Upsell Discovery by Analysing Customer Communication and Integration into the Amazon Bedrock Infrastructure

Šamanić, Lorenzo

Sveučilište u Rijeci
Fakultet informatike
i digitalnih tehnologija

University Graduate Study of Informatics

Lorenzo Šamanić

Upsell Discovery by Analysing Customer Communication and Integration into the Amazon Bedrock Infrastructure

Master thesis

Mentor: prof. dr. sc. Sanda Martinčić-Ipšić

Rijeka, July 2024.

Sveučilište u Rijeci
Fakultet informatike
i digitalnih tehnologija

Sveučilišni diplomski studij Informatika

Lorenzo Šamanić

Otkrivanje mogućnosti za prodaju dodatnih usluga putem analize komunikacije s klijentom na Amazon Bedrock infrastrukturi

Diplomski rad

Mentor: prof. dr. sc. Sanda Martinčić-Ipšić

Rijeka, srpanj 2024

Sveučilište u Rijeci
Fakultet informatike
i digitalnih tehnologija

Rijeka, 10.4.2024.

Zadatak za diplomski rad

Pristupnik: Lorenzo Šamanić

Naziv diplomskog rada: Otkrivanje mogućnosti za prodaju dodatnih usluga putem analize komunikacije s klijentom na Amazon Bedrock infrastrukturi

Naziv diplomskog rada na eng. jeziku: Upsell Discovery by Analysing Customer Communication and Integration into the Amazon Bedrock Infrastructure

Sadržaj zadatka: U ovom diplomskom radu istražuje se inovativni pristup identifikaciji mogućnosti za prodaju dodatnih usluga (upsell) kroz detaljnu analizu komunikacije između agencije i klijenata. Koristeći naprednu analitiku i tehnike strojnog učenja, rad se fokusira na automatizirano prikupljanje i analizu podataka iz komunikacijskih alata poput Teamwork i Slack, kao i iz transkripta sastanaka, kako bi se razumjele specifične potrebe klijenata koje trenutno nisu zadovoljene. Projekt se posebno oslanja na integraciju s Amazon Bedrock infrastrukturom, koja pruža temelje za skalabilnu i učinkovitu obradu i analizu velikih skupova podataka. Cilj je identificirati ključne momente u komunikaciji koji sugeriraju mogućnost prodaje dodatnih usluga (upsell), kao i poboljšati preciznost i relevantnost predloženih usluga kroz dublje razumijevanje konteksta i potreba klijenta. Ovaj pristup omogućava ne samo identifikaciju mogućnosti na temelju postojeće komunikacije, već i razvoj proaktivnih strategija prilagođavanja ponude usluga u realnom vremenu, čime se postiže veća personalizacija i zadovoljstvo klijenata, te se otvaraju novi putevi za rast prihoda.

Sadržaj zadatka na eng. jeziku: In this thesis, an innovative approach to the identification of opportunities for the sale of additional services (upsell) is investigated through a detailed analysis of communication between the agency and clients. Using advanced analytics and machine learning techniques, the work focuses on automated collection and analysis of data from communication tools like Teamwork and Slack, as well as meeting transcripts, to understand specific client needs that are not currently being met. In particular, the project relies on integration with the Amazon Bedrock infrastructure, which provides the foundation for scalable and efficient processing and analysis of large data sets.The goal is to identify key moments in communication that suggest the potential for an upsell and improve the precision and relevance of the proposed services through a deeper understanding of the context and needs of the client. This approach enables identifying upsell opportunities based on existing communication and developing proactive strategies for adjusting the service offer in real-time, achieving greater personalization and client satisfaction, and opening new avenues for revenue growth.

Mentor:
Prof. dr. sc. Sanda Martinčić-Ipšić

Voditeljica za diplomske radove:
Doc. dr. sc. Lucia Načinović Prskalo

Komentor:

Zadatak preuzet: 10.4.2024

(potpis pristupnika)

Radmile Matejčić 2,
51000 Rijeka, Hrvatska

T: +385 (0)51/584-700
E: ured@inf.uniri.hr
www.inf.uniri.hr

OIB: 64218323816
IBAN: HR1524020061400006966

Abstract

This thesis explores an innovative approach to identifying opportunities for selling additional services (upsell) through a detailed analysis of communication between the agency and clients. The aim is to use advanced generative AI techniques for automated data collection and analysis from communication tools such as Teamwork and Slack, as well as from meeting transcripts. The focus is on understanding clients' specific needs that are currently unmet.

The implementation began with integrating Slack as the primary platform for collecting communication data. A custom bot was developed to connect to various Slack channels, systematically extracting messages and user interactions. This bot also provided real-time feedback by posting analysis results back into the channels, ensuring that the team remained informed and engaged. The collected data included detailed communication logs and user profiles, all stored in Amazon Aurora for structured management and analysis. In parallel, Teamwork was utilized for its robust project management capabilities. Custom fields were added to link projects in Teamwork with their corresponding Slack channels. Teamwork provided detailed project data, including timelines, task progress, and milestones. This data was also stored in Amazon Aurora, ensuring a centralized and unified repository for all project-related information. A key aspect of the implementation was the use of Amazon Bedrock, which facilitated the deployment of the Anthropic Claude 3 model. This advanced language model was used to analyze the collected data, leveraging Amazon's Titan embedding algorithm to transform the data into meaningful vectors. The model's capabilities were crucial in identifying patterns and extracting insights that indicated potential upsell opportunities. Creating an effective prompt for the language model involved a week of rigorous fine-tuning and numerous iterations. The process required balancing the amount of context provided in the prompt to ensure relevance without verbosity. Incorporating context prompts significantly improved the model's outputs, leading to more accurate and actionable insights.

The entire analysis process was automated to run every two weeks using AWS Lambda functions. This automation ensured that BB Agency continuously received up-to-date insights without manual intervention, allowing the team to focus on strategic decision-making and client interactions. The results of these analyses were stored in Amazon Aurora, from where they could be accessed and displayed in a readable format in Slack.

The implementation provided significant long-term value to BB Agency. By automating the identification of upsell opportunities and analyzing incomplete tasks, the agency could proactively address client needs and enhance service offerings. This approach led to increased upsell conversion rates and stronger client relationships. The scalability of Amazon Aurora and the flexibility of AWS Lambda ensured that the system could grow with the agency, accommodating an increasing volume of projects and data. The final design and output were rigorously evaluated and approved by three project managers who would be the primary users of these features. Their approval validated the practicality and usability of the implemented solutions. The system's ability to deliver consistent and accurate insights significantly enhanced the agency's project management capabilities and overall operational efficiency.

Keywords: Upsell; Advanced analytics; Machine learning; Client communication; Data analysis; Amazon Bedrock; Teamwork; Slack; Personalization; Revenue growth

Sažetak

Ovaj diplomski rad istražuje pristupe za prepoznavanja prilika za prodaju dodatnih usluga (upsell) kroz detaljnu analizu komunikacije između agencije i klijenata. U diplomskome radu se koriste i generativne AI tehnike za automatizirano prikupljanje i analizu podataka iz komunikacijskih alata kao što su Teamwork i Slack, kao i iz transkripata sastanaka. Fokus je na razumijevanju specifičnih potreba klijenata koje trenutno nisu zadovoljene.

Implementacija je započela integracijom Slacka kao primarne platforme za prikupljanje komunikacijskih podataka. Razvijen je prilagođeni bot za povezivanje s raznim Slack kanalima, sustavno izdvajanje poruka i interakcija korisnika. Ovaj je bot također pružao povratne informacije u stvarnom vremenu objavljivanjem rezultata analize natrag na kanale, osiguravajući da tim ostane informiran i angažiran. Prikupljeni podaci uključivali su detaljne zapise komunikacije i korisničke profile, a svi su pohranjeni u Amazon Aurora za strukturirano upravljanje i analizu. Usporedno s tim, Teamwork je korišten za njegove snažne mogućnosti upravljanja projektima. Dodana su prilagođena polja za povezivanje projekata u Teamworku s njihovim odgovarajućim Slack kanalima. Teamwork je pružio detaljne podatke o projektu, uključujući vremenske rokove, napredak zadatka i prekretnice. Ti su podaci također pohranjeni u Amazon Aurori, osiguravajući centralizirano i jedinstveno spremište za sve informacije povezane s projektom. Ključni aspekt implementacije bilo je korištenje Amazon Bedrocka, što je olakšalo implementaciju modela Anthropic Claude 3. Ovaj napredni jezični model korišten je za analizu prikupljenih podataka, koristeći Amazonov algoritam „Titan embedding“ za transformaciju podataka u smislene vektore. Mogućnosti modela bile su presudne u identificiranju obrazaca i izvlačenju uvida koji su ukazivali na potencijalne prilike za povećanje prodaje. Stvaranje učinkovitog odzivnika za jezični model uključivalo je tjedan dana rigoroznog finog podešavanja i brojnih ponavljanja. Proces je zahtijevao balansiranje količine konteksta danog u upitu kako bi se osigurala relevantnost bez opširnosti. Uključivanje konteksta značajno je poboljšalo rezultate modela, što je dovelo do točnijih i djelotvornijih uvida.

Cijeli proces analize je automatiziran za pokretanje svaka dva tjedna pomoću AWS Lambda funkcija. Ova automatizacija osigurala je da BB Agency kontinuirano prima ažurne uvide bez potrebe po ručnoj intervenciji, omogućujući timu da se usredotoči na donošenje strateških odluka i interakcije s klijentima. Rezultati ovih analiza pohranjeni su u Amazon Aurori, odakle im se moglo pristupiti i prikazati u čitljivom formatu u Slacku.

Implementacija je osigurala značajnu dugoročnu vrijednost BB agenciji. Automatiziranjem identifikacije prilika za povećanje prodaje i analizom nedovršenih zadataka, agencija može proaktivno odgovoriti na potrebe klijenata i poboljšati ponudu usluga. Ovaj pristup doveo je do povećanja stopa konverzije skuplje prodaje i čvršćih odnosa s klijentima. Skalabilnost Amazon Aurore i fleksibilnost AWS Lambde osigurale su da sustav može rasti s agencijom, prilagođavajući se sve većoj količini projekata i podataka. Konačni dizajn i rezultat su rigorozno ocijenjeni i odobreni od strane tri voditelja projekta koji će biti primarni korisnici ovih značajki. Njihovo odobrenje potvrdilo je praktičnost i upotrebljivost implementiranih rješenja. Sposobnost sustava da isporuči dosljedne i točne uvide značajno je poboljšala sposobnosti agencije za upravljanje projektima i ukupnu operativnu učinkovitost.

Ključne riječi: Prodaja dodatnih usluga; Napredna analitika; Strojno učenje; Komunikacija s klijentima; Analiza podataka; Amazon Bedrock; Teamwork; Slack; Personalizacija; Rast prihoda

Contents

1. Introduction

In today's competitive business environment, the ability to identify and capitalize on upsell opportunities is crucial for maximizing revenue and enhancing customer satisfaction. Upselling involves offering additional services or products to existing clients tailored to their specific needs and preferences. The motivation for this thesis is to explore a novel approach to identifying such opportunities through a detailed analysis of communication between an agency (i.e BB agency) and its clients. With the advancement of data analytics and machine learning, it is now possible to automate the collection and analysis of large datasets from various communication tools. This study focuses on technologies to analyze data from platforms like Teamwork and Slack, as well as potentially meeting transcripts, to uncover missed client needs and potential upsell moments.

The thesis's goal is to integrate generative artificial intelligence through Amazon Bedrock infrastructure. Amazon Bedrock provides a scalable and efficient foundation for processing and analyzing vast amounts of communication data using advanced AI capabilities. By utilizing machine learning models, the study aims to identify critical moments in client interactions that indicate upsell opportunities, thereby improving the precision and relevance of service recommendations.

Ultimately, the proposed solution not only enhances the ability to identify upsell opportunities based on existing communications but also enables the development of proactive strategies for real-time service offer adjustments. This leads to greater personalization, increased client satisfaction, and new pathways for revenue growth. This research's findings can transform how agencies approach client relationships and service offerings, paving the way for more dynamic and responsive business practices.

This thesis is organized into Chapters, and Chapter 2 is about the modern economy and data-driven business. Chapter 3 elaborates on large language models (LLMs), while Chapter 4 explains embedding vectors. Chapter 5 covers AWS, specifically Amazon Bedrock AWS lambda, S3, and Amazon Aurora. Chapter 6 is about the communication platform software Slack – used to obtain the data, while Chapter 7 explains project management software Teamwork, used to integrate the obtained results. Chapter 8 elaborates on the developed Upsell Discovery solution, which covers the actual implementation details. Each previously explained part is put into context and shown how it is being used in the BB Agency. Chapter 9 concludes and discusses the thesis.

2. Data-Driven Business in the Modern Economy

Data has become one of the most valuable assets for organizations across all industries in the modern business economy[1]. The concept of a data-driven business revolves around making strategic decisions based on data analysis and interpretation rather than intuition or mere observation. This approach has transformed traditional business models, enabling companies to operate more efficiently, understand their customers better, and gain a competitive edge. Data-driven businesses prioritize the collection of large volumes of data from various sources, such as customer interactions, social media, transaction records, and IoT devices. Effective data management systems are crucial for efficiently storing, processing, and retrieving this data. Leveraging advanced analytics, including machine learning and artificial intelligence, allows businesses to derive actionable insights from their data [[2]. Predictive analytics can forecast future trends, while prescriptive analytics can suggest the best course of action. Making decisions in real-time is a significant advantage of data-driven businesses. Real-time analytics enable companies to respond swiftly to market changes, customer behavior, and operational issues [3].

Understanding customer preferences and behavior is at the core of data-driven strategies [4]. By analyzing customer data, businesses can offer personalized experiences, products, and services, thereby increasing customer satisfaction and loyalty. Additionally, data-driven decision-making helps streamline operations, reduce costs, and improve overall efficiency [5]. For instance, predictive maintenance in manufacturing can prevent downtime, while supply chain analytics can optimize inventory levels and logistics. Some of the benefits can be seen on Figure 1.

