Department of Business and Management

B.Sc. in Management and Computer Science

Chair of Machine Learning and Artificial Intelligence

Beyond Fact-checking: Harnessing Retrieval-Augmented Generation and Large Language Models for Tackling Financial Misinformation

Supervisor:
Prof.
Giuseppe F. Italiano

Candidate:
Matteo Carucci
ID No. 269261

Academic Year 2023/2024

Abstract

Financial misinformation poses a significant threat to global economic stability, necessitating robust fact-checking mechanisms. This thesis explores the application of Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) Fact-Checking methods to mitigate the huge effects that Financial Fake News have on the world economy. We investigate the capabilities of RAG-enhanced LLMs to provide accurate and up-to-date financial information by addressing the limitations of traditional LLMs such as hallucinations and knowledge cut-off dates. Large Language Models are usually bound by a cutoff date (usually the last date of information in the training data) and hallucinations, events in which machine learning models, specifically large language models (LLMs) like GPT-4, produce outputs that are coherent and grammatically correct but factually incorrect or nonsensical. Our study introduces a mixed design methodology that combines Self-RAG, Adaptive-RAG and Corrective RAG techniques powered by AI agents. We apply this methodology to a case study involving Apple's stock market news, utilizing a dataset of Nasdaq financial news articles from November to December 2023. Through detailed data processing, embedding, retrieval, and grading mechanisms, our system demonstrates significant results in the accuracy and relevance of generated responses. The outcomes highlight the effectiveness of RAG-enhanced LLMs in reducing hallucinations and providing contextually relevant answers, and obtaining high accuracy, precision, recall, and F1 scores. Through this study of innovative state-of-the-art RAG architectures and their implications on Fact-Checking and information retrieval, this work also aims at showing and describing the current capabilities and limitations in enhancing LLMs with Retrieval Augmented Generation (RAG) for generating reliable trustworthy timely and accurate information, with a special emphasis on financial data and insights.

Contents

Chapter 1

Introduction

1.1 Background

In the complex world of modern economies, accurate financial information is paramount for decision-making processes ranging from individual investment choices to the strategic planning of listed companies that inevitably rely on stock markets. The vitality of precise and reliable financial data cannot be overlooked; it underpins investor confidence, guides policy formulation, and shapes the economic landscape. Financial information providers like Bloomberg and Thomson Reuters demonstrate how significant timely and accurate economic and financial information are for investors and large financial institutions; the former, for instance, is a financial media company that make over 20 Billion a year by providing financial data and insights to over 325000 financial professionals that individually pay 25000$ for a year subscription to the Bloomberg Terminal, a computer providing the most intricate and detailed economic and financial insights. This shows how much financial investors rely on accurate and timely financial information; that is magnified in an era where rapid information exchange and the pervasive influence of digital media platforms are often sources of unverified, misleading and inaccurate financial information spread for personal interests. In August 2018 Elon Musk CEO of Tesla tweeted that he considered taking Tesla private at $420 per share and that he had "funding secured" for such a deal. This tweet caused a significant surge in Tesla's stock price even though the funding was not secured and the deal was far from certain. The United States Securities and Exchange Commission (SEC) later charged Musk with secu-

rities fraud for the misleading tweets. The incident led to a settlement that included fines for both Musk and Tesla and Musk agreeing to step down as Tesla's chairman for a period. This small example shows how critical information can be to disrupt and affect financial markets; within this ecosystem journalism plays a crucial role acting as a conduit for verified and factual information and as a guardian against misinformation.

1.2 The Pillars of Financial Stability

Financial markets are founded on the principles of risk and reward influenced heavily by information accuracy and timeliness. Investors rely on rapid and accurate data to make informed decisions; whether it's buying stocks, bonds or making other investment choices. The effects of misinformation can be enormous, leading to misallocated resources, inflated asset bubbles or sudden panic selling. The integrity of financial markets therefore is closely linked to the quality of information circulating in them.

1.3 The Economic Impact of Misinformation

The consequences of financial misinformation are vast. At the micro level, it can lead to poor investment decisions, resulting in significant personal losses. At the macro level, pervasive misinformation can worsen the efficiency of markets, distort asset valuations, and potentially trigger financial crises. The global economy, which is interconnected and interdependent, is particularly vulnerable to such disruptions, highlighting the need for vigilance in maintaining the accuracy of financial information.

1.4 The importance of accurate financial information and the role of journalism.

Journalism holds a central place in the ecosystem of financial information. Traditional media outlets financial analysts and increasingly online platforms serve as intermediaries filtering interpreting and disseminating data to the public. This role is not merely about reporting figures and trends but involves contextualizing information highlighting potential implications and critically

fact-checking to ensure reliability and trustworthiness of data that is used by investors. The advent of digital media has transformed the landscape democratizing information dissemination but also complicating the battle against misinformation. The speed at which information spreads online can prevent and obstruct verification efforts, making the journalist's role more challenging yet ever more crucial. Fact-checking is a fundamental journalistic function that emerged as a primary defence against the challenge of financial fake news, underscoring the need for skilled practitioners equipped with the right tools and methodologies. Unclear financial reporting and fraud in the corporate and financial sector necessitate an entity that questions corporate practices and behaviour. Therefore financial journalists take a crucial societal role in acting as watchdogs for the corporate and financial sectors. However these watchdog role perceptions and enactment can vary greatly across outlets. In fact acting as a watchdog could mean respectively: (1) transmitting information to the public (2) providing guidance for acting upon this information (3) the way a journalist works (e.g. investigative journalism vs daily journalism) or (4) the focus on results and to what extent journalistic work has yielded people or the government to act. Similarly more general research on journalistic role perceptions distinguishes between passive role perceptions (e.g. passive mirror public forum) and active role perceptions while the latter also includes the watchdog role and the public mobilizer role. (N. Strauß, 2018)

1.5 The Evolution of Financial Journalism

The field of financial journalism has evolved significantly adapting to technological advancements and shifting investors' preferences and behaviours. From the days of ticker-tape machines and printed stock reports markets are nowadays fed with real-time digital news feeds and interactive financial platforms as the means of information delivery have transformed. This evolution has enhanced accessibility and immediacy but also introduced challenges in maintaining accuracy and depth in reporting crucial information that has a dramatic impact on markets. In response financial journalism is increasingly leveraging technological innovations such as artificial intelligence (AI) and big data analytics to enhance reporting accuracy and efficiency. Companies like Bloomberg and Reuters have implemented AI-driven systems such as Bloomberg's Cyborg to quickly produce financial news stories from earn-

ings reports increasing the speed and volume of news production without sacrificing accuracy. The financial giant also created BloombergGPT a 50 billion-parameter general-purpose financial LLM trained on "FINPILE" a dataset consisting of a range of English financial documents including news filings press releases web-scraped financial documents and social media drawn from the Bloomberg archives (Shijie Wu et al., 2023). Another example of this are the novel "Robo-Advisors" introduced by large investment banks over the last years which provide a cost-effective and immediate way to receive financial decision-making advice. These are online services that use computer algorithms to provide financial advice and manage customers' investment portfolios driven by LLMs models trained on Financial Datasets (Jill E. Fisch et al., 2019).

