human discourse, thereby rendering them indispensable instruments for both natural language comprehension and generation. As a core component of Natural Language Processing (NLP), LLMs significantly enhance the ability of machines to understand, interpret, and interact with human language in a wide range of applications.[22]

2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) represents a sophisticated methodology designed to augment the capabilities of Large Language Models (LLMs) by incorporating external knowledge sources. The primary motivation behind RAG stems from the recognition that a vast portion of the world’s data is private or proprietary, while LLMs are largely trained on publicly available datasets.[22]

At its core, RAG addresses major limitations of LLMs, such as their restricted temporal knowledge and their tendency to produce “hallucinations”—factually incorrect information presented with undue confidence. By combining the intrinsic knowledge of LLMs with large, continually updated external data sources, RAG enables models to generate responses that are not only current and accurate but also capable of leveraging proprietary datasets and accessing contextual insights absent from their original training data.

RAG can be understood as a three-stage process:

- **Retrieval** : The system initiates a search across external knowledge repositories to identify information relevant to the user’s inquiry.
- **Augmentation** The retrieved information is then integrated into the original user prompt, serving as contextual enrichment to guide the LLM’s reason
- **Generation** Using the enhanced prompt—which includes both the query and the associated retrieved context—the LLM generates a response grounded in the newly acquired information.

This structured approach makes the retrieval and reasoning phases explicitly transparent, offering clarity that is typically lacking in responses produced solely from a model’s internal knowledge base.

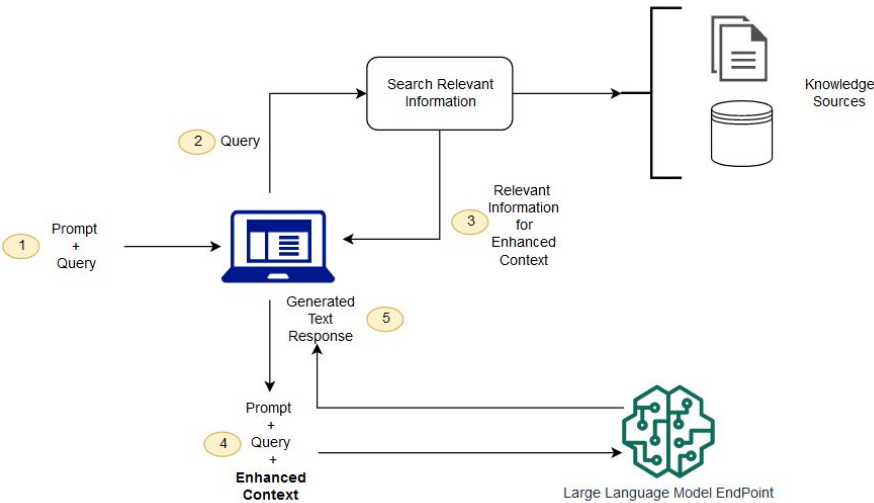


Figure 3.1: RAG Technique

3 Hardware used

The development and testing of the AI Customer Support application were carried out using a single physical laptop and two mobile devices—one operating as a virtual emulator and the other as a physical testing device. The development environment relied on:

- laptop DELL Latitude 7480 equipped with an Intel® Core™ i5-6300U CPU @ 2.50 GHz and 16 GB of RAM, running the Windows 10 (64-bit) operating system.
- A virtual emulator configured to simulate a Google pixel 5 Lite running Android 12.0.0.
- A physical smartphone, specifically the Redmi Note 8 Pro, also running Android 12.0.0, used for real-device testing.

This setup provided sufficient diversity in hardware and software conditions to ensure the application’s functionality and performance across both virtual and physical Android environments.

4 Tool used

4.1 Programming language

The development of the AI Customer Support system involved the use of the following programming languages, each chosen for its suitability to a specific layer of the application architecture:

- **Dart** : Used for building the mobile application interface through the Flutter framework and Provides a reactive and performant user experience across Android devices.
- **Python** : Employed for implementing Natural Language Processing (NLP) tasks, including intent classification and response generation.

4.2 Framework and Libraries

frameworks

- **FLask** : Used to develop a lightweight web API to serve the trained NLP model and handle user inquiries from the mobile application.
- **Flutter** : Used for building the cross-platform mobile user interface.

Libraries

In our chatbot system, we relied on a number of essential Python libraries and frameworks, categorized by their roles:

- **LangChain Libraries:**

- `Langchain.schema.Document` – For wrapping raw text as document objects.
- `Langchain_text_splitters.RecursiveCharacterTextSplitter` – To split documents into overlapping chunks.
- `Langchain.embeddings.OllamaEmbeddings` – For generating text embeddings using a local or hosted model.
- `Langchain.vectorstores.FAISS` – To build and query a FAISS-based vector store.