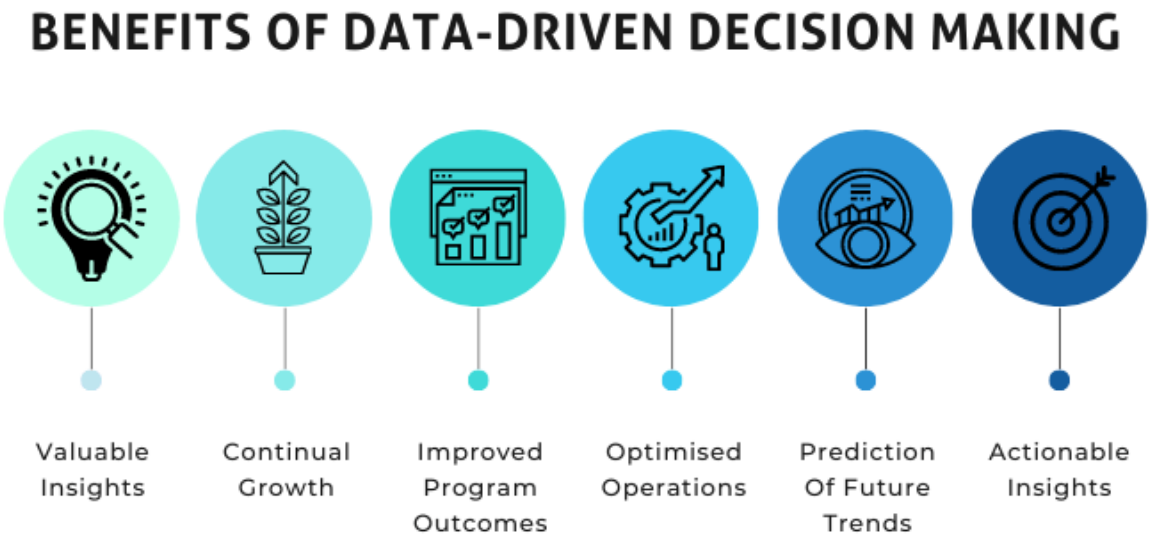

Figure 1 - Data-Driven Business in the Modern Economy.
Adopted from: https://www.analyticsvidhya.com/blog/2023/05/data-driven-decision-making/

The shift towards data-driven business models has profound implications for the business economy. Companies that effectively utilize data analytics can outperform their competitors by making informed decisions, anticipating market trends, and rapidly adapting to changes. Data-driven insights can drive innovation by identifying new market opportunities, customer segments, and product ideas, fostering business growth and expansion into new areas [6]. Automating data collection and analysis processes reduces the time and effort required for manual data handling, allowing employees to focus on more strategic tasks, thus boosting overall productivity and efficiency. Moreover, data analytics can enhance risk management by identifying potential risks and vulnerabilities in advance, enabling businesses to develop mitigation strategies based on predictive models and historical data trends.

Amazon Bedrock plays a crucial role in enabling data-driven business strategies [7]. By providing a robust infrastructure for data processing and AI-powered analytics, Amazon Bedrock allows businesses to handle large datasets and extract meaningful insights easily. The integration of AI capabilities facilitates the automation of data analysis, enhancing the accuracy and speed of decision-making processes.

In conclusion, the evolution towards data-driven business models represents a significant shift in the modern business economy. Companies that embrace data-driven strategies are better positioned to understand their customers, optimize operations, and drive innovation, ultimately achieving sustained growth and success in a competitive market.

1.1. BB Agency (Balkan Brothers)

BB Agency[8], a leading digital agency specializing in UI, UX, and product design for websites, as well as website development, has carved out a significant niche in the modern digital economy. By delivering exceptional design and development services, BB Agency helps businesses create engaging and user-friendly online experiences. The agency's success is heavily driven by its ability to understand client needs and offer tailored solutions that not only meet but exceed expectations. A substantial portion of BB Agency's revenue is generated through strategic upsells, where existing clients are offered additional services that enhance their initial projects. These upsells range from advanced functionality and ongoing maintenance to comprehensive digital marketing solutions, all designed to maximize the client's online presence and performance. This approach not only boosts BB Agency's revenue but also strengthens client relationships and satisfaction, making it a pivotal player in the digital services market. By continuously innovating and expanding its service offerings, BB Agency ensures sustained growth and a competitive edge in the dynamic digital business landscape. The company deck can be seen on Figure 2.

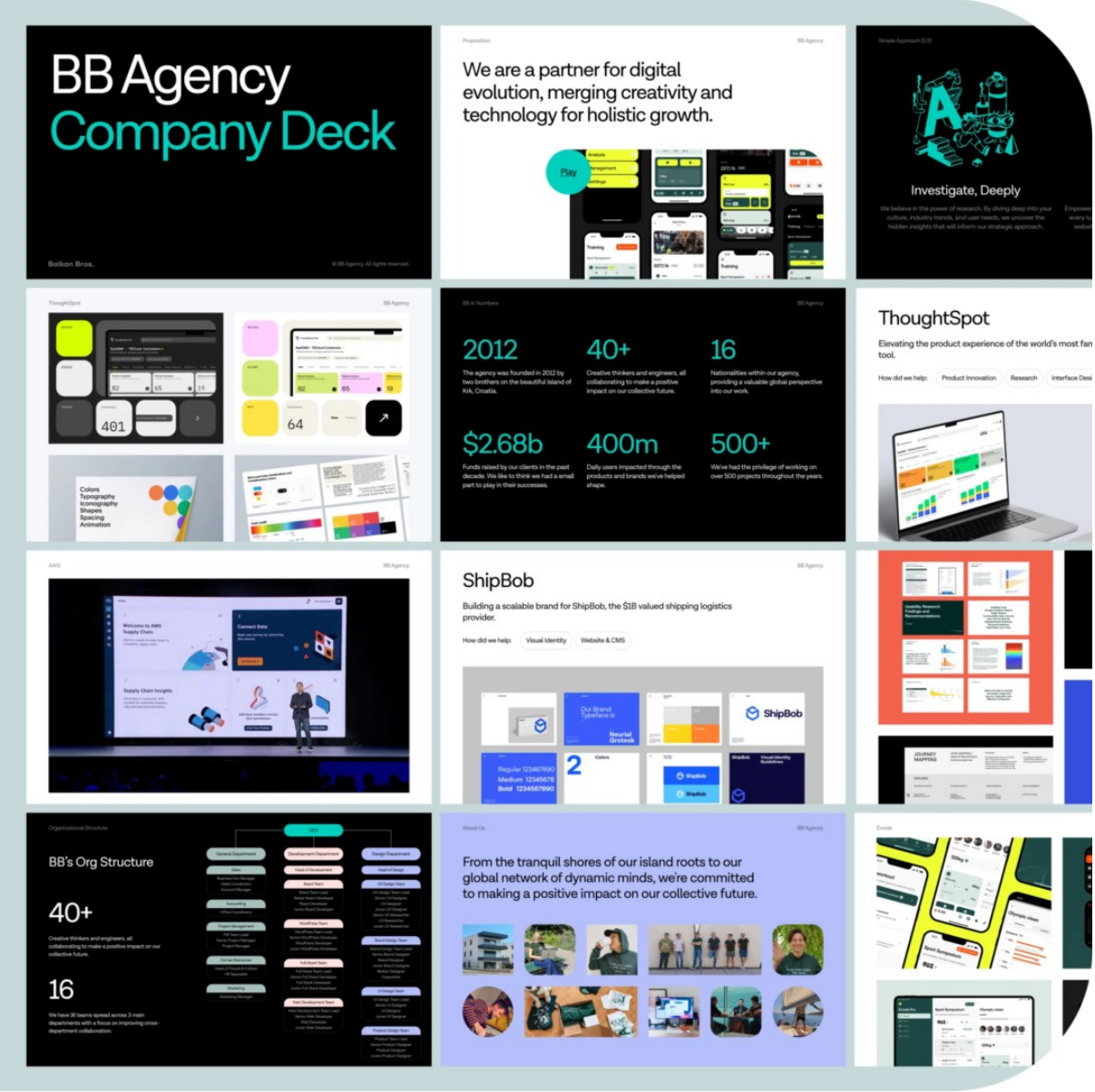

Figure 2 - BB Agency Company Deck.
Adopted from: https://bb.agency/contact/

1.2. Upsell

Upselling is a sales strategy where a business encourages customers to purchase more expensive items, upgrades, or additional services to enhance the value of their initial purchase[9]. The objective of upselling is to increase overall sales and profitability by convincing customers to opt for a higher-end product or add-on services that better meet their needs or provide extra benefits. For example, in the context of a digital agency like BB Agency, upselling involves recommending additional design features that are out of original scope, advanced functionality, ongoing maintenance plans, or comprehensive digital marketing packages to clients who initially sought essential website design services. By effectively identifying and presenting these opportunities, businesses can significantly boost their revenue while also delivering greater value and satisfaction to their customers. This strategy relies on

understanding customer needs and preferences and positioning the additional offerings as beneficial enhancements that complement the original purchase.
In this master thesis, we are focusing on finding upsell opportunities by analyzing communication data with generative AI (i.e. large language models LLMs) techniques to scrutinize communication data. In the thesis, we aim to uncover patterns and key communication moments that indicate potential upsell opportunities, thereby optimizing the upselling process and enhancing client relationships. Figure 3 is an illustration of cross-sell and upsell differences.

Figure 3 - Cross-sell and Upsell.
Adopted from: https://www.danielsavov.com/p/how-to-identify-cross-selling-opportunities

3. Large Language Models (LLMs)

Large language models (LLMs) are state-of-the-art artificial intelligence systems that can generate human-like text based on the input they receive [10], [11], [12]. These models are trained on vast amounts of data from the internet, books, and other sources [13], allowing them to understand and produce natural language with remarkable fluency. LLMs have shown impressive capabilities in various tasks such as question answering, text summarization, and creative writing, some of which are listed in Figure 4 [12]. One of the key strengths of LLMs is their ability to understand and generate context-aware responses. These models can produce relevant and coherent outputs by analyzing the input and the surrounding context. However, LLMs also have limitations, such as the potential for generating biased or factually incorrect information, as well as concerns around privacy and security. Ongoing research aims to address these challenges and further enhance the capabilities of LLMs.

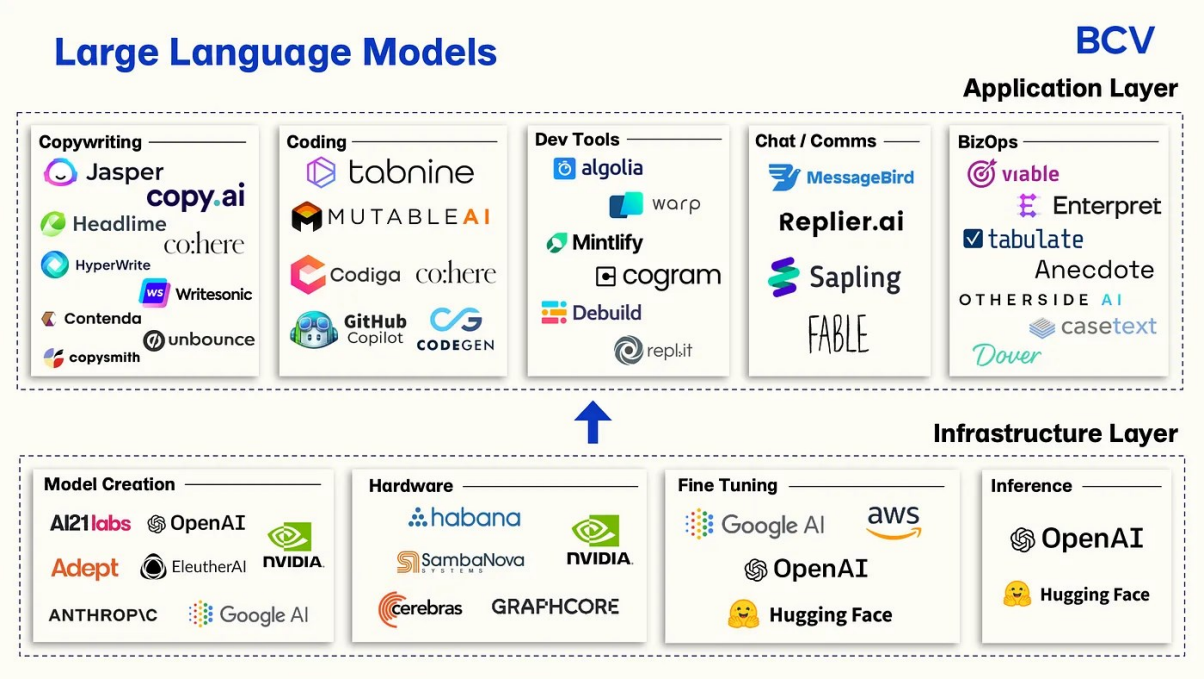

Figure 4 - Examples of LLMs.
Adopted from: https://medium.com/@dbhatt245/llms-101-a85f58b0a31c

LLMs have a wide range of applications across various industries:

- **Content Generation**[12]: LLMs can create articles, stories, and reports, making them useful tools for writers, journalists, and marketers. They can generate human-like text based on given prompts, saving time and effort in content creation.
- **Customer Support**[12]: Many businesses use LLMs to power chatbots and virtual assistants. These AI-driven agents can handle customer inquiries, provide information, and resolve issues, improving efficiency and customer satisfaction.
- **Translation Services**[12]: LLMs can accurately translate text from one language to another, facilitating communication across language barriers and enhancing global collaboration.
- **Sentiment Analysis**[12]: Companies use LLMs to analyze customer reviews, social media posts, and other textual data to gauge public sentiment about their products or services. This insight helps in making informed business decisions.

- **Personalized Recommendations**[12]: LLMs can analyze user behavior and preferences to provide personalized recommendations for products, services, and content, enhancing user experience and engagement.
- **Educational Tools**[12]: LLMs assist in education by providing explanations, answering questions, and even tutoring students in various subjects. They can create custom learning experiences based on individual student needs.

In the context of this master thesis, LLMs are enabling the analysis of communication data to identify upsell opportunities. By leveraging LLMs, businesses can automate the process of sifting through vast amounts of communication data from emails, chat logs, and meeting transcripts. The LLM can identify patterns, key phrases, and sentiments suggesting a client's interest in additional services. This allows businesses to tailor their sales strategies more effectively and personalize their offerings to meet specific client needs.
While LLMs offer significant benefits, they also pose challenges [14]. The computational resources required to train and run these models are substantial, often demanding powerful hardware and significant energy consumption. Additionally, LLMs can sometimes generate biased or inappropriate content based on the data they were trained on. Researchers are continually working on methods to make LLMs more efficient, ethical, and transparent.