As these models become more powerful and useful for investors, data used for training these models is crucial to produce factual and reliable information. Financial data analysts and engineers' figures are highly demanded by financial data providers which mostly focus on Data Quality, one of the most pressing and difficult challenges that AI, Big Data and Data Science researchers are trying to tackle as LLMs and Machine Learning models are positively impacted by high quality data.

1.6 Research Questions and Thesis structure: Main questions guiding the research.

As the major issues caused by financial misinformation and disinformation were introduced, the next ideal step would be to identify and present the possible remedies to those highly costing phenomena. In the subsequent chapters, a comprehensive literature review of RAG, LLMs, and current fact-checking industry standards using Machine Learning standards will be demystified showing their possible applications and highlighting their advantages over some alternatives as well as their inherent limitations and points of improvement that can be addressed. After the review, an introduction to the methodologies and architectures used in the experiment will be studied particularly delving into the data collection (where it was sourced what were its fields etc.) and frameworks to build RAG architectures and open-source Large Language Models such as LangChain and Meta open-source Large Language model Llama3 among others.

In Chapter 4 instead the case study is thoroughly assessed introducing potential metrics and benchmarks to gauge the model performance in avoiding Hallucinations and ensuring Fact-Checking via Retrieval Augmented Generation. In particular, this case study will be revolved around a specific case.

- Some possible questions on the Stock Market targeting some specific occurrences in the market mainly focusing on a publicly traded company (Apple) (i.e. What was the price of Apple stock in December 2023? What did Warren Buffet say on Apple's performance in 2023 Q3?).

In this case, the objective is to showcase the capabilities of RAG-enhanced LLMs in differentiating between legitimate news and misinformation in the rapidly fluctuating stock market. The study aims to illustrate how these models can contextualize news items by cross-referencing (or simply referring to information stored in RAG Databases) multiple data sources (web search and a dataset in our case). This solution will be compared to traditional LLMs and manual fact-checking processes that can be tedious and time-consuming. Ultimately, in Chapters 5 and 6, the challenges and limitations will be discussed, focusing on identifying resolution points and suggestions for further development and study to be carried out by journalists and fact-checkers interested in refining their fact-checking methodologies and recommendations to investors who want to build agnostic systems able to detect misinformation that can lead to economic losses and market instability.

Chapter 2

Literature Review

2.1 Financial Misinformation: Definitions, Types, and Examples

The internet has completely transformed the way people access and disseminate information. It has changed various aspects of daily life including how individuals conduct financial transactions: The advent of online banking, investment, and trading platforms has made it easier for individuals to access financial information and keep updated on the latest financial news affecting markets. However, the widespread availability of financial information online has also led to an increase in the spread of financial misinformation. A typical mistake is to confuse and not differentiate between financial misinformation and disinformation as they have distinct implications. Misinformation specifically in the financial industry refers to the issuing and/or spread of false and/or inaccurate information, most often intentionally, that lead readers to misinterpret facts and events. It is mostly done to manipulate markets, investors, and financial systems for personal gains (Lin, 2023). On the other hand, disinformation is always intentional and aimed at misleading and deceiving people. It is crucial to detect the differences between the two, as the former can be better tackled and solved, but the latter involves a deeper and more comprehensive solution that make people aware of potential inaccurate and biased sources of information that systematically act unfairly. Indeed this kind of news mostly comes from media outlets which have a strong incentive to publish sensational and "scandalous" news to grab readers' attention and interaction leading to abnormal market reactions. According to a study

led by Gordon Pennycook, a psychology scholar at Cornell University, online financial disinformation and misinformation are way less effective on individuals with higher levels of financial literacy, suggesting that financial education alone may be helpful to make people less vulnerable to those as they would possess more knowledge and be more critical in assessing financial news veracity (Pennycook et al., 2023). Currently, financial dissemination and its motives are being explored and other novel rationales are coming up as markets' structures change and scammers find new ways to deceive investors. As a matter of fact, younger investors who are less experienced and acknowledged increasingly rely on social media for financial advice; they are also more likely to act on online misinformation and trust advice generated by artificial intelligence according to recent research produced by both Nationwide and Edelman Financial Engines. The US Nationwide survey found that 34% of non-retired investors aged 18 through 54 reported acting upon misleading or factually inaccurate financial information seen online or on social media. This includes more than 41% of Generation Z and 34% of Millennial investors. Older investors instead were more cautious about online financial advice with just 6% of Baby Boomer investors reporting they had acted on misinformation online, the least of any generational cohort (Lin, 2023). As investors rely more on social media news and information spread online, fake news continuously becomes a harder problem to tackle. The characteristics of fake news, such as its originality, recursive, and periodic nature contribute to its ever-growing dissemination and assimilation. This kind of news is usually designed to attract clicks, shares, and reactions, sacrificing the fundamental aspect of providing accurate and relevant insights to readers. Misleading ads targeting inexperienced people and often people in need, can be harmful too. Fraudulent investment schemes are some of the most common types of financial misinformation where scammers claim to secure unreasonably high returns to investors when instead they simply set "Ponzi Schemes" a pyramid scheme where early investors are paid off with money invested by new joiners. Financial fraud and scams are a concrete threat that pose incredible risks to the overall financial system and its integrity. These can indeed have effects not only on individuals but also on financial institutions, and this can lead to bankruptcy and even the failure of businesses. A significant example of online fraud due to financial misinformation is the "Wirecard Scandal." Wirecard, which was once recognized as a big player in the fintech sector, filed for insolvency in June 2020 after revealing that over €1.9 billion held in trustee accounts in the Philippines likely did not exist. The