- **Vector Store Library:**

- `FAISS` (via `LangChain`) – A high-performance similarity search library for dense vectors.

- **OpenAI-Compatible API Client:**

- `Openai.OpenAI` – Used to send chat completion requests to the OpenRouter API.

- **Flask Framework:**

- `Flask.Flask`, `flask.request`, `Flask.jsonify` – For building the backend REST API.

- **Python Standard Libraries:**

- `Os` – File system operations.
- `Json` – Parsing and encoding JSON content.
- `RE` – Regular expressions for cleaning model outputs.

4.3 Software and API

We use range of software to develop our AIagents and web site to get an API:

In the development of our intelligent chatbot system, we used several key tools, each serving a specific purpose in the workflow:

Software

- **VS Code:** Used as the main code editor and development environment for writing and managing Python, Flask, and frontend code efficiently.
- **Ollama:** Utilized to run local large language models (LLMs) and generate text embeddings, which are used to encode textual data for vector search.

-
- **Android Studio:** Employed for building and testing the Android version of the chatbot interface developed using Flutter.
 - **Anaconda:** Used to manage Python environments and packages in an organized way, ensuring compatibility and reproducibility across development stages.
 - **Jupyter Notebook:** Served for prototyping, testing data loading, embedding generation, and interacting with model outputs in a flexible, visual format.

Application Programming Interface (API)

- **OpenRouter:** Used as the interface to access and interact with advanced Large Language Models (LLMs) such as *DeepSeek-R1*. This API allows us to send user inquiries along with relevant context retrieved from our vector database and receive structured, natural language responses. It plays a key role in enabling the chatbot's reasoning and language generation capabilities.

5 Project Implementation

We employed a chatbot system based on Retrieval-Augmented Generation (RAG). Large language models (LLMs) and retrieval systems are used in RAG-based chatbots to deliver precise and contextually aware responses. Rather than merely using a predetermined list of questions and answers, the chatbot looks through a knowledge base to find pertinent documents or entries, then utilizes the LLM to produce an organic and educational response based on the information it has found.

To enhance accessibility and user experience, the chatbot system supports multilingual interactions. It is capable of understanding and responding to user inquiries in **Arabic, English and French**. This multilingual capability is essential for serving a diverse user base in Algeria, where these three languages are commonly used in both daily communication and technical contexts. The retrieval and generation components are designed to work seamlessly across all supported languages, ensuring that users receive accurate and context-relevant answers regardless of the language used.

The key elements and their functions are broken down as follows:

5.1 Knowledge Datasets Preparation

For our intelligent chatbot system, we used a custom knowledge dataset derived from a virtual store simulation. The dataset represents the full scope of services and product offerings of a real-world enterprise — AymenTech Store — a comprehensive technology service provider located in Msila, Algeria. This store caters to a broad clientele including individuals, professionals, and

businesses across all 58 provinces in Algeria. The dataset was manually structured to reflect real business knowledge that can be queried by users in natural language.

The knowledge base includes structured descriptions of core services and product categories offered by the store. These services include computer hardware sales, custom PC assembly, professional repair and maintenance, software installation, networking solutions, and cloud service deployment. It also contains financial services such as Mastercard creation and recharge support, online transaction guidance, and eCommerce facilitation, , It is stored in a "knowledgs.txt file".

The dataset was designed to support retrieval-based and generative responses. For example, it includes detailed entries on:

- **Hardware Components:** CPU, GPU, RAM, SSD/HDD, power supplies, and PC cases from brands like Intel, AMD, NVIDIA, Corsair, and Kingston.
- **Technical Services:** Data recovery, motherboard repairs, cooling optimization, BIOS updates, and thermal paste application.
- **Software and OS:** Installation and configuration of Windows, Linux, macOS, drivers, antivirus, and backup tools.
- **Networking Solutions:** Wired/wireless networks, router setup, LAN design, NAS configuration, and server deployment.
- **Cloud Licensing:** Integration of services like Google Workspace, Microsoft 365, Adobe, AutoCAD, and cybersecurity software.
- **Accessories:** Monitors, mechanical keyboards, gaming chairs, printers, and content creation tools.
- **Financial Services:** Secure payment setup using virtual/physical Mastercards and COD-based deliveries.
- **Customer Support Programs:** Loyalty discounts, remote troubleshooting, and nationwide delivery with warranty support.