3.1. Anthropic Claude

The Anthropic Claude model is a state-of-the-art large language model developed by Anthropic, an AI research organization[15], [16]. Named after Claude Shannon, the father of information theory, the Claude model emphasizes efficient and accurate information processing and generation. Built on the principles of transformer architecture, like other prominent language models such as OpenAI's GPT series [[17] and Google's BERT [18], the Claude model utilizes a deep neural network with a huge number of parameters, enabling it to handle complex language tasks. The model can understand and generate text across various domains and contexts by training on a diverse and extensive dataset, including books, articles, and websites. Its broad and impactful applications range from content creation, customer support, and translation to sentiment analysis and educational assistance. Businesses leverage the Claude model to enhance customer interactions through chatbots, translate and localize content, and analyze large volumes of textual data for trends and sentiment. In the context of this master thesis, the Claude model is an enabling component in analyzing communication data to identify upsell opportunities. By automatically sifting through communication records such as emails, chat logs, and meeting transcripts, the model detects patterns and sentiments that indicate a client's potential interest in additional services, facilitating more effective and personalized upselling strategies. Despite its advanced capabilities, the Claude model faces challenges such as the need for significant computational resources and the potential for biased content generation. Addressing these issues requires ongoing research to make the model more efficient, ethical, and transparent.

	Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4	GPT-3.5	Gemini 1.0 Ultra	Gemini 1.0 Pro
Undergraduate level knowledge *MMLU*	86.8% 5 shot	79.0% 5-shot	75.2% 5-shot	86.4% 5-shot	70.0% 5-shot	83.7% 5-shot	71.8% 5-shot
Graduate level reasoning *GPQA, Diamond*	50.4% 0-shot CoT	40.4% 0-shot CoT	33.3% 0-shot CoT	35.7% 0-shot CoT	28.1% 0-shot CoT	—	—
Grade school math *GSM8K*	95.0% 0-shot CoT	92.3% 0-shot CoT	88.9% 0-shot CoT	92.0% 5-shot CoT	57.1% 5-shot	94.4% Maj1@32	86.5% Maj1@32
Math problem-solving *MATH*	60.1% 0-shot CoT	43.1% 0-shot CoT	38.9% 0-shot CoT	52.9% 4-shot	34.1% 4-shot	53.2% 4-shot	32.6% 4-shot
Multilingual math *MGSM*	90.7% 0-shot	83.5% 0-shot	75.1% 0-shot	74.5% 8-shot	—	79.0% 8-shot	63.5% 8-shot
Code *HumanEval*	84.9% 0-shot	73.0% 0-shot	75.9% 0-shot	67.0% 0-shot	48.1% 0-shot	74.4% 0-shot	67.7% 0-shot
Reasoning over text *DROP, F1 score*	83.1 3-shot	78.9 3-shot	78.4 3-shot	80.9 3-shot	64.1 3-shot	82.4 Variable shots	74.1 Variable shots
Mixed evaluations *BIG-Bench-Hard*	86.8% 3-shot CoT	82.9% 3-shot CoT	73.7% 3-shot CoT	83.1% 3-shot CoT	66.6% 3-shot CoT	83.6% 3-shot CoT	75.0% 3-shot CoT
Knowledge Q&A *ARC-Challenge*	96.4% 25-shot	93.2% 25-shot	89.2% 25-shot	96.3% 25-shot	85.2% 25-shot	—	—
Common Knowledge *HellaSwag*	95.4% 10-shot	89.0% 10-shot	85.9% 10-shot	95.3% 10-shot	85.5% 10-shot	87.8% 10-shot	84.7% 10-shot

Figure 5 - Claude 3 performance by model.
Adopted from: https://www.anthropic.com/news/claude-3-family

Claude 3 is an advanced iteration of the Anthropic Claude model, and the results are visible in Figure 5. Claude 3 features more parameters and is trained on an expanded dataset, enhancing its ability to understand and generate human-like text [16]. It excels in content creation, customer support, translation, sentiment analysis, and educational assistance. In this master thesis, Claude 3 is used to analyze communication data to identify upsell opportunities. By processing emails, chat logs, and meeting transcripts, it detects patterns and sentiments indicating a client's interest in additional services, enabling personalized and effective upselling strategies. Despite its advancements, Claude 3 faces challenges such as the need for significant computational resources and addressing biases in content.

3.2. ChatGPT

ChatGPT, developed by OpenAI, is a prominent large language model designed to understand and generate human-like text[17], [19]. It is part of the Generative Pre-trained Transformer [12](GPT) series, known for its advanced natural language processing capabilities. ChatGPT uses deep learning techniques based on transformer architecture, enabling it to perform

complex language tasks with billions of parameters that aid in making predictions and generating coherent text. Trained on a vast and diverse dataset that includes books, articles, and web content, ChatGPT can understand and generate text across various domains and contexts. Its extensive applications cover content creation, customer support, translation, sentiment analysis, and personalized recommendations. Businesses utilize ChatGPT to power chatbots and virtual assistants, enhancing customer interactions with accurate and timely responses. It also aids in translating text between languages, providing insights into public sentiment by analyzing social media and reviews and offering personalized recommendations based on user behavior. Despite its advanced capabilities, ChatGPT faces challenges such as the need for substantial computational resources and the risk of generating biased or inappropriate content based on its training data. Ongoing research aims to make the model more efficient, ethical, and transparent. ChatGPT is expected to become even more sophisticated as AI technology advances, further enhancing its applications and value to businesses and society.

Figure 6 - Introduction of ChatGPT-4o.
Adopted from: https://www.linkedin.com/pulse/chatgpt-4o-ai-talks-like-real-person-yuriy-danylov-slamf/

ChatGPT-4o, the logo shown in Figure 6, is a cutting-edge language model developed by OpenAI, representing a significant leap forward in the Generative Pre-trained Transformer (GPT) series[20]. Building on the success of its predecessors, ChatGPT-4o features an even more sophisticated transformer architecture with billions of parameters, enabling it to understand and generate human-like text with remarkable accuracy and coherence. Trained on an extensive and diverse dataset, ChatGPT-4o excels in a variety of language tasks, from content creation and customer support to translation and sentiment analysis. One of ChatGPT-4o's key strengths is its ability to comprehend nuanced prompts and provide detailed, contextually relevant responses. This makes it a valuable tool for businesses looking to enhance customer interactions, automate content production, and gain insights from textual data. Its wide-ranging applications include powering chatbots and virtual assistants, translating text between languages, analyzing public sentiment from social media and reviews, and offering personalized recommendations based on user behavior. Despite its advanced capabilities, ChatGPT-4o also faces challenges, such as requiring significant computational resources and addressing biases in the generated content. Ongoing efforts in research and development aim

to make ChatGPT-4o more efficient, ethical, and transparent, ensuring it remains a versatile and reliable tool in various fields.

3.3. Llama (Large Language Model Meta AI)

Meta's Llama (Large Language Model Meta AI) is an advanced language model developed by Meta (formerly Facebook), designed to push the boundaries of natural language understanding and generation[21], [22]. Llama utilizes a transformer-based architecture known for its efficiency and scalability in handling complex language tasks. With a substantial number of parameters, Llama can process and generate text with high accuracy and contextual relevance. Llama has been trained on an extensive and diverse dataset encompassing a wide range of text sources such as books, articles, and web content. This diverse training data enables Llama to understand and produce coherent and contextually appropriate text across various domains and applications. The model is particularly adept at content creation, customer support, translation, sentiment analysis, and more.
One of the standout features of Llama is its open-source nature, which allows developers and researchers worldwide to access, modify, and enhance the model [23]. This openness fosters innovation and collaboration, enabling a broad community to contribute to its development and application. Businesses looking to enhance their digital interactions and content strategies can leverage Llama's capabilities more flexibly and cost-effectively, integrating them into their existing systems and workflows.

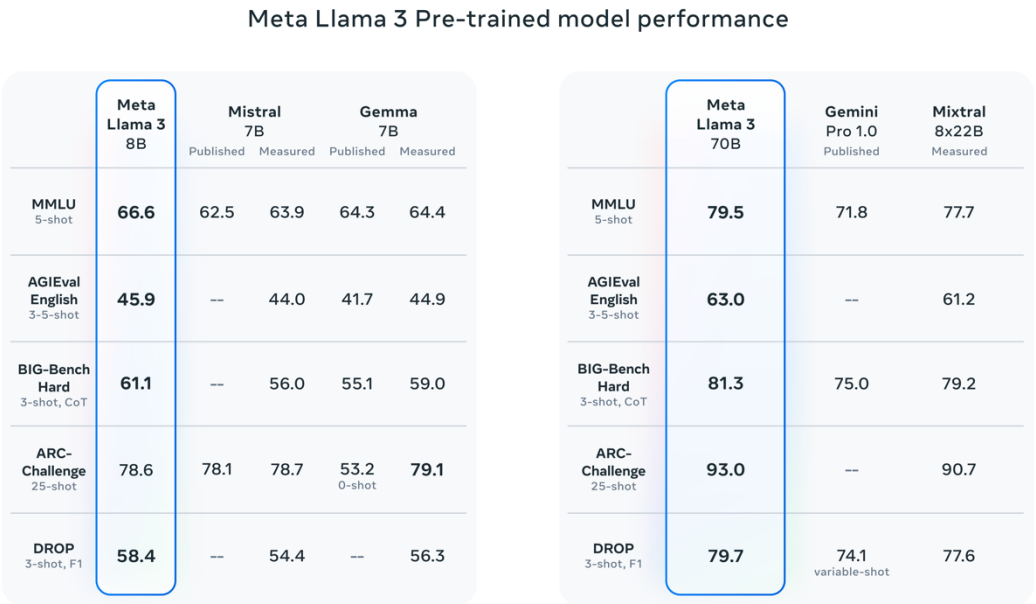

Figure 7 - Llama 3 performance.
Adopted from: https://ollama.com/library/llama3

The latest iteration in the Llama family, Llama 3, brings even more advancements and improvements, and performance can be seen in Figure 7. Llama 3 features an increased number of parameters and enhancements in its training process, resulting in even greater accuracy and contextual understanding. This new version builds on the strengths of its predecessors while addressing previous limitations, offering improved performance and efficiency. Llama 3's enhanced capabilities make it an even more powerful tool for generating high-quality text,

understanding complex queries, and providing more precise insights. Importantly, Llama 3 continues the tradition of being open source, allowing a wide array of applications and further development by the global community.

Despite its advanced capabilities, Llama also faces challenges, such as the need for significant computational resources and addressing potential biases in the generated content. To mitigate these issues, ongoing research and development focus on improving the model's efficiency, ethical standards, and transparency.

4. Embeddings

Embeddings are a fundamental concept in natural language processing (NLP) and machine learning, providing a way to represent words, phrases, or entire documents as vectors of real numbers in a continuous vector space[12], [24]. Embeddings facilitate the processing and analysis of textual data by machine learning algorithms, which operate more effectively on numerical data than on raw text. At the core of embeddings is the idea of capturing the semantic meaning of words and their relationships. Traditional approaches to representing text, such as one-hot encoding, fail to capture these relationships, resulting in sparse, high-dimensional vectors [25]. Embeddings, on the other hand, produce dense, low-dimensional representations that place similar words closer together in the vector space [12]. This allows for a more nuanced understanding of language, where words with similar meanings or contexts have similar vector representations. For easier visualization, see Figure 8.

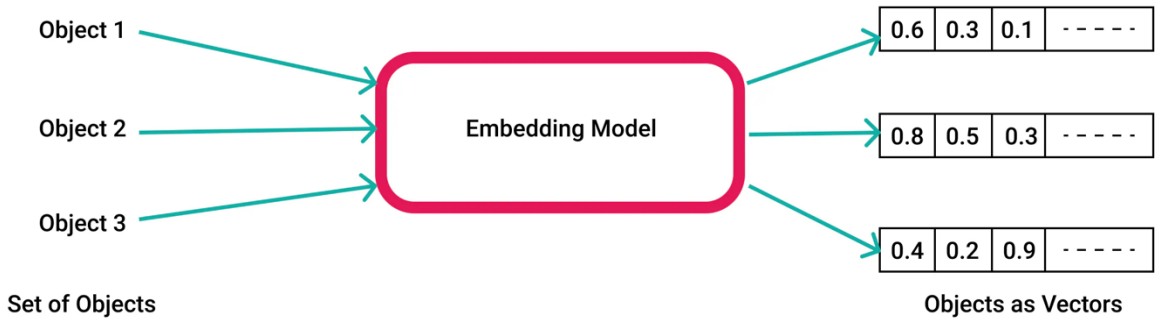

Figure 8 - Visualization of where the embedding model fits in.
Adopted from: https://www.pinecone.io/learn/vector-embeddings/

Large language models (LLMs) learn embedding vectors that represent words in a low-dimensional continuous vector space. The embedding vectors capture semantic similarities between words, whereas words with similar meanings have vectors that are close in the embedding space. This allows LLMs to deal with data sparsity and compute semantic similarities between inputs regardless of their form or modality. The embedding vectors learned by LLMs are defined in a hidden space where the semantic similarity between vectors can be readily computed as their distance. This enables applications like computing semantic similarity between queries and documents in web searches, sentences in different languages for machine translation, and between images and text for image captioning.

Several embedding techniques have been developed, each with its strengths and applications:

- **Word2Vec**[26]: Introduced by Mikolov, Word2Vec uses shallow neural networks to learn word associations from large text corpora. It creates two types of embeddings: Continuous Bag of Words (CBOW) and Skip-gram. CBOW predicts a word based on its context, while Skip-gram predicts the context given a word. Word2Vec is efficient and effective in capturing semantic similarities between words.
- **GloVe (Global Vectors for Word Representation)**[27]: Developed by Stanford, GloVe combines the advantages of global matrix factorization and local context window methods. It constructs a word co-occurrence matrix and uses factorization to derive word vectors. GloVe embeddings capture global statistical information, making them useful for various NLP tasks.

- **BERT (Bidirectional Encoder Representations from Transformers)**[28]: Developed by Google, BERT uses transformer-based neural networks to generate context-aware embeddings. Unlike Word2Vec and GloVe, BERT considers a word's context in both directions (left and right), providing a more accurate representation of polysemous words (words with multiple meanings) and different contexts. BERT embeddings are highly effective for tasks requiring deep language understanding, such as question answering and language inference.