scandal shook the financial world, underpinning severe failures in auditing and regulatory oversight. The company's rapid rise was acclaimed by claims of innovative financial services and partnerships, but reports, particularly by the Financial Times, began to unveil inconsistencies and fraudulent activities within Wirecard's accounting practices. The case led to the arrest of the bank's CEO Markus Braun and several executives accused of fraud and market manipulation. The fallout impacted businesses and investors worldwide, with billions of euros in market value wiped out. Online trading platforms have also contributed to the spread of misinformation as they host multiple fraudulent activities and malicious individuals who intend to disseminate false rumors, create market manipulation schemes, and provide incorrect investment advice. Social media like YouTube, TikTok, and Instagram offer a perfect platform for these individuals as their contents can be easily advertised and hence can be monetized by reaching unaware and naive investors. Stronger regulatory frameworks are needed to avoid this happening, as well as the active participation of social media providers in filtering and removing misleading and inaccurate financial information. Misleading Financial advertising and promotion is becoming the most worrying issue in financial disinformation, and deceptive tactics to let people buy questionable financial products are now considered the norm, especially on platforms like YouTube, where many "presumed" financial professionals promote their activities by promising people unrealistic yields and passive income. Ultimately, insider trading and market manipulation are likely the most common problems in the financial industry. It is clear that people having access to undisclosed and private information can potentially take advantage of that by exploiting non-public information for personal gain. This leads these individuals to generate generous profits, deteriorating market integrity and fairness in competition, distorting and manipulating markets, possibly by colluding with other market participants and/or spreading false information, creating false market supply and demand, and driving artificial market movements. Although many types of financial misinformation exist, this paper section touched on the most common ones, as our main objective is not to organically introduce all kinds of online financial misinformation but rather, mitigate and prevent their effects by studying state-of-the-art techniques.

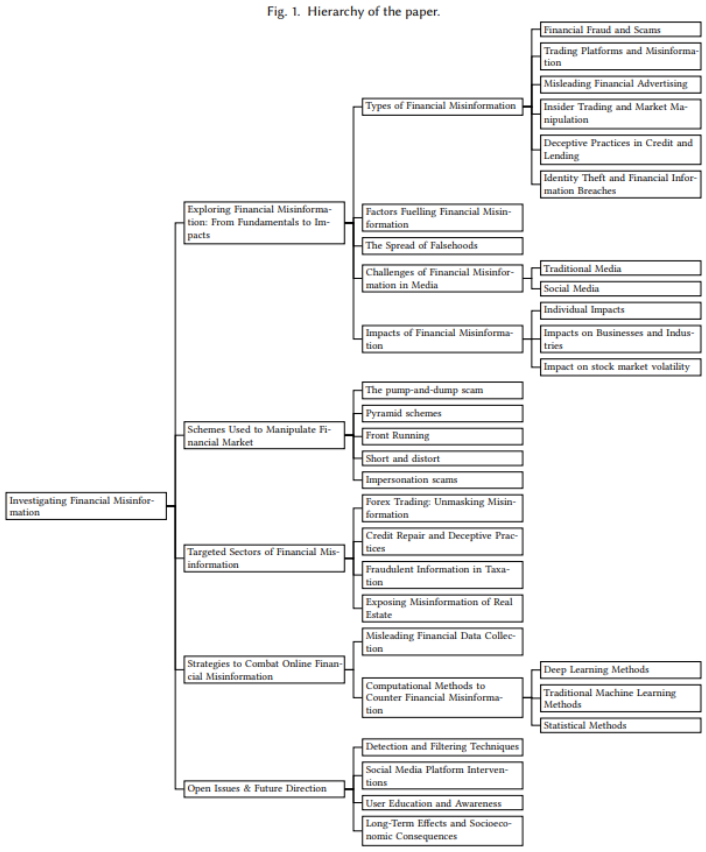

Figure 2.1: The figure above shows the main different types of financial misinformation (Rangapur et al., 2023).

2.2 The Economic Loss caused by Financial Misinformation

The rapid dissemination of information and news via digital platforms such as Forums and Traditional opinion-based social network like Twitter has transformed the way in which people access and read news. Although this revolution largely improved the speed and the width of information consumed, it also brought about the rise of misinformation and Disinformation. "Fake news" as it is commonly known, refers to the proliferation of online reporting that is poorly sourced or completely made up. This first-ever in-depth economic analysis of the economic impact of the problem says the price

tag to the global economy is $78 billion each year, with economic damage being inflicted on major sectors including politics, finance, advertising, online retail, and media. The Financial Information Industry is surely one of the most impacted sectors by this phenomenon. The same study reveals that Financial Fake News, which is defined as "the deliberate creation and sharing of false or manipulated market information that is intended to deceive the general public, financial investors, and finance professionals either to cause harm or for political, personal, or financial gain" has contributed a loss in stock market value amounting to $39 billion each year, which is half of the total loss due to fake news. This number gets even more significant as the study continued. Financial Misinformation alone, which refers to "False or inaccurate Financial information that is either deliberately or unintentionally spread" caused a loss of $19 Billion. (Cavazos, 2019) This underlines the rise of the spread of financial misinformation and disinformation as a major global risk impacting almost all sectors, as finance stands as the backbone of most businesses in the world. Examples of Financial Misinformation range from fraudulent investment schemes to misleading news articles or social media posts aimed at manipulating stock prices or influencing market sentiment. Such misinformation poses serious risks, including market volatility, investor losses, and erosion of trust in financial institutions. (Rangapur et al., 2023).

2.3 Current Fact-Checking Methods: Overview of Manual and Automated Approaches

In the task to maintain the integrity of financial information and tackle the spread of misinformation, Fact-checking stands as a pivotal process. Fact-checking is the task of assessing whether claims made in written or spoken language are true. The complexity of financial data, often miscellaneous in their nature, coupled with the potential consequences of misinformation underscores the importance of robust Fact-checking methods. This section provides an overview of both manual and automated approaches to fact-checking in the context of financial misinformation. This task is highly time-consuming as a journalist or publisher would likely need to go through several sources before validating a claim and this can take hours and even days given the amount of new information that appears and the speed with which it spreads; manual validation is insufficient. There are currently five main