This dataset forms the foundation for the chatbot's retrieval mechanism. When a user submits an inquiry, relevant chunks of this knowledge base are retrieved and passed to the language model to generate a coherent and helpful response. This hybrid RAG approach ensures both coverage of domain-specific knowledge and flexibility in language understanding.

```
data > Knowledge.txt
1 Company Overview
2 AyemTech Store is a comprehensive technology service provider based in Msila, Algeria. The company specializes in
3 Core Services
4 Hardware Sales and Custom PC Assembly
5 Supply of original components from reputable global brands including Intel (i3/i5/i7/i9), AMD Ryzen (5/7/9), NVIDIA
6 Custom-built PCs optimized for gaming, design, development, and office use.
7 Expert installation with professional cable management and advanced cooling solutions.
8 Technical Maintenance and Repairs
9 Maintenance of desktops and laptops including:
10 Thermal paste replacement
11 Cooling system cleaning
12 Screen and keyboard replacement
13 Motherboard repair and BIOS updates
14 Hardware diagnostics and data recovery
15 Software and Operating System Services
16 Installation of Windows, Linux, and macOS operating systems.
17 Setup of essential drivers and software.
18 Antivirus Installation and malware protection.
19 Backup configuration and data restoration services.
20 Networking Solutions
21 Home and enterprise networking setup:
22 Router configuration
23 Wired and wireless network setup
24 Structured cabling and network hardening
25 Local area network (LAN) design for small businesses
26 Internal server configuration, including NAS, Windows Server, and Linux Server systems.
27 Cloud and Licensing Services
28 Deployment of cloud-based services including Google Workspace and Microsoft 365.
29 Provision of genuine software licenses for:
30 Adobe Creative Suite
31 AutoCAD
32 Cybersecurity solutions (e.g., Bitdefender, Kaspersky)
33 Products and Accessories
```

Figure 3.2: Dataset Used

5.2 Code Implementation and Methodology

In this section, we explain the methodology and implementation details used to build our intelligent chatbot system. The entire development process is divided into three major steps:

1. Loading and Processing Data
2. Embedding and Vector Storage
3. Query Handling and Response Generation

Each step is critical to ensure that the chatbot effectively retrieves relevant knowledge and generates accurate responses based on user inquiries.

Step 1: Loading and Processing Data

The first step involves reading and preparing the data from a text-based knowledge source. This content is then transformed into smaller, manageable chunks suitable for vector storage and retrieval.

The following Python function demonstrates how we load and process the data:

```
1 def readdata(file_path):
2     with open(file_path, "r", encoding="utf-8") as f:
3         full_text = f.read()
4
5         document = Document(page_content=full_text.strip())
6
7         splitter = RecursiveCharacterTextSplitter(
8             chunk_size=700,
9             chunk_overlap=200
10        )
11    return splitter.split_documents([document])
```

Figure 3.3: Loading-data

Explanation:

- The data file is read entirely into memory and stripped of unnecessary whitespace.
- A Document object is created to represent the text in a structured format suitable for LangChain processing.
- The RecursiveCharacterTextSplitter is used to divide the text into overlapping chunks. This technique helps maintain context between chunks during retrieval.
- The chunk size is set to 700 characters with an overlap of 200, which balances performance with contextual relevance.

This preprocessing ensures that the knowledge base is properly segmented, allowing the retrieval system to later identify and return the most contextually relevant pieces of information when a user submits an inquiry.

Step 2: Embedding and Creating the Vector Database

Once the dataset is loaded and split into chunks, the next step is to convert those text chunks into high-dimensional vector representations using a language model embedding. These embeddings are stored in a vector database to allow efficient similarity search during user interactions.

The following code demonstrates how this process is implemented:

```
1 documents = readdata("data/Knowledgs.txt")
2 embeddings = OllamaEmbeddings(model="nomic-embed-text:latest")
3
4 def create_vector_store():
5     db = FAISS.from_documents(documents, embeddings)
6     os.makedirs(os.path.dirname(VECTOR_STORE_PATH), exist_ok=True)
7     db.save_local(VECTOR_STORE_PATH)
8     print(f"✅ Vector store saved to {VECTOR_STORE_PATH}")
9
10 def load_vector_store():
11     return FAISS.load_local(
12         folder_path=VECTOR_STORE_PATH,
13         embeddings=embeddings,
14         allow_dangerous_deserialization=TRUSTED_SOURCE
15     )
```

Figure 3.4: Embeddings and create vector database

Explanation:

- OllamaEmbeddings is used to generate vector representations for each document chunk.
- The function FAISS.from-documents creates an index of vectors from the processed documents.
- The vector store is saved locally using the function db.save-local().
- The function load-vector-store() is used to reload the vector store when the application starts.

This embedding and storage process is essential for enabling similarity search. When a user submits an inquiry, the chatbot can quickly retrieve the most relevant chunks by comparing the semantic similarity between the user input and the stored vectors.