Recognizing the power of embeddings, many leading technology companies have developed their own custom embedding models tailored to their specific needs and applications. For example:

- **Google**: In addition to developing BERT, Google continues to advance embedding techniques through models like ALBERT and T5, integrating these embeddings into its search engine, translation services, and various AI applications.
- **Meta (Facebook)**: Meta's AI Research (FAIR) has developed FastText, an efficient word embedding model that extends the capabilities of traditional embeddings by considering subword information. These embeddings are used across Facebook's platforms to improve content recommendations, language translation, and user interaction.
- **Microsoft**: Microsoft utilizes embeddings within its Azure AI services, incorporating models like ELMo and custom embeddings for applications in Office 365, Bing search, and its virtual assistant, Cortana.
- **Amazon**: Amazon employs embeddings in its Alexa virtual assistant and various AWS services. Embeddings are crucial for understanding and generating natural language in customer interactions and improving search relevance on platforms like Amazon.com.

Embeddings have revolutionized NLP by enabling a wide range of applications:

- **Text Classification**[12]: Embeddings improve the performance of text classification models by providing rich semantic representations of words and documents. This is crucial for spam detection, sentiment analysis, and topic categorization applications.
- **Machine Translation**[12]: Embeddings enhance machine translation systems by providing better word and phrase representations, leading to more accurate translations.
- **Information Retrieval**[12]: In search engines and recommendation systems, embeddings help match queries with relevant documents or items based on semantic similarity.
- **Named Entity Recognition (NER)** [12]: Embeddings improve the identification and classification of entities within the text, such as names, dates, and locations, by providing contextual understanding.
- **Question Answering Systems**[12]: Embeddings enable more effective question answering by understanding the context and meaning of both questions and answers.

Despite their successes, embeddings face challenges such as capturing complex linguistic phenomena like sarcasm and idioms [29]. They also require substantial computational resources for training, especially context-aware embeddings like BERT. Ongoing research aims to address these issues by developing more efficient algorithms and exploring new architectures, such as those combining embeddings with other AI techniques. In conclusion, embeddings have become a cornerstone of modern NLP, transforming how machines

understand and process human language. By capturing the semantic relationships between words, embeddings enable more accurate and efficient language models, driving advancements in various applications from text analysis to machine translation. As research progresses and major companies continue to innovate, embeddings will continue to evolve, offering even more powerful tools for understanding and generating natural language.

5. AWS

Amazon Web Services (AWS) is a comprehensive and widely adopted cloud computing platform provided by Amazon[30]. Launched in 2006, AWS offers a broad set of global cloud-based products, including computing power, storage options, networking capabilities, advanced tools for machine learning, artificial intelligence, and analytics. AWS is designed to be scalable, flexible, and cost-effective, making it an ideal choice for businesses of all sizes, from startups to large enterprises. The platform supports diverse workloads, including web and mobile applications, data processing, warehousing, storage, and archiving. AWS's extensive infrastructure spans multiple geographic regions worldwide, ensuring high availability and reliability. Additionally, AWS provides robust security measures and compliance certifications, enabling organizations to meet stringent regulatory requirements. By leveraging AWS, businesses can innovate faster, reduce IT costs, and easily scale applications globally. Some logos of the services are shown in Figure 9.

Figure 9 - AWS services.
Adopted from: https://allcode.com/top-aws-services/

5.1. Amazon Bedrock

Amazon Bedrock is a managed service within Amazon Web Services (AWS) that simplifies the deployment, management, and scaling of machine learning models and applications[31]. Designed to cater to developers and data scientists, Bedrock provides a comprehensive foundation that integrates seamlessly with popular machine learning frameworks and tools such as TensorFlow, PyTorch, and MXNet[32], [33], [34]. This service significantly reduces the complexity and time required to transition machine learning projects from conception to production by offering features like automated data preprocessing, model training, and hyperparameter tuning. Amazon Bedrock also includes capabilities for real-time inference, batch processing, and integration with other AWS services, such as Amazon SageMaker, for further customization and optimization of machine learning workflows[35].

With Bedrock, businesses can leverage pre-built, domain-specific models for tasks such as natural language processing, computer vision, and predictive analytics, enabling them to derive actionable insights from their data quickly [36]. Additionally, Bedrock supports robust monitoring and logging features, ensuring that models perform reliably and meet predefined performance metrics. The platform's scalability allows organizations to efficiently handle varying workloads and scale their applications globally while maintaining stringent security and compliance standards. By utilizing Amazon Bedrock, businesses can accelerate their machine learning initiatives, reduce operational overhead, and innovate more effectively in a competitive landscape.

In this master thesis, Amazon Bedrock is utilized to facilitate the analysis of communication data to identify upsell opportunities. The integration with Bedrock allows for the efficient handling and processing of large datasets, leveraging advanced machine learning models to detect patterns and insights that suggest potential upsell moments. This application of Amazon Bedrock demonstrates its versatility and powerful capabilities in transforming raw data into strategic business insights.

5.2. AWS Lambda

AWS Lambda is a serverless computing service provided by Amazon Web Services (AWS) that allows developers to run code without provisioning or managing servers[37]. Lambda automatically scales applications by running code in response to events such as changes to data in an Amazon S3 bucket or updates to a DynamoDB table. This event-driven model simplifies the development process, enabling developers to focus on writing code and business logic rather than on infrastructure management. AWS Lambda supports various programming languages, including Python, Node.js, Java, and Go, making it flexible and accessible to a wide range of developers. By charging only for the compute time consumed—meaning developers pay only for the milliseconds their code runs—AWS Lambda offers a cost-effective solution for building scalable, resilient applications. Its seamless integration with other AWS services and its ability to handle real-time processing tasks make AWS Lambda an essential tool for modern cloud-native applications.

5.3. Amazon Simple Storage Service (Amazon S3)

Amazon Simple Storage Service (Amazon S3) is a scalable, high-speed, web-based cloud storage service designed for online backup and archiving of data and applications on Amazon Web Services (AWS) [38]. Launched in 2006, S3 provides developers and IT teams with secure, durable, and highly scalable object storage. S3's design aims to deliver 99. 99% of durability, ensuring that data is reliably preserved. Users can store and retrieve any amount of data at any time, from anywhere on the web, making it an essential service for modern cloud-based applications. Amazon S3 supports a wide range of use cases, including data lakes, mobile applications, backup and restore, archive, enterprise applications, and big data analytics. The service offers robust security features, such as encryption and access management tools, to protect data and ensure compliance with regulatory requirements. Additionally, S3 seamlessly integrates with a variety of AWS services, enabling users to build comprehensive, scalable solutions. Its pay-as-you-go pricing model ensures cost efficiency, making Amazon S3 a versatile and economical choice for businesses of all sizes.

5.4. Amazon Aurora

Amazon Aurora is a fully managed relational database engine provided by Amazon Web Services (AWS) that is compatible with MySQL and PostgreSQL[39], [40], [41]. Designed for high performance and availability, Aurora combines the speed and reliability of high-end

commercial databases with the simplicity and cost-effectiveness of open-source databases. It offers up to five times better performance than standard MySQL databases and three times better performance than standard PostgreSQL databases, all at a fraction of the cost of traditional commercial database solutions. Amazon Aurora automatically replicates data across multiple Availability Zones, ensuring high availability and durability with automatic failover and backup capabilities. The service also features seamless scaling, allowing storage to automatically grow as needed, up to 128 terabytes per database instance. Additionally, Aurora integrates with other AWS services, such as AWS Lambda, Amazon S3, and Amazon Redshift, to enable sophisticated data processing and analytics workflows. Its pay-as-you-go pricing model and managed service offerings make Amazon Aurora an attractive choice for enterprises seeking robust, scalable, and cost-effective database solutions.

6. Slack

Slack is a cloud-based collaboration platform designed to facilitate communication and teamwork within organizations[42]. Launched in 2013 by Stewart Butterfield and his team, Slack has rapidly become a cornerstone tool for modern workplaces, particularly those with remote or distributed teams. Its user-friendly interface and robust feature set make it a versatile tool for enhancing productivity and collaboration. Representation of Slack's UI is shown in Figure 10.

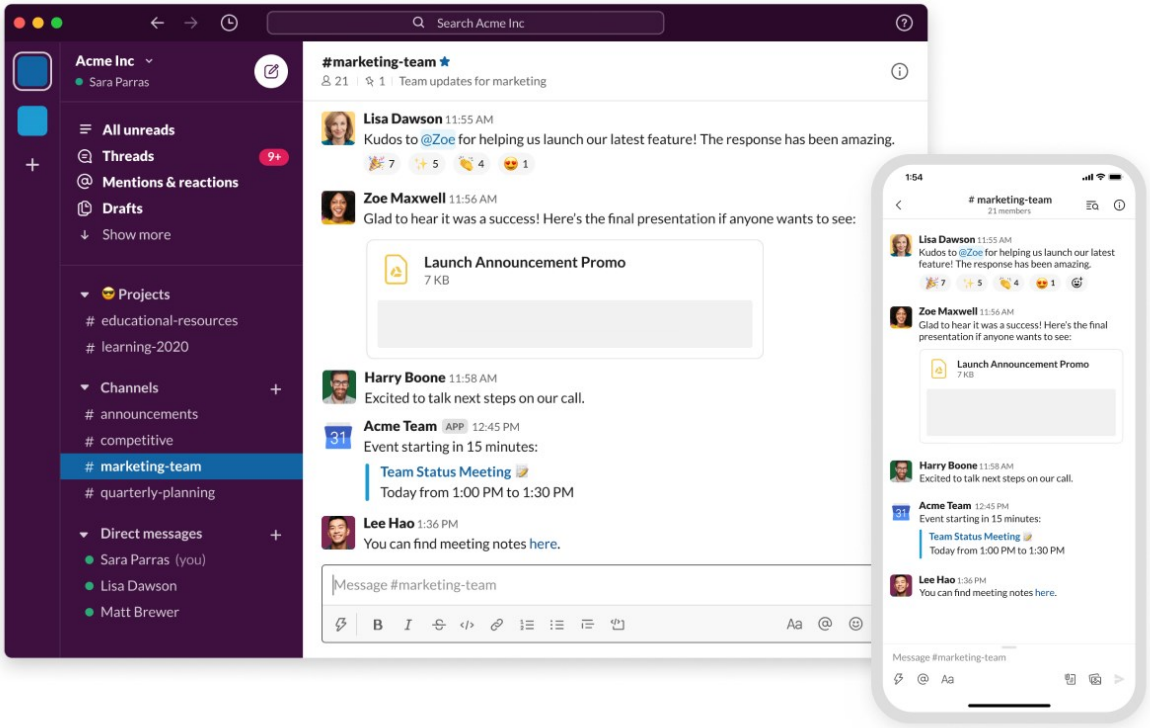

Figure 10 - Representation of Slack.
Adopted from: https://slack.com/features

Slack offers a wide range of features that streamline communication and improve team coordination. The platform is organized into channels, which can be dedicated to specific projects, teams, or topics, allowing for organized and focused conversations. Channels can be public for team-wide discussions or private for more confidential topics. In addition to channels, Slack provides direct messaging for one-on-one interactions and group chats for small team discussions. Slack's functionality extends beyond simple messaging. It supports file sharing, voice and video calls, and offers a powerful search function allowing users to find past messages, files, and discussions quickly. Integrations with numerous third-party applications, such as Google Drive, Trello, and GitHub, enhance Slack's capabilities by bringing various tools into one unified platform. This centralization of tools and conversations helps reduce the clutter of emails and meetings, leading to more efficient workflows and better communication.

One of Slack's key strengths is its ability to integrate with other software and services, making it highly customizable. Users can add apps from the Slack App Directory to extend the functionality of their workspace, ranging from productivity tools to fun and engaging apps that improve team morale. Custom integrations allow teams to tailor Slack to fit their unique workflows and requirements, further enhancing its utility and adoption.

6.1 Slack API

The Slack API (Application Programming Interface) allows developers to extend Slack's functionality even further by creating custom integrations, bots, and automation workflows [43]. The API allows access to Slack's core features, including messaging capabilities, channel management, user information, and more. Developers can leverage the API to build interactive applications that can respond to user inputs, automate repetitive tasks, and integrate Slack with other systems and services.

A key application of the Slack API in this thesis is in analyzing communication within an organization. By accessing message histories, channel activities, and user interactions, developers can create tools that analyze communication patterns and extract valuable insights. This analysis can help identify trends, measure engagement, and detect opportunities for upselling or improving collaboration. For instance, sentiment analysis tools can gauge the mood of the team, while keyword tracking can highlight recurring issues or topics of interest.

The Slack API supports various programming languages and offers comprehensive documentation, making it accessible to a wide range of developers. Key features of the Slack API include:

- **Messaging and Interactivity**: Developers can create bots and apps that send and receive messages, post interactive components like buttons and forms, and respond to events within Slack.
- **Workspace Management**: The API allows for the management of channels, users, and user groups, enabling automated workflows for workspace administration.
- **Data Synchronization**: Integrations can synchronize data between Slack and other tools, ensuring that information is always up-to-date and accessible from within Slack.
- **Event Subscriptions**: The API can subscribe to specific events, such as message posts or user activities, allowing apps to react in real time.

Organizations can gain deeper insights into how their teams interact and collaborate by utilizing the Slack API for communication analysis. This capability makes Slack not just a communication tool but also a vital source of data for improving organizational efficiency and effectiveness.

Slack and its API provide a powerful combination for enhancing team communication and collaboration. With its rich feature set, extensive integrations, and customizable API, Slack helps organizations streamline their workflows, reduce communication barriers, and improve overall efficiency. As businesses continue to adapt to remote and hybrid work environments, tools like Slack will remain essential for fostering effective teamwork and maintaining productivity. The ability to analyze communication through Slack’s API further empowers organizations to optimize their operations and better understand their internal dynamics

7. Teamwork

Teamwork is a cloud-based project management and collaboration platform designed to help teams improve productivity and streamline their workflows[44]. Teamwork provides a comprehensive suite of tools for project planning, task management, time tracking, and collaboration. Its intuitive interface and robust feature set make it an essential tool for businesses of all sizes, particularly those managing complex projects or working with distributed teams. Teamwork's UI is shown in Figure 11.