types of fact-checkers in use—professional fact-checking organizations, mainstream news outlets, social media platforms, AI, and crowdsourcing with each having distinctive characteristics. Professional fact-checking organizations such as FactCheck.org and PolitiFact have experienced journalists and editors trained to spot inaccurate news and verify true claims by using statistical or scientific databases to fact-check information. PolitiFact fact-checks a wide range of political actors and groups including both elected and non-elected government officials, political candidates, media pundits, celebrities, and special interest groups. PolitiFact's uses a group approach for settling the veracity of a claim. The first step involves the lead writer submitting an article with a recommended rating to a panel of at least three editors according to Adair's "Principles of Politifact" article as well as researchers and reporters who have observed the PolitiFact evaluation process. The panel of editors evaluates the article and the author's recommended rating. The panel then discusses whether PolitiFact should follow the author's recommendation or assign a new rating. Instead, FactCheck.org targets "major U.S. political players in the form of TV ads, debates, speeches, interviews, and news releases. Like PolitiFact, FactCheck.org also fact checks social media claims and chain emails. Unlike PolitiFact though, it tends to avoid fact-checking media figures and pundits. (Ballotpedia, 2023). News Outlets like Jeff Bezos's Washington Post uses similar methodologies to fact-check news and information. Like its counterparts PolitiFact and FactCheck.org, the Post's Fact Checkers reach out to the individual or organization responsible for a claim and use raw data and original sources to examine it. Similarly to PolitiFact, the Post's Fact Checker uses a rating system where it assigns a rating based on how suspicious and unclaimed the claim is as well as if it was previously debunked. (Kessler, 2023). Manual fact-checking is praised for its thoroughness and the nuanced understanding humans bring to complex issues. However, it is time-consuming and challenging to scale, especially in the fast-paced world of financial news, and most importantly it is biased. There is also reason to think newer sources of fact-checks might be perceived as more credible than legacy sources. For example, fact-checking sources that use crowdsourcing may evoke the bandwagon heuristic, that is: "psychological phenomenon in which people do something primarily because other people are doing it, regardless of their own beliefs" (Wikipedia2023Bandwagon, 2023) which could lead users to assume verdicts rendered by such sources are objective because they involve consensus across many independent reviewers. Crowdsourced fact-checking systems may be less prone to charges of bias as

judges from different political outlooks contribute to the verdicts, but still present the main issue in professional Fact-Checking efficiency (Liu et al., 2023). To address the limitations of manual fact-checking and crowdsourcing, automated approaches leveraging advanced technologies like artificial intelligence (AI) and machine learning (ML) have been developed. These tools are less susceptible to bias as machines are not (yet) able to express and feel any kind of feelings that may erode its system reliability and consistency in fact-checking. Automated fact-checking represents a comprehensive approach aimed at addressing the challenges posed by misinformation in our social-media and online information-based society. Its core components which will be explained below include claim detection, evidence retrieval, and verdict prediction supplemented by justification production to explain the rationale behind the verdicts.

2.3.1 Claim Detection

The initial step in automated fact-checking involves identifying statements that need verification. This process often hinges on the concept of check-worthiness which evaluates the public's interest in the veracity of a claim. Techniques employed range from binary classification to importance-ranking of potential claims, aiming to emulate the prioritization practices of journalistic fact-checking under tight deadlines (Guo et al., 2023).

2.3.2 Evidence Retrieval

Following the identification of a claim, the next step involves locating credible sources that either support or reject the claim. This stage is crucial for assembling the factual groundwork upon which the veracity of the claim will be assessed (Guo et al., 2023). The complexity of this task varies significantly depending on the nature of the claim and the availability of authoritative sources. However, as of now, these systems can only identify simple declarative statements, missing implied claims or claims embedded in complex sentences which humans can recognize with ease. This is a particular challenge with conversational sources like discussion programs and chatbots where people might refer to previous points made to showcase a specific claim (Graves, 2018).

2.3.3 Verdict Prediction and Justification Production

The final step of the automated fact-checking process is the assessment of a claim's truthfulness based on the evidence gathered. This involves assigning truthfulness labels to claims (an example is FEVER, a publicly available dataset for fact extraction and verification against textual sources (Thorne et al., 2018)) and ideally generating justifications that explain the reasoning behind these verdicts. The goal is not only to determine the accuracy of a claim but also to provide transparent and understandable explanations for the verdicts reached. A basic form of justification is to show which pieces of evidence were used to reach a verdict. However, a justification must also explain how the retrieved evidence was used, explain any assumptions or common-sense facts employed, and show the reasoning process taken to reach the verdict (Guo et al., 2023).

The development and implementation of automated fact-checking systems have been propelled by advancements in natural language processing (NLP), machine learning, knowledge representation, and databases. These technologies enable the parsing of textual information, identification of relevant facts, and prediction of claims' veracity with increasing sophistication. Despite these advances, the automation of fact-checking faces inherent challenges, including the nuanced understanding of context, the synthesis of evidence from diverse sources, and the subjective nature of some claims' importance or check-worthiness.

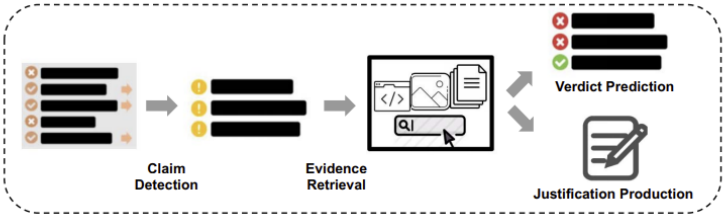

Figure 2.2: This workflow shows the typical journalism fact-checking process (Guo et al., 2021).

2.4 Large Language Models (LLMs): Evolution, Capabilities, and Limitations

Large Language Models (LLMs) have revolutionized the way people access information. LLMs are the ultimate result of the extensive research and experimentation in Natural Language Processing (NLP), a subset of Artificial Intelligence and linguistics which has the primary objective of making computers understand the statements or words written in human languages. It involves the creation of phrases, sentences, and paragraphs that entail precise and meaningful content (Khurana et al., 2019). Before diving into what LLMs specifically address, an introduction to some NLP pillars is necessary, hence the concepts of tokenization, word embeddings, positional encoding, encoders, and decoders will be discussed. In natural language processing, understanding human language requires breaking down text into manageable, and interpretable units for a computer. This fundamental unit of text is the token. Tokens are simply the basic units of data processed by LLMs and in the context of text, they might be a word, a subword, or even a complete sentence/paragraph that entails specific meaning (Naveed et al., 2023). The process of converting raw text into a sequence of tokens is called tokenization. For example, Subword tokenization as used in models like BERT and GPT splits words into smaller units to handle out-of-vocabulary words and rare words more effectively. Once the text has been tokenized, the tokens are typically converted into numerical representations called word embeddings since computers cannot understand words but rather they handle numbers extremely well. Word embeddings are dense, low-dimensional vector representations of words that capture semantic and syntactic relationships between words. These embeddings allow models to understand similarity, analogy, and the rich semantic relationships between words, facilitating tasks ranging from text classification to sentiment analysis.