Step3: LLMs Integration

In this final step, we integrate large language models (LLMs) into the chatbot to generate responses based on relevant context. Instead of relying on static responses, we use prompt engineering to instruct the LLM to answer using retrieved content from the vector store.

The method involves the following:

- When a user submits a question, the function searches for the most relevant content using vector similarity (top 3 results).
- A system prompt is created to instruct the model to act as an expert assistant for Ay-menTech, answering based only on the given context and replying in JSON format.
- The user's inquiry and the retrieved documents are combined in a single prompt sent to the LLM.
- We use the DeepSeek-R1 model hosted on OpenRouter to generate the answer.

- The response is cleaned to remove any unwanted tags or formatting.
- If the response is not valid JSON, it is safely wrapped to maintain consistency.

This approach enables the chatbot to respond in a natural, accurate, and multi-language manner (Arabic, English, or French) depending on the user's inquiry. The combination of retrieval with LLM generation ensures contextual and reliable answers aligned with AymenTech's services and knowledge base.

Server creation

in this step involves creating a server to host and expose the chatbot interface. We use Flask, a lightweight Python web framework, to create a simple REST API endpoint.

- The server loads the previously created vector store using the function `load_vector_store()`.
- A single route `/chat` is defined to handle POST requests, where the user's inquiry is received in JSON format under the key `"message"`.
- The inquiry is passed to the `generate_response_from_vector_db()` function, which performs retrieval and calls the LLM to generate an appropriate response.
- The server ensures the returned content is a valid JSON structure. If parsing fails, a fallback mechanism wraps the output as a string.
- The final response is sent back to the client as JSON, making the chatbot system easily consumable by any frontend or external app.



```
1 @app.route("/chat", methods=["POST"])
2 def chat():
3     data = request.get_json()
4     query = data.get("message", "")
5
6     if not query:
7         return jsonify({"error": "No query provided"}), 400
8
9     response = generate_response_from_vector_db(query, db)
10
11     if isinstance(response, str):
12         try:
13             response_list = json.loads(response)
14             if isinstance(response_list, list) and len(response_list) > 0:
15                 return jsonify(response_list[0])
16             else:
17                 return jsonify({"answer": "Unexpected response format"}), 500
18         except json.JSONDecodeError:
19             return jsonify({"answer": response})
20     else:
21         return jsonify(response)
22
23 if __name__ == "__main__":
```

Figure 3.5: Api code

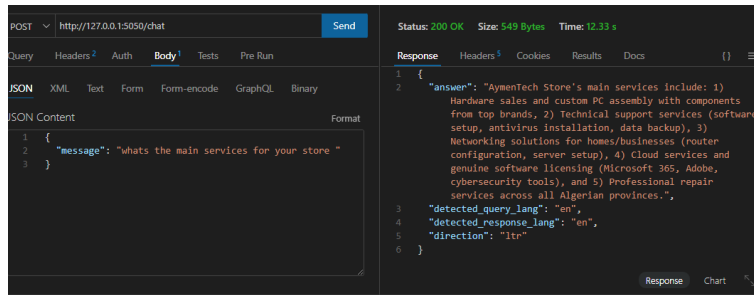


Figure 3.6: API Request and Response Result from the Chatbot Server

This architecture allows the chatbot to run as a web service, making it accessible for integration with web or mobile interfaces. The server listens on port 5050 and is configured to accept connections from all IP addresses.

5.3 Chatbot Interface Explanation

We integrated our intelligent chatbot into a virtual online store to assist users with their inquiries in real-time. The chatbot interface allows customers to ask questions about products, services, availability, prices, and more. It is designed to provide fast, accurate, and multilingual responses (Arabic, English, and French) using a user-friendly chat-based interface. This integration enhances user experience, reduces support workload, and ensures that customers receive consistent and informative answers directly within the store environment.

Chatbot Interface Integration

A dedicated section was added to the bottom navigation bar of the online store interface to allow users to easily access the chatbot from any page.

The image on the right displays the homepage of the virtual online store, where the chatbot access icon is clearly visible at the bottom of the screen.

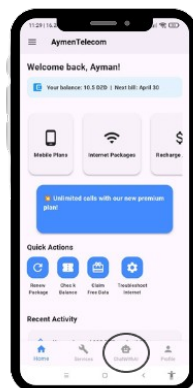


Figure 3.7: Chatbot Interface in the Online Store

English Conversation Interface

The chatbot interface supports fluent English conversations, allowing users to interact naturally and receive accurate responses.

This interface is designed to be user-friendly, providing quick access to assistance on products, services, or support queries in English. The system also ensures proper language understanding and context-aware answers through integration with large language models (LLMs).