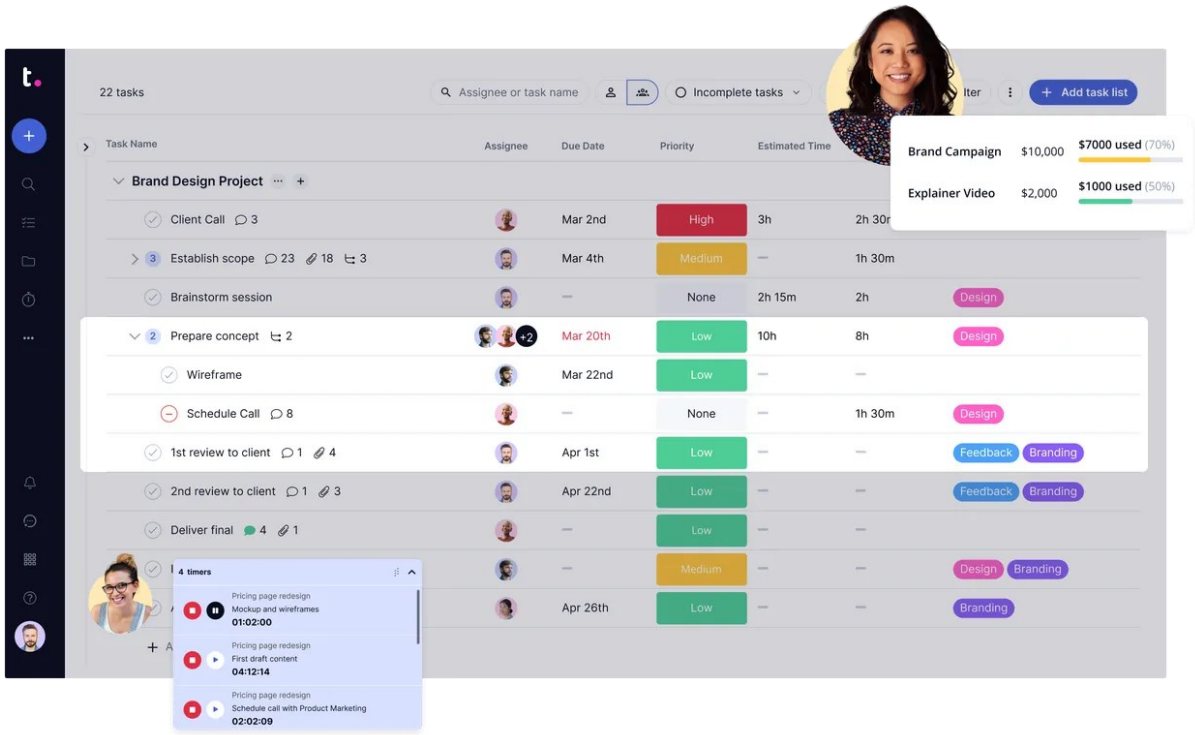

Figure 11 - Teamwork dashboard.
Adopted from: https://www.teamwork.com/product/

Teamwork offers a wide range of features that enhance project management and team coordination. The platform allows users to create detailed project plans with milestones, tasks, and subtasks, ensuring that all team members are aligned and aware of their responsibilities. Tasks can be assigned to individuals or groups, with due dates and priority levels, facilitating clear and organized project execution. Teamwork's functionality extends beyond basic task management. It includes features for time tracking, enabling team members to log hours worked on specific tasks and projects. This is particularly useful for billing clients or analyzing productivity. The platform also supports file sharing and document collaboration, allowing teams to store and edit project-related files in a centralized location. Communication within Teamwork is streamlined through its messaging and commenting system, where team members can discuss tasks, share updates, and provide feedback directly within the platform. This reduces the need for external communication tools and keeps all project-related information in one place.

Teamwork is highly customizable, allowing users to tailor the platform to fit their specific workflows and preferences. Custom fields, templates, and tags enable teams to organize their projects in a way that best suits their needs. Additionally, Teamwork integrates with numerous

third-party applications, such as Google Drive, Slack, and Microsoft Office, enhancing its capabilities and providing a seamless workflow across different tools.

7.1 Teamwork API

The Teamwork API (Application Programming Interface) provides developers with the ability to extend the platform's functionality and integrate it with other systems[45]. The API allows access to Teamwork's core features, including project management, task assignments, time tracking, and more. Developers can leverage the API to build custom integrations, automate workflows, and create tailored solutions that meet their unique business needs.

A key application of the Teamwork API is in analyzing project and communication data within an organization. By accessing project details, task updates, and user interactions, developers can create tools that analyze project progress, team performance, and collaboration patterns. This analysis can help identify bottlenecks, measure productivity, and detect opportunities for improving project management practices.

The Teamwork API supports various programming languages and offers comprehensive documentation, making it accessible to a wide range of developers. Key features of the Teamwork API include:

- **Project Management**: Developers can create, update, and manage projects, tasks, and milestones, automating various aspects of project planning and execution.
- **Time Tracking**: The API allows for the integration of time tracking data, enabling automated logging and reporting of hours worked on tasks and projects.
- **User and Team Management**: The API provides tools for managing users, teams, and permissions, facilitating the administration of complex projects and large teams.
- **Data Synchronization**: Integrations can synchronize data between Teamwork and other tools, ensuring that project information is always up-to-date and accessible from within Teamwork.
- **Event Subscriptions**: The API can subscribe to specific events, such as task updates or project changes, allowing apps to react in real time.

By utilizing the Teamwork API for project and communication analysis, organizations can gain deeper insights into how their teams are performing and collaborating. This capability makes Teamwork a project management tool and a vital source of data for improving organizational efficiency and effectiveness.

Teamwork and its API provide a powerful combination for enhancing project management and team collaboration. With its rich feature set, extensive integrations, and customizable API, Teamwork helps organizations streamline their workflows, reduce communication barriers, and improve overall efficiency. As businesses continue to manage complex projects and adapt to changing work environments, tools like Teamwork will remain essential for fostering effective teamwork and maintaining productivity. The ability to analyze project and communication data through Teamwork’s API further empowers organizations to optimize their operations and better understand their internal dynamics.

8. Upsell Discovery

In this chapter, it will be explained how the whole upsell discovery was created and implemented in the real use case for the company BB Agency. At BB Agency, identifying upsell opportunities is critical for maximizing revenue and enhancing client satisfaction. The projects undertaken by BB Agency are generally categorized into two types: one-time projects and retainers. One-time projects are specific, finite engagements with a clear start and end, such as website redesigns or mobile app developments. Retainers, on the other hand, involve ongoing support and development services, providing a steady stream of income and continuous client engagement.

Initially, BB Agency focused on identifying upsell opportunities by analyzing communication data within its primary collaboration platform, Slack. Given that Slack channels are the main hub for the team's interactions, they provide a rich source of data for understanding client needs and project dynamics. By leveraging the Slack API to access and analyze message histories, channel activities, and user interactions, BB Agency aimed to detect potential upsell signals—such as clients expressing additional needs, requesting extra features, or showing interest in services beyond the current scope.

This approach involved using LLM enabled natural language processing (NLP) techniques to perform analysis within Slack messages. This allowed for the identification of interest around additional services, which were then flagged as potential upsell opportunities. These insights will help tailor pitches and proposals more effectively, aligning offerings with the client's evolving needs.

8.1. Types of Projects and Searching for Incomplete Tasks

The distinction between one-time projects and retainers significantly influenced the upsell strategy at BB Agency. In one-time projects, upsell opportunities typically emerge during the project's lifecycle, often when clients request enhancements or additional features not included in the original project scope. For retainers, the ongoing nature of the relationship provided more frequent opportunities for upsells, such as introducing new services, upgrades, or advanced support packages. An infographic explaining both can be seen in Figure 12.

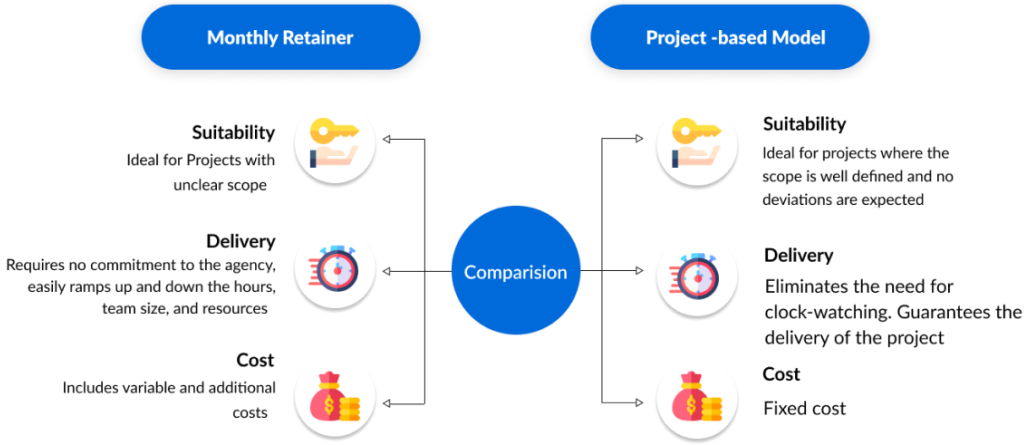

Figure 12 - Retainer vs One-time Project.
Adopted from: https://www.doodleblue.com/blogs/6-reasons-why-retainer-agreements-work/

As the thesis progressed, it became apparent that focusing solely on explicit upsell signals was not yielding the desired results for BB agency. Further analysis revealed a more effective and general approach: targeting incomplete tasks as indicators of potential upsell opportunities. Incomplete tasks often represented unmet client needs or features that had been deprioritized due to time or budget constraints. Hence, the detection of incomplete tasks includes the detection of upselling opportunities. By addressing these incomplete tasks, BB Agency could offer tangible value and directly address clients' pending requirements.

BB Agency adapted its strategy by utilizing both Slack and the project management tool, Teamwork. Using the Slack API and Teamwork API, a system was created to track and analyze incomplete tasks across all active projects. This involved:

- **Integrating Data**: Synchronizing data between Slack and Teamwork ensured comprehensive visibility of project statuses and task completion rates. This integration allowed for the correlation of communication data with task progress, identifying gaps and pending work.
- **Automated Alerts**: Automated alerts were set up to notify project managers and sales teams whenever a task remained incomplete for an extended period or was marked as low priority despite client discussions suggesting its importance.
- **Context Analysis**: By combining analysis from Slack messages with task data from Teamwork, BB Agency gained insights into the context around incomplete tasks. This helped in tracking the progress of tasks through the time.
- **Client Communication**: BB Agency proactively reached out to clients with customized proposals to address these incomplete tasks. By highlighting the benefits of completing these tasks—such as enhanced functionality, improved user experience, or better performance—these tasks were positioned as valuable upsell opportunities that aligned with the client's goals.

This refined approach proved to be highly effective for BB Agency. By targeting incomplete tasks, immediate client needs were addressed, demonstrating a commitment to their success. This not only increased upsell conversion rates but also strengthened client relationships and satisfaction. Clients appreciated the proactive stance and the value-added services offered, leading to higher retention rates and more positive referrals.

Implementing a data-driven approach to identify upsell opportunities through Slack and Teamwork has significantly enhanced BB Agency's business strategy. By transitioning from searching for explicit upsell signals to identifying incomplete tasks as potential upsells, services were more closely aligned with client needs, resulting in increased revenue and client satisfaction. This approach demonstrates the power of leveraging advanced analytics and API integrations to drive business growth and operational efficiency at BB Agency.

8.2. Amazon Bedrock

In the quest to enhance the identification of upsell opportunities and streamline communication analysis, BB Agency integrated the advanced capabilities of a large language model (LLM), specifically the Anthropic Claude 3 Sonnet, into its workflow. This integration was facilitated through Amazon Bedrock, a robust and scalable platform designed to support the deployment and management of machine learning models.

Amazon Bedrock played a crucial role in deploying and managing Anthropic Claude 3 within BB Agency's operations. Bedrock is designed to simplify the deployment of machine learning models by providing a managed service that handles the underlying infrastructure, scalability, and integration with other AWS services. This allowed BB Agency to focus on leveraging the advanced capabilities of Claude 3 without the overhead of managing complex ML infrastructure. By utilizing Anthropic Claude 3 through Amazon Bedrock, BB Agency was able to enhance its data-driven approach to identifying upsell opportunities and analyzing communication data. The key benefits included:

- **Scalability**: Amazon Bedrock provided the necessary scalability to handle large volumes of communication data from Slack and Teamwork. This ensured that the analysis could be performed efficiently, even as the amount of data increased.
- **Integration**: Bedrock's seamless integration with other AWS services enabled BB Agency to create a cohesive data pipeline. Data from Slack and Teamwork was ingested, processed, and analyzed using Claude 3
- **Performance**: The advanced infrastructure provided by Amazon Bedrock ensured high performance and reliability. This allowed Claude 3 to process complex language tasks quickly and accurately, providing timely insights that were critical for identifying upsell opportunities.
- **Ease of Use**: Amazon Bedrock's managed service model reduced the complexity associated with deploying and maintaining machine learning models. This enabled BB Agency to focus on extracting actionable insights rather than dealing with the intricacies of ML infrastructure management.

The integration of Claude 3 through Amazon Bedrock allowed BB Agency to enhance its method of identifying upsell opportunities. By analyzing communication data from Slack channels and Teamwork projects, Claude 3 was able to detect subtle cues and patterns indicating client needs and potential upsell moments. The model analyzed and understood the context of discussions to highlight opportunities that might otherwise have been overlooked. Invoking of Claude 3 Sonnet can be seen in Figure 13.

```
response = client.messages.create(
    max_tokens=10000,
    system=system_context,
    temperature=0.1,
    messages=[
        {
            "role": "user",
            "content": [{"type": "text", "text": prompt}],
        },
        {
            "role": "assistant",
            "content": "Here is the JSON requested:"
        }
    ],
    model="anthropic.claude-3-sonnet-20240229-v1:0",
)

input_tokens = response.usage.input_tokens
output_tokens = response.usage.output_tokens
output_text = response.content[0].text
```

Figure 13 - Invoking Claude 3 Sonnet from Amazon Bedrock.

As the strategy evolved, BB Agency transitioned to focusing on incomplete tasks as potential upsell opportunities. Claude 3, deployed via Amazon Bedrock, analyzed task data from Teamwork in conjunction with communication data from Slack. This approach provided a

comprehensive understanding of why certain tasks remained incomplete, whether due to budget constraints, shifting priorities, missed oportunities or other factors. By addressing these tasks proactively, BB Agency was able to offer tailored solutions that directly met client needs, thereby increasing the likelihood of successful upsells.