Given that Transformers process sequences in parallel unlike their predecessors (RNNs and LSTMs) which process sequentially, they lack the inherent ability to recognize the order of tokens. Positional encodings are used to address this issue; they are units added to the embeddings to give models information about each token's position in the sequence. These encodings are designed with patterns that the model can learn to demystify and interpret, making it understand the word order, a paramount aspect for grasping the meaning in sentences that might have different word ordering. The last

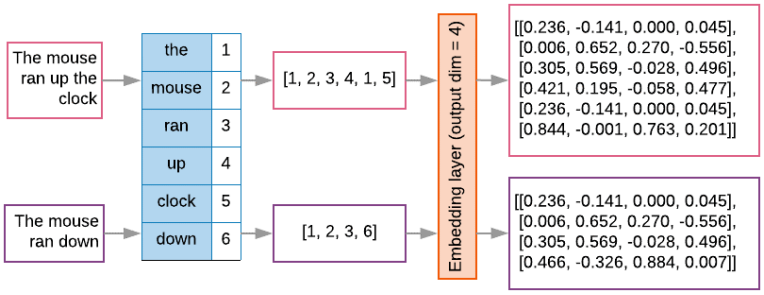

Figure 2.3: Visual representation of how tokenization and embedding work (Rathinapandi, 2023)

two concepts to be introduced are encoders and decoders, two key components of transformers introduced in the Transformers' original paper which will be discussed later. Encoders are the parts of Neural networks that process each token into a rich and context-aware representation. Each encoder layer uses a self-attention mechanism (which will be explained later) that allows each token to interact with all its connected tokens in the sequence, assessing their relevance to one another; that is, how useful the other tokens are to understand a specific token meaning. It could be imagined as a librarian ordering books by their style, themes, and relevance, creating a very detailed map (hidden states) of its library's collection. The hidden states represent the depth and degree of connection of all the books at multiple layers, from the more superficial relations to the most intricate and detailed. As a result, the output from the encoder encapsulates not just the individual token information, but their contextual relationships within the sequence. Decoders, while mirroring the encoder structure, are designed to generate the output sequence token by token. Each decoder layer starts with a self-attention mechanism, enabling it to focus on relevant parts of the sequence it has generated so far (considering the order of generation). An important addition is the encoder-decoder attention layer that allows the decoder to focus on relevant parts of the input sequence based on the context provided by the encoder's output. This intricate process between focusing on its own output and drawing from the input sequence's context allows the decoder to generate relevant and coherent text sequences that will be the query result of human inquiries in LLMs.

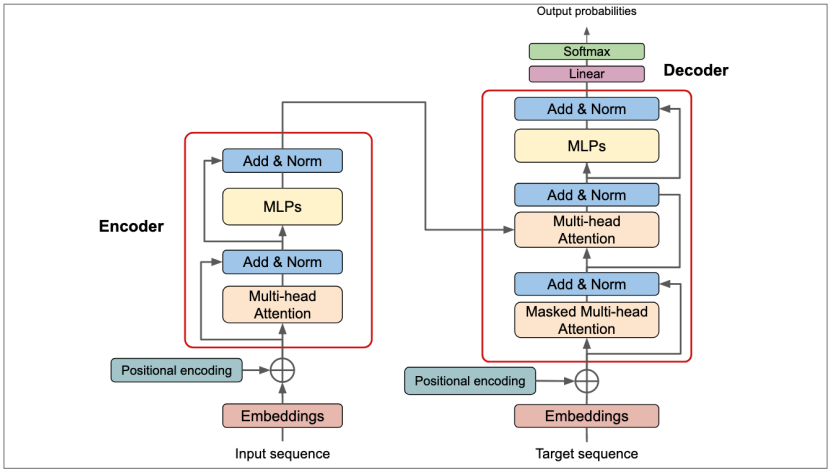

Figure 2.4: Simple Visual Representation of how the Encoder and Decoders Blocks interact (Nyandwi, 2023)

LLMs represent a subcategory of deep neural networks which are based on the Transformer Architecture introduced by University of Toronto's scholars and Google Researchers in the famous paper "Attention is all you need". This work opened the development of state-of-the-art Natural Language Processing Architectures and technologies such as Generative Pre-Trained Transformers (GPTs) and Bidirectional Encoder Representations from Transformers (BERT) with the latter being the first noticeable improvement over previous state-of-the-art models. On a high level, LLMs are language models that provide general–purpose language generation by acquiring information in text documents by detecting statistical relationships. Most LLMs have the function of generating text, a form of generative AI, by taking an input text and repeatedly predicting the next token or word while others can translate text or perform various other language-related tasks. To understand their capabilities and limitations, it is worth introducing the aforementioned architecture that enabled GPT models like GPT-4, Llama2, and Claude 3 to shine. The transformer architecture came as a disrupting discovery in NLP, revolutionizing the way LLMs and other Language-related models are built and designed. Previous state-of-the-art architectures like Recurrent neural networks (RNNs), long short-term memory (LSTMs), and gated recurrent neural networks (GNNs) faced significant issues due to their sequential processing nature and their recurrence. This design approach led

to big challenges in handling long-range dependencies within text (long texts that comprehend multiple context windows) as the ability of these models to retain information and context from earlier tokens diminishes as the distance between tokens increases. Now the concept of the Self-Attention mechanism comes. This novel introduction presented in the aforementioned paper enables the model to dynamically allocate focus across different parts of the input sequence, determining the importance of each token in relation to others for a given task. This allows the transformer to model the dependencies between tokens effectively, regardless of their distance from each other in the text. Moreover, it needs fewer parameters to model long-term dependencies since it only must pay attention to the inputs that matter and it's remarkably good at handling inputs if different lengths since it can adjust its attention based on the sequence length. The self-attention mechanism allows the transformer model to process the entire input sequence in parallel and dynamically weigh the importance of different parts of the input when computing the representation of a particular token. This is achieved through the following key steps:

- **Queries, Keys, and Values:** The input sequence is then converted to the following three matrices: queries (Q), keys (K), and values (V). The matrices of the queries correspond to the present token, the keys' ones to all the tokens in the sequence, and the values to what information needs to be passed along for each token.

- **Attention Scores:** For each token, the self-attention mechanism will compute the scores by taking the dot product of the query with each key and then by scaling the result by the square root of the dimension of the key. It generates a vector of one attention score for each token in the sequence.

- **Attention Weights:** Now, a SoftMax function is applied to the just generated attention scores to map their values between 0 and 1.