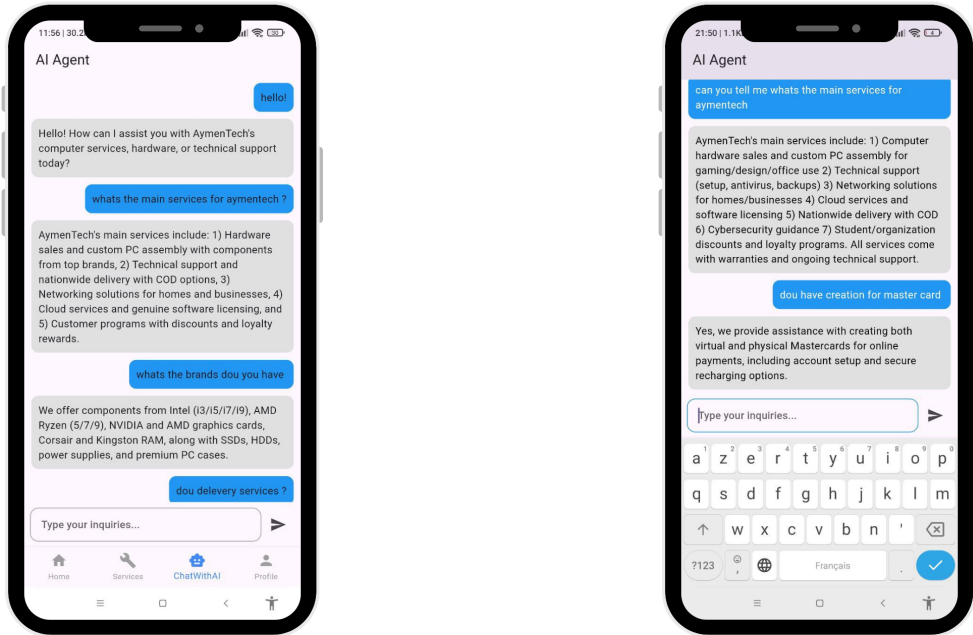


Figure 3.8: English Chatbot Conversation Interface

Arabic Chatbot Conversation Interface

The chatbot interface enables smooth Arabic conversations, offering quick, accurate responses to product and service inquiries through an easy-to-use design powered by language models.



Figure 3.9: Arabic Chatbot Conversation Interface

Multilanguage Chatbot Conversation Interface

We can in the same conversation interact with chatbo use the Arabic and English and Frensh .