8.3. Storing Data

In the initial phase, BB Agency leveraged Amazon Bedrock's knowledge base to store and manage data. This approach functioned as a Retrieval-Augmented Generation (RAG) system, enhancing the model's ability to generate relevant responses by utilizing a knowledge base of unstructured data. Data was first loaded into an Amazon S3 bucket, which served as a scalable and reliable storage solution. We can choose between a few embedding models that are shown in Figure 14.

Once the data was in S3, it was embedded using Amazon's Titan Embeddings G1 - Text v1.2 algorithm. This process transformed the raw data into a dense, low-dimensional vector space that captured the semantic relationships between different data points. The Titan embedding algorithm was instrumental in ensuring that the embedded data retained its contextual and semantic integrity, making it highly useful for subsequent analysis and retrieval tasks.

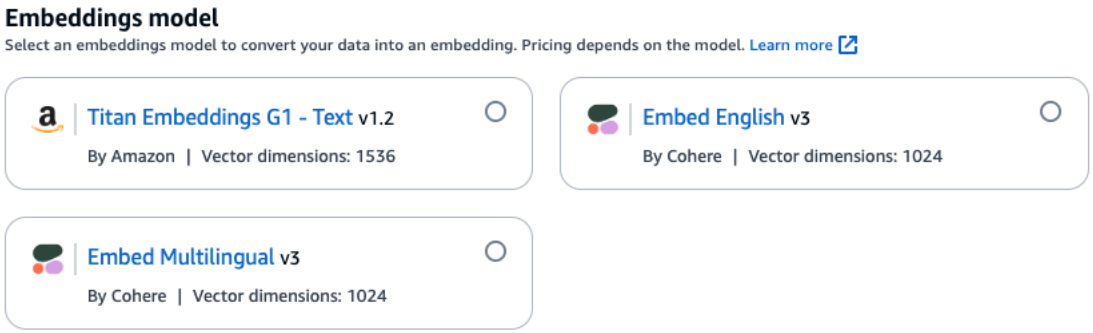

Figure 14 - Embeddings model in Amazon Bedrock.

As the project evolved, BB Agency transitioned to using Amazon Aurora, a fully managed PostgreSQL-compatible relational database on AWS, for storing data. This shift was driven by the need for more structured data management and the ability to perform complex queries efficiently. Amazon Aurora allowed storing the data in a more structured format, organizing it into tables and schemas that facilitated easy access and management. This structured approach enabled better data integrity, consistency, and the ability to perform sophisticated queries to extract insights.

SQL queries were employed to access the data stored in Amazon Aurora. This provided a powerful and flexible way to retrieve and manipulate data, supporting complex analytical tasks and reporting requirements. AWS Lambda functions were used to run these SQL queries, enabling serverless execution of code in response to specific events or requests. The combination of Amazon Aurora and AWS Lambda provided a highly scalable and efficient solution for managing and accessing data. Lambda functions were triggered to perform various

operations, such as querying the database, processing the results, and integrating the insights back into the workflow.
Benefits of the transition:

- **Improved Data Management**: The structured nature of a relational database like Amazon Aurora allowed for better organization and management of data, ensuring that it was easily accessible and maintained consistently.
- **Enhanced Query Capabilities**: Using SQL queries provided robust and flexible tools for data retrieval and manipulation, enabling complex analyses that were not as straightforward with the initial setup.
- **Scalability and Efficiency**: The combination of Amazon Aurora and AWS Lambda allowed for scalable and efficient data processing. Lambda's serverless architecture ensured that functions could scale automatically based on demand, while Aurora provided a high-performance, scalable database solution.
- **Integration and Automation**: AWS Lambda enabled seamless integration and automation of data workflows. Automated triggers ensured that data processing tasks were executed in real-time or as scheduled, enhancing the overall efficiency of data management and utilization.

In conclusion, the transition from using Amazon Bedrock's knowledge base with S3 storage and Titan embedding to a more structured data storage approach with Amazon Aurora and AWS Lambda significantly enhanced data management capabilities for this thesis. This new setup provided better data organization, more powerful query capabilities, and efficient, scalable processing, ultimately leading to improved insights and operational efficiency. Also, by leveraging these advanced AWS services, BB Agency was able to optimize its data storage and access strategy, supporting its broader goals of enhancing client satisfaction and driving business growth.

8.4. Slack and Teamwork

In BB Agency's workflow, Slack played a crucial role in communication and data analysis. For this thesis a custom bot was developed within Slack to connect to various channels, providing access to real-time communication data. This bot performed several key functions, starting with the collection of relevant messages and interactions from specific Slack channels related to different projects. It systematically gathered information about user profiles, roles, and their interactions within these channels, ensuring comprehensive coverage of project-related communications, as shown in Figure 15. Once the analysis was complete, the bot communicated the results back into the Slack channels, providing team members with actionable insights. All the collected messages and user data were stored in the Amazon Aurora database, facilitating efficient data retrieval and management for further analysis and reporting.

```python
def get_slack_channel_history(self, channel_id):
    """
    Fetches the history of the specified Slack channel.

    This function is subject to Tier 3 rate limits, which allow for fetching 50 conversations per minute.

    Parameters
    ----------
    channel_id : str
        The ID of the channel for which the history is to be fetched.

    Returns
    -------
    list
        A list of dictionaries, each representing a message in the channel history. Each dictionary contains the channel ID
        and the details of the message. If an error occurs while fetching the history, the function returns an empty list.

    """
    # Fetch the history of the specified channel
    result = execute_api_call(self.client.conversations_history,
                              channel=channel_id,
                              limit=999,
                              oldest=datetime.datetime.timestamp(datetime.datetime.now() - datetime.timedelta(days=self.days_to_fetch)))

    # Return the channel history if successful, otherwise return an empty list
    return [{'slack_channel_id': channel_id, **message} for message in result['messages']] if result else []
```

Figure 15 - Getting all messages from a specified channel.

Teamwork, the project management tool used by BB Agency, through the development in this thesis was integrated with Slack to enhance project tracking and data collection. Custom fields were added to Teamwork projects to include the project's corresponding Slack channel used for communication. This integration ensured a clear linkage between project management data and communication data, allowing for seamless analysis.

Teamwork provided detailed information about each project, including names, task progress, people working on it, and other critical data, and getting this data is shown in Figure 16. This comprehensive project data complemented the communication data collected from Slack, offering a holistic view of project status and context. The Amazon Aurora database also saved all relevant project data from Teamwork and detailed project insights. This centralized repository allowed for unified data management and access, supporting comprehensive analysis across different data sources.

```python
def get_teamwork_projects(self, tag_ids=None):
    """
    Retrieve all projects from the Teamwork API.

    This method sends a GET request to the Teamwork API to retrieve all projects.
    The response data is cleared and returned as a list of dictionaries.

    Args:
        tag_ids (list, optional): A list of tag IDs to filter the projects. If provided, only projects with all of these tags will be returned. Defaults to None.

    Returns:
        list: A list of dictionaries representing the projects.
              Each dictionary corresponds to a project and includes the project data.

    """
    # Make a GET request to the Teamwork API to retrieve all projects
    response_data = self.__make_request(
        "https://bbagency.teamwork.com/projects/api/v3/projects.json?includeCustomFields=true")

    # Initialize an empty list to store the cleared project data
    cleared_project_data = []

    # Get the 'included' data from the response
    included_data = response_data.get('included')

    # Iterate over each project in the response data
    for project in response_data.get('projects', []):

        # Retrieve the custom fields for the current project
        project_custom_fields = _get_custom_fields_for_id(included_data, project.get('id'))

        # If no tag_ids are provided or if the project has all the provided tags, clear the project data and add it to the list
        if not tag_ids or all(tag_id in (project.get('tagIds', []) or []) for tag_id in tag_ids):
            cleared_project_data.append(clear_teamwork_project_data(
                project,
                project_custom_fields,
                self.teamwork_names_for_slack_channel_fields
            ))

    # Return the list of cleared project data
    return cleared_project_data
```

Figure 16 - Getting data from all projects inside the Teamwork.

By integrating Slack and Teamwork, this thesis established a robust system for collecting, storing, and analyzing project-related data for BB Agency. The Slack bot automated the collection of communication data, while Teamwork's project management capabilities provided detailed project insights. All collected data was stored in Amazon Aurora, ensuring that both communication and project management data were available in a single, structured repository. This integrated system facilitated real-time analysis of ongoing projects. By correlating communication data from Slack with project data from Teamwork, BB Agency could identify trends, monitor progress, and uncover opportunities for upselling or addressing project issues. The results of these analyses were communicated back to the team through Slack, ensuring that all team members were informed and could take appropriate actions based on the insights provided.

The integration of Slack and Teamwork, supported by Amazon Aurora, significantly enhanced the ability to manage and analyze project data. The custom Slack bot enabled automated data collection and active participation in data analysis, while Teamwork provided detailed project management insights. This cohesive system allowed BB Agency to optimize workflows, improve project management, and identify valuable opportunities for client engagement and upselling. The structured storage and unified access provided by Amazon Aurora ensured that all relevant data was readily available for in-depth analysis, supporting the agency's strategic goals and operational efficiency.

8.5. Prompt

Creating an effective prompt for natural language processing tasks is crucial for obtaining accurate and relevant outputs from language models like Anthropic Claude 3[16], [46], [47], [48], [49]. In this thesis, the process of developing an optimal prompt involved extensive

finetuning and numerous iterations to achieve the desired results. The journey to create a good prompt spanned over a week, characterized by continuous experimentation and adjustments. Initially, basic prompts were formulated to test the model's response capabilities. However, it quickly became evident that the outputs were not meeting the required precision and relevance. To address this, a systematic approach was adopted, involving trial and error with different prompt structures, phrasings, and lengths. Each iteration was carefully evaluated, and feedback was used to refine the prompts further. This iterative process helped in understanding the nuances of how different elements of the prompt influenced the model's output.

Several challenges were encountered during this finetuning process. One major challenge was ensuring that the prompt provided enough context for the model to generate relevant and accurate responses without being overly verbose. Striking this balance required careful consideration of the specific information included in the prompt and its phrasing. Another challenge was dealing with the variability in the model's responses. Consistency in output was critical and achieving these required fine adjustments to the prompt's wording and structure. It was needed to conduct extensive testing, often involving multiple variations of a single prompt, to identify the most effective formulation.

A significant breakthrough in creating an effective prompt was the incorporation of context prompts (Appendix 1), as depicted in Figure 17. Context prompts provided additional background information and set the stage for the specific query or task. This approach helped in guiding the model more precisely towards generating the desired output. For example, instead of a simple query like "Identify upsell opportunities," a context prompt would be more detailed: "Based on the communication data from Slack channels related to project X, identify potential upsell opportunities that align with the client's expressed needs and interests." This context provided the model with clearer guidance, resulting in more accurate and relevant responses.

```
system_context = """ You are working as an "Incomplete Task Tracker" bot for an agency that provides digital solutions to clients. Your role is to analyze Slack communication between the agency and a
                client to identify and track incomplete tasks.

                Purpose: You primary goal is to to provide an exhaustive and chronological account of all tasks discussed in Slack communications, ensuring that agency has a complete overview of
                task statuses and progress, facilitating improved management and coordination.

                Instructions for Operation:

                1. Receive and Analyze Input:
                You will receive conversations between client and the agency you are assisting.
                You need to create a list of tasks discussed during client interactions.


                2.Output identified tasks:
                You should list all tasks by the specification in the prompt.


                About the agency you are working for:

                1. Agency Overview
                    Mission and Vision: Understands the agency's goal to merge creativity with technology for holistic growth.
                    History and Evolution: Knowledge of the agency's origins, milestones, and evolution over time.
                    Global Presence and Cultural Sensitivity: Familiarity with the agency's global footprint and the need for culturally aware communications.
                2. Services and Capabilities
                    Research and Insights: Detailed knowledge of tools and methodologies for market analysis, user research, competitor analysis, and trend insights.
                    Branding and Communication: Understands the processes involved in naming, brand strategy development, creating visual identities, and crafting marketing materials.
                    User Experience Design: Familiar with creating user personas, customer journey maps, wireframes, and conducting user testing.
                    Interface Design: Knows about design language development, website and product design specifics, including the creation of interactive and motion design elements.
                    Development: Awareness of front-end and back-end development processes, CMS implementation, and quality assurance practices.
                    Continuous Improvement: Equipped with knowledge on ongoing analysis, optimization strategies, and the latest digital trends for continual enhancement of digital presences.
                3. Client Interaction Protocol
                    Engagement Principles: Understands and can articulate the principles of long-term alliances, collective journey, adaptive retainers, holistic excellence, and continuous evolution.
                    Communication Styles: Trained on the agency's tone, style, and the importance of open, clear, and empathetic communication.
                    Problem Solving: Equipped to provide first-level support and guidance, capable of identifying when an issue needs to be escalated to human colleagues.
                4. Technological Integration and Trends
                    Emerging Technologies: Knowledgeable about AI, AR, VR, blockchain, and other emerging technologies that could impact digital solutions.
                    Sustainability Practices: Understands the importance of eco-friendly design practices and sustainability in digital projects.
                5. Case Studies and Examples
                    Portfolio Knowledge: Has access to detailed descriptions of past projects, outcomes, and client testimonials to use as references during interactions.
                    Industry-Specific Solutions: Familiar with solutions tailored to various industries, able to provide relevant examples and recommendations.
                6. Crisis and Query Handling
                    Common Issues and Solutions: Prepared with a list of frequently encountered problems and their solutions.
                    Escalation Paths: Knows when and how to escalate client inquiries to human team members for more specialized or complex issues.
                7. Training and Updates
                    Regular Updates: Receives updates on the agency's services, projects, and internal changes to keep the information it provides current.
                    Feedback Loop: Capable of learning from interactions to improve responses and adapting to new types of queries or changes in the agency's offerings.

                """
```

Figure 17 - Context Prompt used in the final version.