- **Weighted Sum:** The model will now compute a weighted sum of the values based on attention weights, where the weights tell how much each value's contribution to the final representation of the current token is. And the fact that the model can process all the tokens simultaneously, in combination with self-attention, dramatically boosts the model's ability to comprehend and generate natural language. (Raschka, 2023)

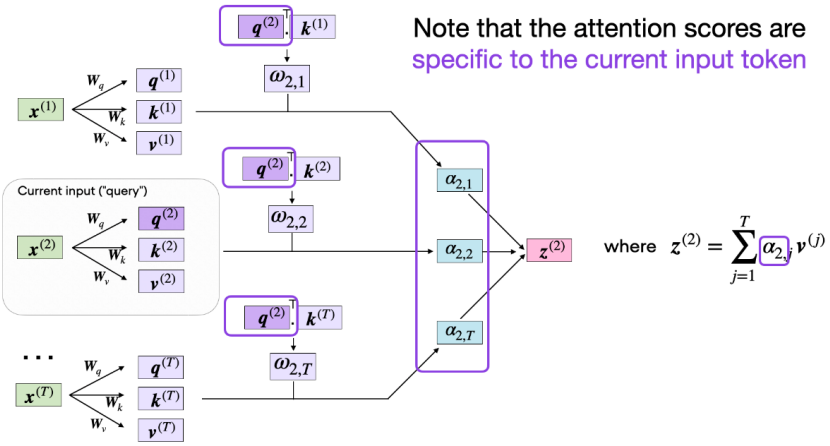

Figure 2.5: How the weighted sums are computed (Raschka, 2023)

Building on the foundation of self-attention, the Transformer architecture uses multi-head attention to enhance its ability to capture multiple relationships between tokens. Instead of performing a single set of Q, K, V transformations, the model does this multiple times with different learned linear transformations. This process results in multiple sets of attention scores and outputs for each token, each representing different aspects of the token's relationships within the sequence. This process, known as "Multi-Head attention", presents two main advantages:

- **Richer Representation:** By allowing the model to attend to information from different representation subspaces at different positions, multi-head attention provides a more detailed understanding of the sequence.
- **Flexibility:** This mechanism enables the model to capture a variety of linguistic features, from syntax to semantics, within the same layer, enhancing its expressiveness and adaptability (Vaswani et al., 2017).

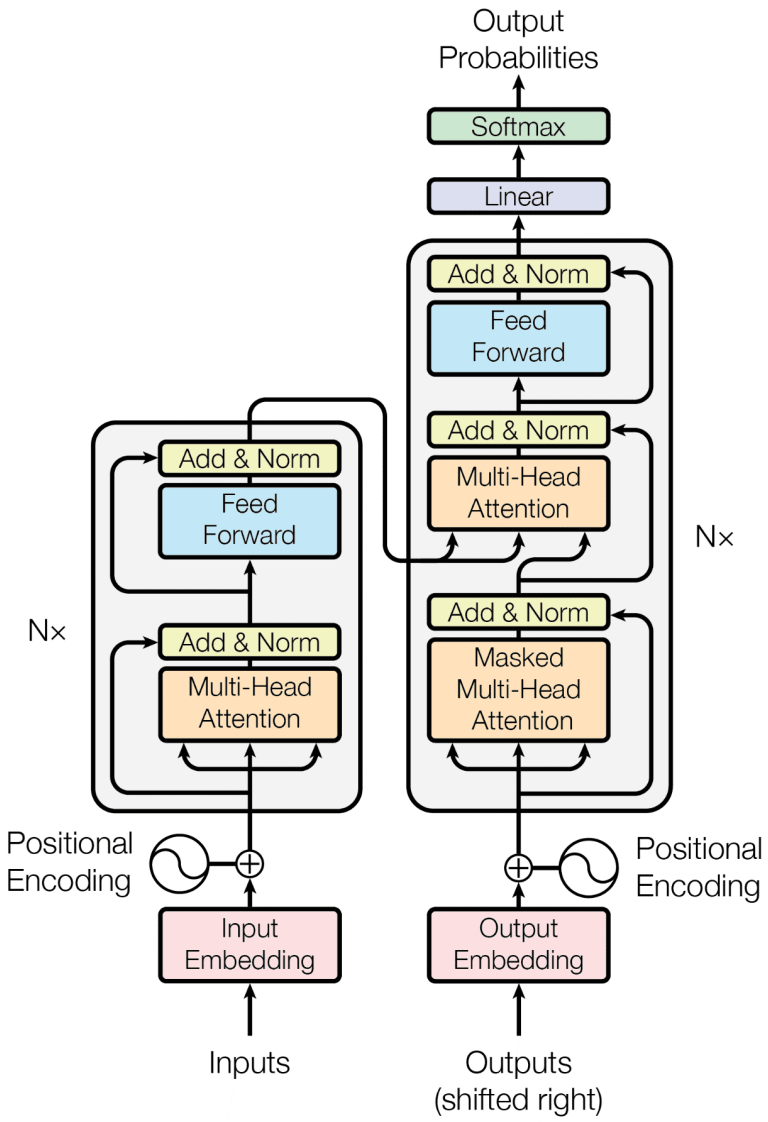

Figure 2.6: The original Transformer-Model architecture (Vaswani et al., 2017)

2.5 Retrieval Augmented Generation (RAG): Introduction, How it Works, and its Advantages

As LLMs evolved, new contributions to their development arose. Although generally high performing on general knowledge queries, these chatbots suffer when dealing with unseen data; for example, OpenAI's best-performing model GPT-4 completed its training in April 2023 with its training data cut off until then. This leads to unpleasant and emerging issues in GPTs' prompts' answers as those provide misleading and incorrect responses to users' prompts since they might lack up-to-date and necessary information to provide users with informed answers. Language models have come a long way, but they can still sometimes generate incorrect information (a phenomenon known as hallucination, which was introduced in the abstract), leading to concerns about their reliability as they might fall into grammatically correct but factually wrong answers.

Figure 2.7: GPT-4 hallucinates. Indeed, it gives an answer that is factually wrong and not up-to-date (8 Apr 2024) (ChatGPT, 2024).