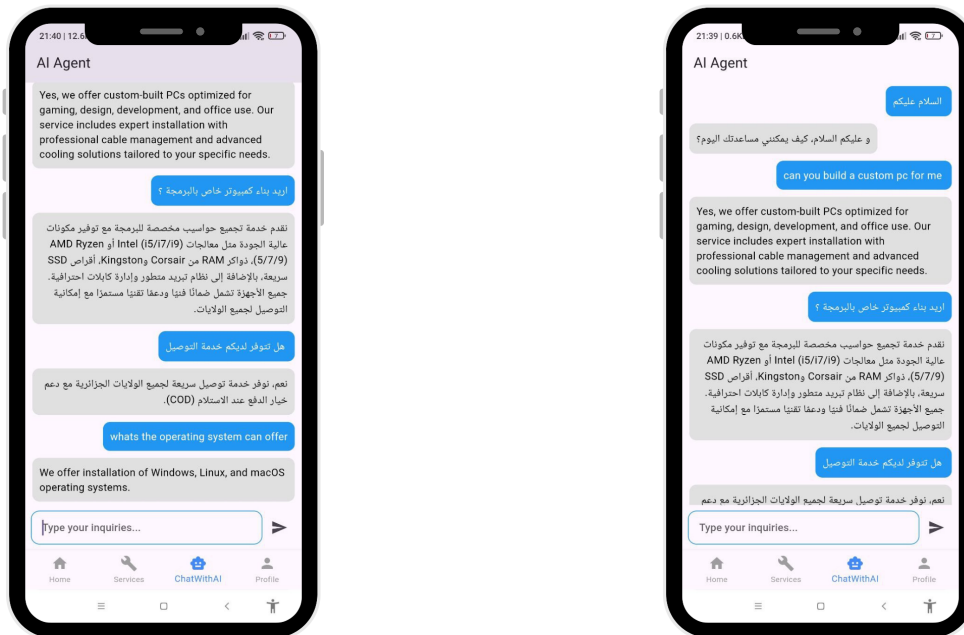


Figure 3.10: Multi language

5.4 Evaluation and Testing

Our multilingual chatbot evaluation relied on a semantic similarity-based methodology to assess the quality of responses across three languages: Arabic, French, and English. Instead of relying on strict string matching, we applied the MiniLM-L12-v2 paraphrase model to calculate the cosine similarity between the generated responses and the corresponding reference answers, taking into account that multiple semantically equivalent expressions may constitute correct responses. The evaluation dataset included 100 carefully selected queries, distributed across the three target languages, with each generated response classified as correct or incorrect according to a similarity threshold of 0.75. This threshold was set to strike a balance between accepting semantically equivalent responses and maintaining quality standards. The methodology achieved an overall accuracy of 78.2%, demonstrating the system's ability to understand and generate answers. This similarity-based approach is clearly superior to traditional exact matching metrics for evaluating natural language generation tasks, especially in multilingual contexts where semantic equivalence may manifest across a variety of linguistic structures, thus providing a more objective assessment of a chatbot actual effectiveness in understanding and processing user queries across diverse languages.

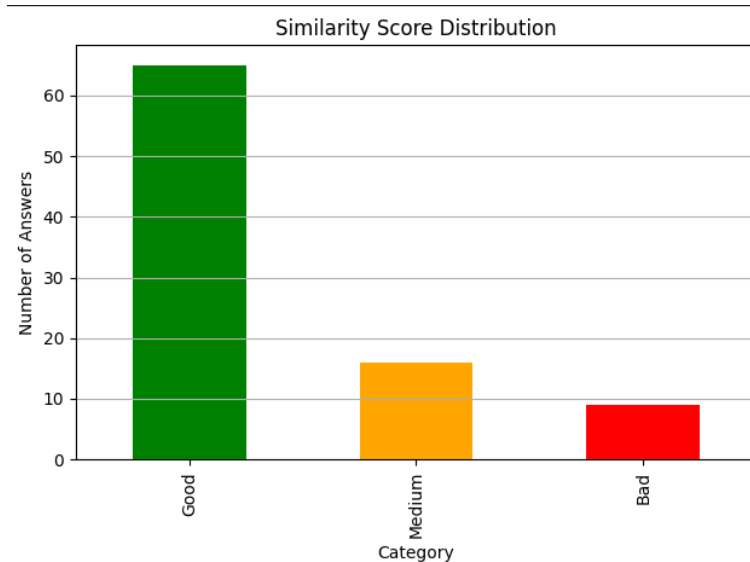


Figure 3.11: Similarity Score Distribution

6 Future Work and Enhancements

To further enhance the performance, adaptability, and user experience of the proposed RAG-based chatbot system, several areas for future development are identified. These enhancements are aimed at improving the chatbot’s reliability, scalability, and intelligence in serving users across various domains.

6.1 Interaction Logging and Analysis

Future versions of the system can include detailed logging of user interactions and chatbot responses. This data can be used for analytical purposes, such as identifying frequently asked questions, improving the quality of the knowledge base, and fine-tuning model performance. Additionally, an administrative dashboard could be developed to monitor and analyze chatbot usage.

6.2 Intent Detection Model

Incorporating a dedicated intent classification model can improve the system’s ability to understand user queries more effectively. By detecting user intent (e.g., inquiry about a product, complaint submission, or order tracking), the chatbot can better tailor its responses. This can be achieved using pre-trained transformers such as BERT or DistilBERT, fine-tuned on labeled intent datasets.

6.3 Real-Time Database Integration

Integrating the chatbot with a live backend database would allow it to retrieve dynamic and real-time information, such as product availability, stock levels, and pricing. This can significantly enhance the relevance and accuracy of the chatbot's responses, especially in an e-commerce setting.

6.4 Fine-Tuning of Language Models

Fine-tuning the underlying language model using domain-specific datasets can improve its performance in specialized tasks. Techniques such as supervised fine-tuning or parameter-efficient tuning (e.g., LoRA) may be employed to adapt general-purpose models to the specific knowledge and terminology of the application domain.

6.5 Voice Interaction Capabilities

To increase accessibility and user engagement, voice interaction features can be introduced. These include speech-to-text (STT) for voice input and text-to-speech (TTS) for audio responses. Integrating APIs such as Whisper for STT and Google TTS or Amazon Polly for TTS would enable a fully voice-enabled chatbot interface.

6.6 Alerting and Notification System

An alerting system could be implemented to notify human agents or administrators when the chatbot receives urgent or critical inquiries. Integration with communication platforms such as email or messaging APIs would ensure that such inquiries are addressed promptly.

6.7 Dynamic Knowledge Base Updates

To maintain the relevance of responses, the system can be extended to support automatic updates of the knowledge base. File monitoring tools such as `watchdog` can be employed to detect content changes and trigger re-embedding processes without requiring manual intervention.

6.8 Administrative Knowledge Management Interface

An administrative interface for managing the knowledge base would provide a user-friendly environment for curating and maintaining question-answer pairs. This feature would be particularly useful for non-technical users tasked with updating domain-specific content.