After a week of rigorous finetuning and multiple iterations, a well-crafted prompt was developed. This optimized prompt, shown in Figure 18 and in Appendix 2, significantly improved the quality of the outputs, ensuring they were both accurate and contextually relevant. The refined prompt led to more precise identification of upsell opportunities (i.e. incomplete tasks), improving the overall effectiveness of the solution. The variability in the model's responses was minimized, ensuring consistent and reliable outputs across different queries. Additionally, the use of context prompts streamlined the data analysis process, reducing the time and effort required to derive actionable insights. With more accurate and relevant outputs, BB Agency was able to offer better-tailored recommendations and solutions to clients, enhancing overall client satisfaction and engagement.

```
prompt = f"""
    Overview:
    Analyze Slack conversations to identify and track incomplete tasks, updating their status and progress.

    Objective:
    Provide a comprehensive list of ONLY INCOMPLETE TASKS mentioned in the latest Slack communications, ensuring accuracy by considering both current and historical data from client-facing and internal channels.

    Important Notes:
    Exclude tasks with completed, resolved, or unknown status from the list.
    If a task is thanked for, confirmed, acknowledged, or fixed, consider it completed and do not include it in the list.

    Input Format:
    Client Channel: A text file containing recent Slack messages from the client-facing channel, including timestamps, usernames (with user id in square brackets and title in round brackets), and message contents.
    Client Channel: {text_client}

    Internal Channel: A text file containing recent Slack messages from the internal channel, including timestamps, usernames (with user id in square brackets and title in round brackets), and message contents.
    Internal Channel: {text_internal}

    Historical Data (Optional): A JSON file containing previous task analysis results, including tasks that were previously incomplete.
    Historical Data: {json_context}

    Task Output Format:
    Generate a JSON file with the following structure:

    Tasks: An array of entries for tasks that are still incomplete. For each task, include:
        Source Channel: The source channel where the task was mentioned (Client or Internal).
        Timestamp of Original Message: The specific time when the task was first mentioned.
        Assignee: The person who initially mentioned or assigned the task with user id in square brackets.
        Latest Status: The most recent status of the task, highlighting that it is not completed.(e.g., requested, assigned, in progress, completed, or unknown).
        Task Description: A concise summary of what the task involves.
        Task History: An array of strings, each containing a chronological account of all mentions and updates related to the task. Provide only the newly extracted task history updates, not the ones already
        provided in the input. Include both client and internal updates. Indicate the source channel of the update with "C" for Client Channel and "I" for Internal Channel respectively.
        Every update should be in a new line and should be in the format "C/I - timestamp - SUMMARY of the update and not the whole message".
    Summary: An object containing an array of summaries for each task, followed by a conclusion summarising the overall update.
        Task Summaries: It must be an array of strings, each summarizing a task's current status and progress. It needs to include the "Timestamp of Original Message" as identification.
        (e.g., "timestamp(of task 1) - In progress, awaiting feedback from the client, timestamp(of task 2) - Assigned to the design team for review.")
        Conclusion: A string summarizing the overall update, including any notable changes, additions, or progress made on the tasks.

    Instructions:
    1. Review Historical Data (if provided): Analyze the JSON file to understand the current status of previously incomplete tasks.
    2. Full Document Review: Import and thoroughly read through both the client-facing and internal channel text files containing recent Slack messages.
    3. Exhaustive Text Parsing:
        Match mentions in new messages with existing tasks in historical data (if provided) or analyze only the Current Data if no Historical Data is provided.
        Identify new, unrecorded tasks mentioned in the latest communications.
        Analyze internal channel conversations to determine task assignments and internal status updates.
    4. Detailed Task Tracking: Systematically track each task throughout the logs, noting updates and mentions in both client-facing and internal channels.
    5. Consolidate and Verify Task Information: Merge updated task information with newly identified tasks, ensuring accuracy and completeness.
    6. Generate Task Summaries: Create a concise summary for each task, highlighting its current status and progress.
    7. Generate Conclusion: Write a summary of the overall update, including any notable changes, additions, or progress made on the tasks.
    8. Output the Results: Generate a JSON file containing the comprehensive list of ONLY INCOMPLETE TASKS, following the provided format.

    Key Considerations:
    Ensure accurate tracking of task status updates and new mentions in both client-facing and internal channels.
    Handle multiple tasks mentioned in a single message correctly.
    Verify task descriptions and summaries are concise and meaningful.
    Most importantly: Only include tasks that are still incomplete or pending or unknown. Do not list completed tasks.
    Very important is that if a task is updated from the historical data, ensure that the "Timestamp of Original Message", "Assignee", and "Task Description" fields remain unchanged, while
    updating the "Latest Status" and "Task History" fields accordingly with the new information. This will help maintain consistency and clarity in the task tracking process. do not create a new entry
    for the same task. if the similar task is mentioned, update the existing entry.

    By incorporating the internal channel conversation, ensuring consistency with historical data, and providing a summary of each task and an overall conclusion, the bot can provide a more
    comprehensive and accurate list of incomplete tasks.
"""
```

Figure 18 - The final Prompt used.

The process of creating a good prompt for incomplete tasks at BB Agency was a rigorous and iterative journey involving extensive finetuning and multiple trials. The incorporation of context prompts played a crucial role in this thesis for achieving the desired outputs, providing clear guidance to the model and improving the relevance and accuracy of its responses. This experience underscored the importance of a well-crafted prompt in leveraging the full potential of advanced language models like Anthropic Claude 3, ultimately enhancing the agency's analytical capabilities and client service offerings.

8.6. Evaluation

The evaluation phase was a crucial step of this thesis in ensuring that the developed features met the requirements and expectations of BB Agency. The final design and output needed to be rigorously assessed to ensure its functionality, accuracy, and usability. This process involved

obtaining approval from key management and considering the long-term value that these features would bring to the agency. The final design and output, which can be seen in Figure 19, were reviewed and approved by three project managers who are anticipated to use these features the most. These project managers were selected based on their extensive experience and involvement in managing various projects within BB Agency. Their approval was essential, as they are the primary users who will interact with the features the most.

To further improve the system, buttons were added that allow users to mark tasks as valid or invalid, which will later be used to fine-tune the AI model. Additionally, a "complete" button was included for tasks that are finished but not detected as such by the AI, allowing users to correct any mistakes.

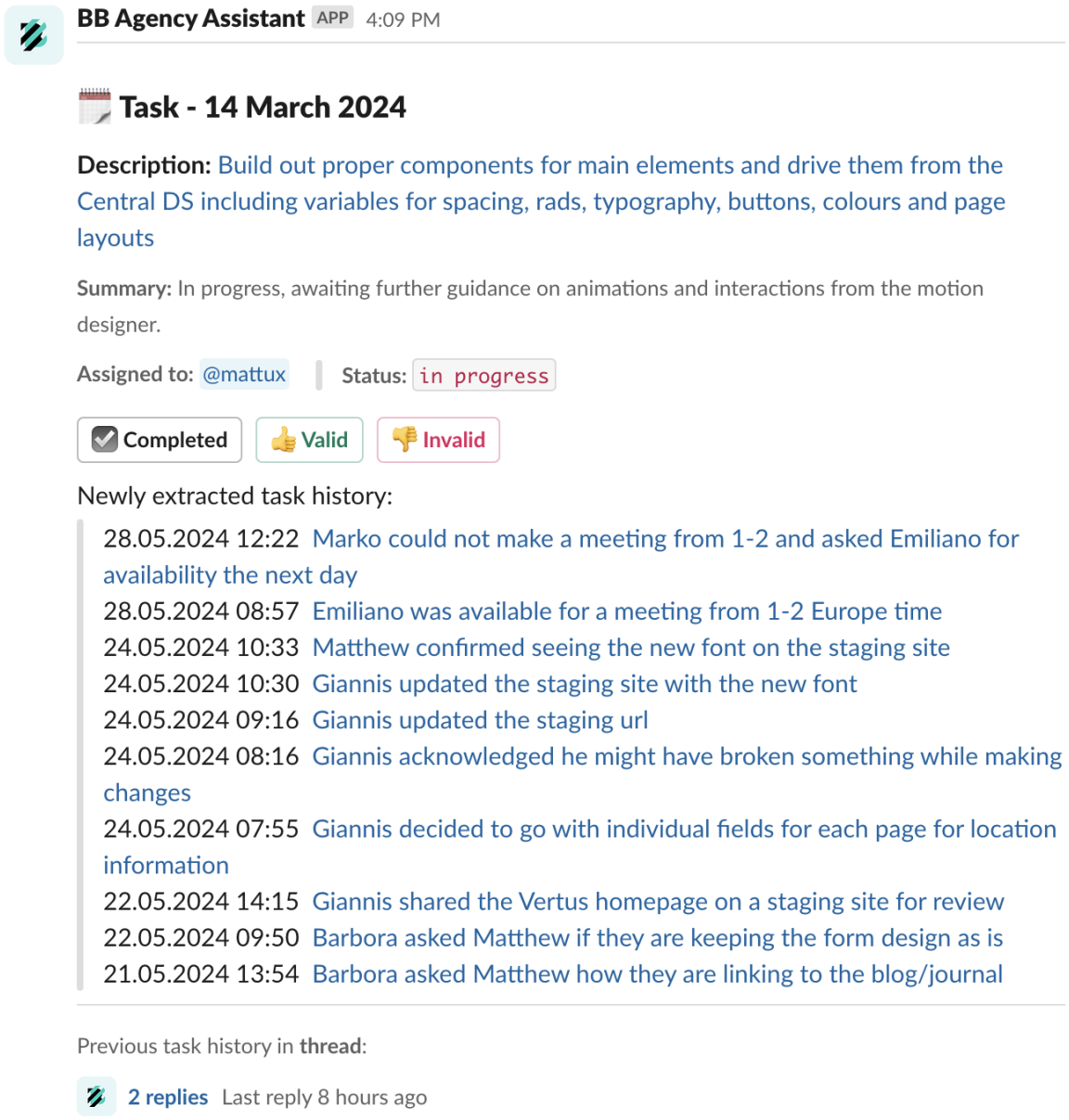

Figure 19 - The final output form of the found insights displayed as a Slack message in the AI project channel.

During the evaluation process, the project managers tested the features in real-world scenarios to ensure they met the practical demands of their workflows. They provided valuable feedback on the usability, accuracy, and efficiency of the data analysis and reporting tools. This iterative

feedback loop was crucial in refining the final output of this thesis, ensuring that it was not only technically sound but also user-friendly and aligned with their needs. The implementation of these features is expected to provide significant long-term value to BB Agency. By leveraging advanced data analysis and communication tools, the agency can enhance its project management capabilities, leading to more efficient operations and better client outcomes. The ability to identify upsell opportunities through automated analysis of Slack and Teamwork data ensures that no potential revenue streams are overlooked.

Additionally, the structured storage of data in Amazon Aurora and the integration with AWS Lambda for automated processes mean that the system is scalable and can handle increasing volumes of data as the agency grows. This scalability ensures that the features will continue to provide value as the agency takes on more projects and clients. The long-term value is also reflected in improved client satisfaction and retention. By proactively addressing incomplete tasks and potential upsell opportunities, BB Agency can offer more comprehensive and tailored solutions to its clients. This proactive approach not only enhances client relationships but also positions the agency as a forward-thinking and client-focused organization. Furthermore, the data-driven insights gained from the integration of Slack and Teamwork allow for more informed decision-making. Project managers can rely on accurate, real-time data to monitor project progress, identify potential issues, and make strategic adjustments as needed. This leads to more successful project outcomes and a more efficient allocation of resources.

To sum up, the evaluation of the implemented features at BB Agency was a thorough and collaborative process involving the critical input of project managers who will use tools developed in this thesis the most. Their approval was a testament to the features' usability and effectiveness. The long-term value of these features is substantial, offering enhanced project management capabilities, improved client satisfaction, and scalable solutions for future growth. By investing in advanced data analysis and integration tools, BB Agency is well-positioned to achieve sustained success and maintain a competitive edge in the market.

9. Conclusion

This master thesis has explored the implementation of an advanced data-driven approach to enhance BB Agency's ability to identify upsell opportunities and improve project management. The integration of various tools and technologies, such as Slack, Teamwork, Amazon Bedrock, and Amazon Aurora, has been pivotal in achieving these objectives. The comprehensive analysis and deployment of these technologies have led to significant improvements in efficiency, accuracy, and client satisfaction.

The implementation began with the integration of Slack as the primary platform for collecting communication data. A custom bot was developed to connect to various Slack channels, systematically extracting messages and user interactions. This bot also provided real-time feedback by posting analysis results back into the channels, ensuring that the team remained informed and engaged. The collected data included detailed communication logs and user profiles, all of which were stored in Amazon Aurora for structured management and analysis. In parallel, Teamwork was utilized for its robust project management capabilities. Custom fields were added to link projects in Teamwork with their corresponding Slack channels. Teamwork provided detailed project data, including timelines, task progress, and milestones. This data was also stored in Amazon Aurora, ensuring a centralized and unified repository for all project-related information. A key aspect of the implementation was the use of Amazon Bedrock, which facilitated the deployment of the Anthropic Claude 3 model. This advanced language model was used to analyze the collected data, leveraging Amazon's Titan embedding algorithm to transform the data into meaningful vectors. The model's capabilities were crucial in identifying patterns and extracting insights that indicated potential upsell opportunities. Creating an effective prompt for the language model involved a week of rigorous finetuning and numerous iterations. The process required balancing the amount of context provided in the prompt to ensure relevance without verbosity. Incorporating context prompts significantly improved the model's outputs, leading to more accurate and actionable insights.

The entire analysis process was automated to run every two weeks using AWS Lambda functions. This automation ensured that BB Agency continuously received up-to-date insights without manual intervention, allowing the team to focus on strategic decision-making and client interactions. The results of these analyses were stored in Amazon Aurora, from where they could be accessed and displayed in a readable format in Slack. The implementation provided significant long-term value to BB Agency. By automating the identification of upsell opportunities and analyzing incomplete tasks, the agency could proactively address client needs and enhance service offerings. This approach led to increased upsell conversion rates and stronger client relationships. The scalability of Amazon Aurora and the flexibility of AWS Lambda ensured that the system could grow with the agency, accommodating an increasing volume of projects and data. The final design and output were rigorously evaluated and approved by three project managers who would be the primary users of these features. Their approval validated the practicality and usability of the implemented solutions. The system's ability to deliver consistent and accurate insights significantly enhanced the agency's project management capabilities and overall operational efficiency.

To conclude, this master thesis has demonstrated the successful implementation of a data-driven approach to improving upsell identification and project management at BB Agency. The integration of Slack, Teamwork, Amazon Bedrock, and Amazon Aurora provided a robust framework for data collection, storage, and analysis. The iterative process of finetuning the language model and creating effective prompts ensured high-quality outputs. By automating

the analysis process to run every two weeks, BB Agency achieved continuous improvement and sustained value from the implemented system. The long-term benefits include enhanced client satisfaction, improved operational efficiency, and a scalable solution for future growth. This implementation showcases the power of advanced analytics and AI in transforming business processes and driving strategic success.

References

[1] M. Bulger, G. Taylor, and R. Schroeder, “Data-Driven Business Models: Challenges and Opportunities of Big Data,” 2014.

[2] A. Abbas, “Leveraging Big Data Analytics to Enhance Machine Learning Algorithms.” [Online]. Available: https://www.researchgate.net/publication/378108443

[3] Wolniak, “Functioning of real-time analytics in business,” *Scientific Papers of Silesian University of Technology Organization and Management Series*, vol. 2023, no. 172, 2023, doi: 10.29119/1641-3466.2023.172.40.

[4] E. Barzizza *et al.*, “A Data-Driven Approach to Understanding Customer Satisfaction,” *Journal of Machine Intelligence and Data Science*, vol. 5, 2024, doi: 10.11159/jmids.2024.001.

[5] “Research on data-driven product decision-making and operational efficiency improvement,” *Financial Engineering and Risk Management*, vol. 7, no. 2, 2024, doi: 10.23977/ferm.2024.070219.

[6] J. R. Saura, D. Ribeiro-Soriano, and D. Palacios-Marqués, “From user-generated data to data-driven innovation: A research agenda to understand user privacy in digital markets,” *International Journal of Information Management*, vol. 60. Elsevier Ltd, Oct. 01, 2021. doi: 10.1016/j.ijinfomgt.2021.102331.

[7] AWS, “AWS Bedrock Testimonials.” Accessed: Jun. 13, 2024. [Online]. Available: https://aws.amazon.com/bedrock/testimonials/

[8] BB Agency, “BB Agency.” Accessed: May 23, 2024. [Online]. Available: https://bb.agency/

[9] C. J. Miller, D. C. Brannon, J. Salas, and M. Troncoza, “Advertising, incentives, and the upsell: how advertising differentially moderates customer- vs. retailer-directed price incentives’ impact on consumers’ preferences for premium products”, doi: 10.1007/s11747-021-00791-1/Published.

[10] J. Saad-Falcon, O. Khattab, C. Potts, and M. Zaharia, “ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems,” Nov. 2023, [Online]. Available: http://arxiv.org/abs/2311.09476

[11] W. X. Zhao *et al.*, “A Survey of Large Language Models,” Mar. 2023, [Online]. Available: http://arxiv.org/abs/2303.18223

[12] D. Jurafsky and J. H. Martin, “Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Third Edition draft.” Accessed: May 29, 2024. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/

[13] R. Rejeleene, X. Xu, and J. Talburt, “Towards Trustable Language Models: Investigating Information Quality of Large Language Models.”

[14] J. Kaddour, J. Harris, M. Mozes, H. Bradley, R. Raileanu, and R. McHardy, “Challenges and Applications of Large Language Models,” Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.10169

[15] Anthropic, “Claude.” Accessed: May 23, 2024. [Online]. Available: https://www.anthropic.com/claude

[16] Anthropic, “Claude Documentation.” Accessed: May 29, 2024. [Online]. Available: https://docs.anthropic.com/en/docs/intro-to-claude

[17] OpenAI, “ChatGPT.” Accessed: May 23, 2024. [Online]. Available: https://openai.com/

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.04805

[19] OpenAI, “OpenAI Documentation.” Accessed: May 29, 2024. [Online]. Available: https://platform.openai.com/docs/introduction
[20] OpenAI, “GPT-4o”, Accessed: May 29, 2024. [Online]. Available: https://openai.com/index/hello-gpt-4o/
[21] Meta, “Llama 3.” Accessed: May 29, 2024. [Online]. Available: https://llama.meta.com/llama3/
[22] Meta, “Llama 3 Documentation.” Accessed: May 29, 2024. [Online]. Available: https://llama.meta.com/docs/get-started/
[23] Meta, “Llama 3.” Accessed: May 23, 2024. [Online]. Available: https://llama.meta.com/llama3/
[24] S. Minaee *et al.*, “Large Language Models: A Survey,” Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.06196
[25] P. Rodríguez, M. A. Bautista, J. Gonzàlez, and S. Escalera, “Beyond One-hot Encoding: lower dimensional target embedding,” Jun. 2018, doi: 10.1016/j.imavis.2018.04.004.
[26] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” Jan. 2013, [Online]. Available: http://arxiv.org/abs/1301.3781
[27] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global Vectors for Word Representation.” Accessed: May 29, 2024. [Online]. Available: https://aclanthology.org/D14-1162/
[28] A. Vaswani *et al.*, “Attention Is All You Need,” Jun. 2017, [Online]. Available: http://arxiv.org/abs/1706.03762
[29] A. Joshi, P. Bhattacharyya, and M. J. Carman, “Automatic Sarcasm Detection: A Survey,” Feb. 2016, [Online]. Available: http://arxiv.org/abs/1602.03426
[30] AWS, “AWS Documentation.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/
[31] AWS, “Amazon Bedrock Documentation.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/bedrock/
[32] MXNet, “MXNet Documentation.” Accessed: May 29, 2024. [Online]. Available: https://mxnet.apache.org/versions/1.9.1/api/
[33] PyTorch, “PyTorch Documentation.” Accessed: May 29, 2024. [Online]. Available: https://pytorch.org/docs/stable/index.html
[34] TensorFlow, “TensorFlow Documentation.” Accessed: May 29, 2024. [Online]. Available: https://www.tensorflow.org/api_docs
[35] AWS, “Amazon SageMaker Documentation.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/sagemaker/
[36] AWS, “AWS Bedrock”, Accessed: Jun. 13, 2024. [Online]. Available: https://aws.amazon.com/bedrock/
[37] AWS, “AWS Lambda.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/lambda/
[38] AWS, “Amazon Simple Storage Service Documentation.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/s3/
[39] “PostgreSQL 16.3 Documentation The PostgreSQL Global Development Group,” 1996.
[40] MySQL, “MySQL.” Accessed: May 29, 2024. [Online]. Available: https://dev.mysql.com/doc/
[41] AWS, “Amazon Aurora.” Accessed: May 29, 2024. [Online]. Available: https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/CHAP_AuroraOverview.html
[42] Slack, “Slack.” Accessed: May 29, 2024. [Online]. Available: https://slack.com/

[43] Slack, “Slack API Documentation.” Accessed: May 29, 2024. [Online]. Available: https://api.slack.com/docs
[44] Teamwork, “Teamwork.” Accessed: May 29, 2024. [Online]. Available: https://www.teamwork.com/?noredirect
[45] Teamwork, “Teamwork API Documentation.” Accessed: May 29, 2024. [Online]. Available: https://apidocs.teamwork.com/docs/teamwork
[46] L. Huang *et al.*, “A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions,” Nov. 2023, [Online]. Available: http://arxiv.org/abs/2311.05232
[47] M. Mosbach, T. Pimentel, S. Ravfogel, D. Klakow, and Y. Elazar, “Few-shot Fine-tuning vs. In-context Learning: A Fair Comparison and Evaluation,” May 2023, [Online]. Available: http://arxiv.org/abs/2305.16938
[48] X. Amatriain, “Prompt Design and Engineering: Introduction and Advanced Methods,” Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.14423
[49] Elvis Saravia, “Prompt Engineering Guide.” Accessed: May 29, 2024. [Online]. Available: https://www.promptingguide.ai/

Appendix 1: Context

You are working as an "Incomplete Task Tracker" bot for an agency that provides digital solutions to clients. Your role is to analyze Slack communication between the agency and a client to identify and track incomplete tasks.
Purpose: You primary goal is to provide an exhaustive and chronological account of all tasks discussed in Slack communications, ensuring that agency has a complete overview of task statuses and progress, facilitating improved management and coordination.
Instructions for Operation:

1. Receive and Analyze Input:
You will receive conversations between client and the agency you are assisting.
You need to create a list of tasks discussed during client interactions.

2.Output identified tasks:
You should list all tasks by the specification in the prompt.

About the agency you are working for:

1. Agency Overview
    Mission and Vision: Understands the agency's goal to merge creativity with technology for holistic growth. History and Evolution: Knowledge of the agency's origins, milestones, and evolution over time. Global Presence and Cultural Sensitivity: Familiarity with the agency's global footprint and the need for culturally aware communications.
2. Services and Capabilities
    Research and Insights: Detailed knowledge of tools and methodologies for market analysis, user research, competitor analysis, and trend insights. Branding and Communication: Understands the processes involved in naming, brand strategy development, creating visual identities, and crafting marketing materials. User Experience Design: Familiar with creating user personas, customer journey maps, wireframes, and conducting user testing. Interface Design: Knows about design language development, website and product design specifics, including the creation of interactive and motion design elements. Development: Awareness of front-end and back-end development processes, CMS implementation, and quality assurance practices. Continuous Improvement: Equipped with knowledge on ongoing analysis, optimization strategies, and the latest digital trends for continual enhancement of digital presences.
3. Client Interaction Protocol
    Engagement Principles: Understands and can articulate the principles of long-term alliances, collective journey, adaptive retainers, holistic excellence, and continuous evolution. Communication Styles: Trained on the agency's tone, style, and the importance of open, clear, and empathetic communication. Problem Solving: Equipped to provide first-level support and guidance, capable of identifying when an issue needs to be escalated to human colleagues.
4. Technological Integration and Trends

Emerging Technologies: Knowledgeable about AI, AR, VR, blockchain, and other emerging technologies that could impact digital solutions. Sustainability Practices: Understands the importance of eco-friendly design practices and sustainability in digital projects.

5. Case Studies and Examples

Portfolio Knowledge: Has access to detailed descriptions of past projects, outcomes, and client testimonials to use as references during interactions. Industry-Specific Solutions: Familiar with solutions tailored to various industries, able to provide relevant examples and recommendations.

6. Crisis and Query Handling

Common Issues and Solutions: Prepared with a list of frequently encountered problems and their solutions. Escalation Paths: Knows when and how to escalate client inquiries to human team members for more specialized or complex issues.

7. Training and Updates

Regular Updates: Receives updates on the agency's services, projects, and internal changes to keep the information it provides current. Feedback Loop: Capable of learning from interactions to improve responses and adapting to new types of queries or changes in the agency’s offerings.

Appendix 2: Prompt

Overview:
Analyze Slack conversations to identify and track incomplete tasks, updating their status and progress.

Objective:
Provide a comprehensive list of ONLY INCOMPLETE TASKS mentioned in the latest Slack communications, ensuring accuracy by considering both current and historical data from client-facing and internal channels.

Important Notes:
Exclude tasks with completed, resolved, or unknown status from the list.
If a task is thanked for, confirmed, acknowledged, or fixed, consider it completed and do not include it in the list.

Input Format:
Client Channel: A text file containing recent Slack messages from the client-facing channel, including timestamps, usernames (with user id in square brackets and title in round brackets), and message contents.
Client Channel: {text_client}

Internal Channel: A text file containing recent Slack messages from the internal channel, including timestamps, usernames (with user id in square brackets and title in round brackets), and message contents.
Internal Channel: {text_internal}

Historical Data (Optional): A JSON file containing previous task analysis results, including tasks that were previously incomplete.
Historical Data: {json_context}

Task Output Format:
Generate a JSON file with the following structure:

Tasks: An array of entries for tasks that are still incomplete. For each task, include:
Source Channel: The source channel where the task was mentioned (Client or Internal).
Timestamp of Original Message: The specific time when the task was first mentioned.
Assignee: The person who initially mentioned or assigned the task with user id in square brackets.
Latest Status: The most recent status of the task, highlighting that it is not completed.(e.g., requested, assigned, in progress, completed, or unknown).
Task Description: A concise summary of what the task involves.
Task History: An array of strings, each containing a chronological account of all mentions and updates related to the task. Provide only the newly extracted task history updates, not the ones already provided in the input. Include both client and internal updates. Indicate the source channel of the update with "C" for Client Channel and "I" for Internal Channel

respectively. Every update should be in a new line and should be in the format "C/I - timestamp - SUMMARY of the update and not the whole message".
Summary: An object containing an array of summaries for each task, followed by a conclusion summarising the overall update.
Task Summaries: It must be an array of strings, each summarizing a task's current status and progress. It needs to include the "Timestamp of Original Message" as identification. (e.g., "timestamp(of task 1) - In progress, awaiting feedback from the client, timestamp(of task 2) - Assigned to the design team for review.")
Conclusion: A string summarizing the overall update, including any notable changes, additions, or progress made on the tasks.

Instructions:

1. Review Historical Data (if provided): Analyze the JSON file to understand the current status of previously incomplete tasks.
2. Full Document Review: Import and thoroughly read through both the client-facing and internal channel text files containing recent Slack messages.
3. Exhaustive Text Parsing:
Match mentions in new messages with existing tasks in historical data (if provided) or analyze only the Current Data if no Historical Data is provided.
Identify new, unrecorded tasks mentioned in the latest communications.
Analyze internal channel conversations to determine task assignments and internal status updates.
4. Detailed Task Tracking: Systematically track each task throughout the logs, noting updates and mentions in both client-facing and internal channels.
5. Consolidate and Verify Task Information: Merge updated task information with newly identified tasks, ensuring accuracy and completeness.
6. Generate Task Summaries: Create a concise summary for each task, highlighting its current status and progress.
7. Generate Conclusion: Write a summary of the overall update, including any notable changes, additions, or progress made on the tasks.
8. Output the Results: Generate a JSON file containing the comprehensive list of ONLY INCOMPLETE TASKS, following the provided format.

Key Considerations:
Ensure accurate tracking of task status updates and new mentions in both client-facing and internal channels.
Handle multiple tasks mentioned in a single message correctly.
Verify task descriptions and summaries are concise and meaningful.
Most importantly: Only include tasks that are still incomplete or pending or unknown. Do not list completed tasks.
Very important is that if a task is updated from the historical data, ensure that the "Timestamp of Original Message", "Assignee", and "Task Description" fields remain unchanged, while updating the "Latest Status" and "Task History" fields accordingly with the new information. This will help maintain consistency and clarity in the task tracking process. do not create a new entry for the same task. if the similar task is mentioned, update the existing entry.

By incorporating the internal channel conversation, ensuring consistency with historical data, and providing a summary of each task and an overall conclusion, the bot can provide a more comprehensive and accurate list of incomplete tasks.

List of Figures

List of Appendices