These proposed enhancements aim to transform the current prototype into a robust, adaptive, and scalable AI-driven assistant. By incorporating these features, the chatbot will better align with user needs, real-time data requirements, and evolving language technologies.

7 Conclusion

This chapter describes an advanced chatbot application, specifically designed for our hardware-based e-commerce and computing services platform. The system utilizes comprehensive language models along with natural language processing techniques and a RAG-based architectural framework to generate contextually relevant responses. The system effectively meets the needs of Algerian users by enabling multilingual communication in Arabic, English, and French. In addition to improving customer service,

General Conclusion

The integration of AI-driven chatbots into e-commerce platforms represents a transformative advancement in customer support, merging scalability with personalized user engagement. This thesis presents the development of a multilingual chatbot designed to address product inquiries in online stores, with a specialized focus on the Algerian market. By leveraging state-of-the-art technologies—Retrieval-Augmented Generation (RAG) for dynamic data retrieval, Natural Language Processing (NLP) for semantic understanding, and Large Language Models (LLMs) for coherent response generation—the system delivers accurate, context-aware assistance in real time. These innovations demonstrate how automation can enhance efficiency without compromising the nuance of human interaction, ensuring both precision and adaptability in customer service.

The chatbot's architecture synthesizes RAG, NLP, and LLMs to create a robust framework for knowledge retrieval and response generation. By grounding answers in structured, up-to-date data, the system fosters user trust and reliability, while its multilingual capacity (supporting Arabic, English, and French) addresses Algeria's linguistic diversity, broadening accessibility. This dual emphasis on technical rigor and cultural relevance highlights the chatbot's ability to navigate complex linguistic and contextual landscapes, thereby elevating user satisfaction. Such capabilities underscore the importance of designing AI systems that are not only technologically advanced but also socially and linguistically inclusive.

Future enhancements could further refine the system's utility, such as real-time database synchronization for live inventory updates, advanced intent detection to streamline user queries, and voice-enabled interfaces to improve accessibility. Beyond e-commerce, the adaptability of this framework suggests promising applications in healthcare, education, and public administration, where multilingual, context-aware support is equally critical. In summary, this research exemplifies the potential of conversational AI to redefine digital customer service, offering a scalable and adaptive model for intelligent support systems that balance automation with human-centric design.

References

- [1] M. McTear, *Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots*. Springer, 2020. DOI: [10.1007/978-3-030-53959-7](https://doi.org/10.1007/978-3-030-53959-7).
- [2] O. W. Studio, *What is online store?* <https://www.oxfordwebstudio.com/en/did-you-know/what-is-online-store>, Accessed: 2023-11-15, 2023. [Online]. Available: <https://www.oxfordwebstudio.com/en/did-you-know/what-is-online-store>.
- [3] J. Almeida and M. Silva, "Spread and importance of e-commerce stores," *Journal of Digital Business*, vol. 3, pp. 100–115, 2023. DOI: [10.1016/j.sdb.2023.100015](https://doi.org/10.1016/j.sdb.2023.100015). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2666412723000156>.
- [4] W. Team, *The different types of e-commerce business models*, <https://www.wix.com/blog/types-of-ecommerce>, Accessed: June 29, 2025, 2024. [Online]. Available: <https://www.wix.com/blog/types-of-ecommerce>.
- [5] M. CRM, *What is customer support? the complete definition*, <https://www.momentcrm.com/blog/what-is-customer-support-definition/>, Accessed: June 29, 2025, 2025. [Online]. Available: <https://www.momentcrm.com/blog/what-is-customer-support-definition/>.
- [6] M. CRM, *The essential role of customer support in business success*, <https://www.momentcrm.com/blog/what-is-customer-support-definition/>, Accessed: June 29, 2025, 2025. [Online]. Available: <https://www.momentcrm.com/blog/what-is-customer-support-definition/>.
- [7] M. CRM, *Why customer support matters: The key to customer retention*, <https://www.momentcrm.com/blog/what-is-customer-support-definition/>, Accessed: June 29, 2025, 2025. [Online]. Available: <https://www.momentcrm.com/blog/what-is-customer-support-definition/>.
- [8] I. Think, *Customer support channels: The complete guide*, <https://www.ibm.com/think/topics/customer-support>, Accessed: June 29, 2025, 2023. [Online]. Available: <https://www.ibm.com/think/topics/customer-support>.

-
- [9] Zendesk, *Benefits of using ai customer service chatbots*, <https://www.zendesk.com/service/ai/chatbots-customer-service/>, Accessed: June 29, 2025, 2025. [Online]. Available: <https://www.zendesk.com/service/ai/chatbots-customer-service/>.
- [10] C. A. Nwafor, "Enhancing customer service and user experience through the use of machine learning powered intelligent chatbots," *World Journal of Advanced Research and*, 2024, Accessed: June 29, 2025. [Online]. Available: <https://www.researchgate.net/publication/383561289>.
- [11] S. Vajjala, B. Majumder, A. Gupta, and H. Surana, *NLP Tasks*. Sebastopol, CA: O'Reilly Media, 2020.
- [12] S. Nandika. "Natural language processing:applications of natural language processing." Accessed: 2025-05-29. (2025), [Online]. Available: <https://snatika.com/single-blog/natural-language-processing%3A-advancements--applications--and-future-possibilities>.
- [13] S. Nandika. "Natural language processing:applications of natural language processing." Accessed: 2025-05-29. (2025), [Online]. Available: <https://snatika.com/single-blog/natural-language-processing%3A-advancements--applications--and-future-possibilities>.
- [14] A. to NLP, *NLP Tasks*. Sebastopol, CA: O'Reilly Media, 2020.
- [15] S. Raschka, *Naive bayes and text classification i: Introduction and theory*, Accessed: 2025-05-29, 2014. [Online]. Available: https://sebastianraschka.com/Articles/2014_naive_bayes_1.html.
- [16] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proceedings of the 10th European Conference on Machine Learning (ECML)*, Chemnitz, Germany, 1998, pp. 137–142.
- [17] D. Jurafsky and J. H. Martin, *Speech and Language Processing*. 2025, Draft of January 12, 2025. Accessed: 2025-05-29. [Online]. Available: <https://web.stanford.edu/~jurafsky/slp3/>.
- [18] R. Venkatesh and M. Thangarathinam, "Recurrent neural networks for text classification," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 48, no. 5, pp. 253–257, Jun. 2017. [Online]. Available: <https://ijettjournal.org/archive/ijett-v48p253>.
- [19] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1746–1751.

-
- [20] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA, 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91f%5Cb053c1c4a845aa-Paper.pdf.
- [21] Apriorit, *Ai-based vs rule-based chatbots: How to choose the right one for your business*, Accessed: 2025-05-29, 2022. [Online]. Available: <https://www.apriorit.com/dev-blog/web-ai-vs-rule-based-chatbots>.
- [22] Amazon Web Services, *What is a large language model (llm)?* Accessed: 2025-06-08, 2024. [Online]. Available: <https://aws.amazon.com/what-is/large-language-model/>.

ملخص

شهد التجارة الإلكترونية تحولاً نوعياً بفضل دمج تقنيات الذكاء الاصطناعي، لا سيما روبوتات المحادثة الذكية التي تعزز تجربة المستخدم وتدعم العملاء بشكل فوري وفعال. يقدم هذا البحث تطوير روبوت محادثة متعدد اللغات، يعتمد على تقنية الجيل المعزز بالاسترجاع (RAG) ونماذج اللغة الكبيرة (LLMs) لتقديم إجابات دقيقة وسياقية باللغات العربية والفرنسية والإنجليزية. يساهم النظام في تقليل الضغط على فرق الدعم البشري، وزيادة رضا العملاء، مع إمكانية التحديث الديناميكي لقاعدة المعرفة. يُظهر النموذج قدرة عالية على التكيف، مما يجعله قابلاً للتطبيق في مجالات متعددة تتجاوز التجارة الإلكترونية.

Abstract

E-commerce is undergoing a paradigm shift thanks to the integration of artificial intelligence technologies, particularly intelligent chatbots, which enhance user experience and provide immediate and efficient customer support. This research presents the development of a multilingual chatbot, based on Retrieval Augmented Generation (RAG) technology and Large Language Models (LLMs), to provide accurate and contextual answers in Arabic, French, and English. The system contributes to reducing the burden on human support teams and increasing customer satisfaction, with the ability to dynamically update the knowledge base. The model demonstrates high adaptability, making it applicable to multiple fields beyond e-commerce.

Résumé

Le commerce électronique connaît un changement de paradigme grâce à l'intégration des technologies d'intelligence artificielle, notamment les chatbots intelligents, qui améliorent l'expérience utilisateur et offrent un support client immédiat et efficace. Cette recherche présente le développement d'un chatbot multilingue, basé sur la technologie de génération augmentée de recherche (RAG) et les grands modèles linguistiques (LLM), pour fournir des réponses précises et contextuelles en arabe, français et anglais. Ce système contribue à alléger la charge de travail des équipes de support et à accroître la satisfaction client, grâce à la mise à jour dynamique de la base de connaissances. Ce modèle fait preuve d'une grande adaptabilité, ce qui le rend applicable à de nombreux domaines autres que le commerce électronique.