As one may argue, specificity in prompts drastically reduces the chances of hallucinations, and it is important to provide detailed and accurate instructions to LLMs to enhance their performance and reliability. However, context and specificity are not always needed; in the example above, the current market cap of Tesla is asked and ChatGPT-4 answers incorrectly. In this instance, specifying the current market cap is enough to give the model context (the request is simple and unambiguous) and nonetheless, the LLM

failed to respond factually and accurately without disclaiming an eventual mistake that it may make as it does not have access to real-time data and hence is unable to answer the question. There are a set of techniques that may be deployed to tackle hallucinations such as fine-tuning, reinforcement learning, and Retrieval-Augmented Generation. Fine-tuning is the process of adjusting a pre-trained model on a specific, often narrower dataset or task to enhance its performance in that domain. This includes both supervised and unsupervised FT with the first shown to be very good at improving the quality of models especially on its zero-shot and reasoning capabilities, but it does not solve the lack of relevant information that the model did not possess in the training data as it only gives specific instructions to the model rather than injecting new information. The same applies to reinforcement learning (RL), another form of FT which does not improve the model information breadth. Unsupervised Fine-Tuning instead provides a partial solution to this problem as in this method, the FT process is a direct continuation of the pre-training phase. It starts with a checkpoint of the original LLM and then it is trained in a causal autoregressive manner, i.e., predicting the next token. One major difference in comparison to actual pre-training is the learning rate, a fundamental hyperparameter in neural networks. This method enables the model to continue injecting knowledge as it gives the model the capacity to learning new information (Ovadia et al., 2023). Ultimately, the best-performing and transformative technique for fact-checking and domain-knowledge queries is RAG. The underlying idea of RAG is straightforward; Retrieval Augmented Generation (RAG) is a technique that expands the capabilities of large language models (LLMs) by combining them with a retrieval system. The key idea behind RAG is to augment the LLM with additional knowledge from an external source, such as a knowledge base or a document collection, to enhance its performance on tasks that require factual information or reasoning. Indeed, LLMs can be inconsistent; sometimes they provide accurate answers to questions whilst at other times they make up random facts from their training data. RAG ensures that the model has access to the most current, reliable facts and that users have access to the model's sources ensuring that its claims can be checked for accuracy and ultimately trusted (IBM, 2023). RAG has proven to be working much better than fine-tuning alone, especially for knowledge-intensive tasks where knowledge retrieval and checking is key – RAG outperforms fine-tuning by a large margin as in most cases it adds knowledge to the model and does not affect catastrophic forgetting (knowledge that models forget after some

time) whilst fine-tuning proved to even lower these models' performance at times (Ovadia et al., 2023). Before deep diving into the RAG architecture, some key concepts will be demystified to better comprehend how RAG systems work. The first concept to be introduced is Vector Databases which are key in RAG systems. A vector Database is generally used to store high-dimensional data which can't be stored in DBMS. It is a type of database that stores data as high-dimensional vectors, with each one having a certain number of dimensions. Those are generated by applying embeddings or other transformations. They generally shine over alternatives as they ensure fast and accurate similarity search and retrieval, which are 2 fundamental aspects in RAG as it looks for the documents and the contents that are most similar with the user' query providing the flexibility that DBMS do not offer (Indeed DBMS databases are generally limited as one cannot capture semantic and intrinsic contextual meaning) whilst vector databases can easily reveal correlations in highly dimensional spaces. Another aspect that traditional DBMS lacks is the support for complex and unstructured data which is granular and complex (audio, video, and images are some examples). Whilst vector databases can capture their characteristics by transforming this data into vectors. Finally, another advantage is general scalability and performance as these vectors can also handle real-time data and can use sharding, caching, and replication to optimize resource allocation and exploit parallel computing advantages (Han et al., 2023). The question now comes naturally; after storing the data, how can one retrieve it? Dense Retrieval is another paramount concept to grab in RAG systems. In a nutshell, suppose we have a query q and a text collection D composed of d_i components where i is the i-th component. Then, given a query q, text retrieval returns a ranked list of n most relevant texts $L = [d_1, d_2, d_3, \ldots, d_n]$ according to the relevance scores of a retrieval model. It is important to note that both Sparse Retrieval models and Dense ones can be used for this task. However, Dense Retrieval has an edge as it can model and detect the semantic interaction between queries and texts using large-scale data whilst Sparse retrieval relies on keyword matching and can miss semantically relevant documents that don't share keywords with the query. PLMs such as BERT and its variants are at the core of dense retrieval systems. They are used to encode queries and documents into dense vector spaces, facilitating semantic matching beyond simple lexical overlap. The "Transformer" architecture is a foundational element for PLMs, emphasizing its role in handling sequence data efficiently and its adaptability for massive parallelization. Once the data is transformed into these dense

representations, the retrieval process begins. The retrieval process in dense retrieval systems is characterized by mapping both queries and documents into high-dimensional continuous vector spaces. These dense representations capture the semantic essence of the texts, enabling the retrieval of relevant documents even in the absence of exact keyword matches. There are 2 fundamental aspects and processes in the Dense Retrieval Process – the first is the need to train this model (Retrieval Model Training). Utilizing labelled datasets, retrieval models are trained to fine-tune the PLMs. This process involves adjusting the model to improve its ability to semantically match queries with relevant documents. Once this is done, data is indexed to ensure efficient retrieval; Dense vectors for documents are indexed using specialized data structures that support efficient similarity search such as FAISS. This setup allows for the rapid retrieval of top-ranking documents from a potentially vast corpus (Zhao et al., 2022). Similarity search is a method used to find documents or articles that have similar content to a given query. This is achieved after vectorization, that is transforming text (or whatever our data content is, may as well be videos, images, and other unstructured data types) into vectors. Once vectorization is done, now the vectors are represented in highly dimensional vector spaces and their distance/similarity can be measured using multiple techniques. Unlike traditional search methods such as keyword-based search, similarity search does not rely on explicit queries or exact matches. Instead, it leverages mathematical and statistical techniques to determine the degree of similarity between objects (Paltiel, 2023). One of the most used similarity measures used in RAG architectures is Cosine Similarity. CS simply measures the cosine of the angle between two vectors in a multi-dimensional space. It ranges from -1 to 1, where 1 indicates identical direction and hence similarity and -1 meaning they are orthogonal (opposite direction). This measure is quite ideal for text retrieval as it captures the semantic similarity between documents irrespective of their size and magnitude. Another important measure is L2 Distance (also known as Euclidean Distance) that is widely used in FAISS where the magnitude of the vectors plays a significant role in their similarity. It is particularly effective for clustering and nearest neighbour searches in spaces where the Euclidean norm reflects the inherent structure of the data. In FAISS, the L2 distance is beneficial for exact and approximate nearest neighbour searches. Instead, for scenarios where the angle between vectors is more important than their magnitude (common in text and image retrieval tasks), the inner product serves as an alternative measure to Cosine Similarity. When vectors are normalized,

the inner product is equivalent to measuring the cosine similarity making it highly suitable for identifying semantically similar items in a vector space. FAISS optimizes the inner product search to accommodate high-dimensional data ensuring efficient retrieval even in very large datasets. These are the most widely used measures used in Similarity search and others have been omitted; however, the goal was to present some of these measures and how they work. The last pillar concept to introduce is the role and importance of Pre-Trained Language models (PLMs). For the sake of clarity and conciseness, a brief introduction to these will be given as they would deserve a thesis alone to be properly explained. PLMs are essentially deep learning models that have been trained on vast amounts of text data. This training process enables them to understand language patterns, syntax, semantics, and even some aspects of common sense and world knowledge. Models like BERT that we previously identified as a foundational element for NLP tasks are trained to predict missing words in a sentence, helping them grasp context and language structure while GPT models are trained to predict the next word in a sequence enhancing their generative capabilities (Devlin et al., 2018). PLMs have different functions in RAG in particular:

- **Encoding Queries and Documents:** PLMs encode queries and documents into high-dimensional vector spaces. This encoding transforms the textual information into a format that machines can understand, preserving semantic information such as the context, meaning, and nuances of language. These dense representations allow RAG systems to perform similarity searches between queries and a vast collection of documents, identifying the most relevant content for retrieval.

- **Dense Retrieval:** Dense-Retrieval capabilities of RAG can also refer to their use with PLMs in Dense Retrieval, an encoding process where a user query is encoded with its vector form, and then this vector representation is compared to the vector representation of documents in the database. In a way, RAG identifies and then gets to bring out semantically related information, even if there is no exact keyword match. This process is vital considering sourcing content that is relevant to be used while generating.

- **Generation:** Once the relevant set of documents is retrieved, the second PLM, which is usually the optionally more advanced GPT, is employed by RAG for generating responses. First of all, the model takes

the original query and the retrieved documents and synthesizes them into coherent, relevant, and contextually appropriate responses. The generative PLM, based on its training, might produce text not only in service of answering the query but also done that with stylistic and thematic consistency with the input data.

- **Fine-Tuning:** Pre-trained capabilities of PLMs can be fine-tuned based on explicit tasks or domains, allowing RAG systems to be attuned to specific contexts or content types. Fine-tuning enhances the model's ability of the model to gain responses more specific and relevant to the user's needs.

This combination offers a deep and context-aware approach to delivering highly relevant answers to a wide array of queries. PLMs allow RAG systems to be adapted to many domains, be it legal text, medical literature, or financial data, according to the case, making them more accurate and relevant in certain scopes.

2.6 The RAG Architecture: a General Overview

Now that we introduced the fundamental components of RAG, it is time to explain how this architecture came to life. Retrieval Augmented Generation (RAG) was introduced in 2020 by some Facebook, UCL, and NYU researchers to improve fact-verification and up-to-date knowledge for intensive NLP tasks in LLMs. These were found to be particularly struggling in providing new information as their pre-trained nature limited their knowledge until the cut-off date as we anticipated earlier in multiple instances. The researchers effectively combined the capabilities of LLMs like BART and T5 (2 of the earliest Pre-Trained seq2seq models based on the transformer architecture) with an external retrieval system. This synthesis aims to mitigate the limitations of LLMs, particularly their propensity to "hallucinate" or produce factually incorrect information due to their static knowledge base which is limited to the information they were trained on as we said in the thesis' introduction. In a nutshell, the original RAG architecture comprises 2 main components: a retriever and a generator.

- **Retriever:** The retriever component is responsible for the retrieval of relevant documents or passages from an external knowledge source

(non-parametric hence not intrinsic of LLMs) in response to a given user's query on LLMs. It uses techniques such as dense retrieval to match the query with relevant content in the knowledge base. The crucial innovation brought by the paper's researchers is the use of a transformer-based model to encode both the user's query and the documents in the knowledge base into high-dimensional vector spaces (these concepts were already introduced before). Techniques like vector databases and dense retrieval models are employed to efficiently search and retrieve information from large collections of documents using PLMs to generate embeddings that capture semantic meaning. The retriever's objective is to provide the generator component with a set of contextually relevant information that can be used to generate accurate and informative responses.

- **Generator:** As the desired content has been fetched, the Generator now comes into play. This component synthetizes the retrieved content into an intuitive and contextually relevant content for the LLM to process. As mentioned earlier, famous generators are mostly sequence-to-sequence models such as BART or T5, which use both the queries' and retrieved contents' embeddings as inputs; These models process this input to produce a natural language output that is not only relevant but also consistent with the style and thematic elements of the input data. This is achieved by leveraging the model's capabilities to understand and generate human-like text based on the training it received on a diverse corpus during the pre-training phase (Lewis et al., 2020).

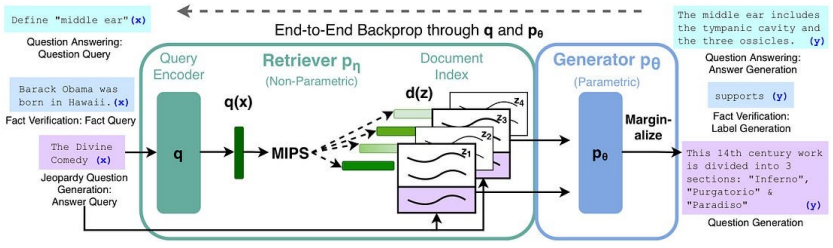

Figure 2.8: The original RAG architecture introduced in the aforementioned paper (Lewis et al., 2020)

In their experiment, they tested this system with a Wikipedia dense vector (non-parametric memory) to evaluate the system's performance – in particular, they extracted from the dense vector x the z relevant text documents to produce the output y from the generator.

The retriever collects the top k relevant documents given a query q while the generator returns the probability of similarity of each document to q mathematically:

$$\begin{aligned} \mathrm{P}_{\text{RAG-Sequence}}(y|x) &\approx \sum_{z \in \text{top-k}(p_\eta(.|x))} p_\eta(z|x) p_\theta(y|x,z) \\ &= \sum_{z \in \text{top-k}(p_\eta(.|x))} p_\eta(z|x) \prod_{i}^{N} p_\theta(y_i|x,z,y_{1:i-1}) \end{aligned}$$

The RAG-token model instead allows the generator to choose from multiple documents when it is generating the output y producing a distribution for the next output token for each document before marginalizing and repeating the process with the following output token. Formally:

$$\mathrm{P}_{\text{RAG-Token}}(y|x) \approx \prod_{i}^{N} \sum_{z \in \text{top-k}(p_\eta(.|x))} p_\eta(z|x) p_\theta(y_i|x,z,y_{1:i-1})$$

The experiments touched Open-domain question answering, jeopardy question generation, and fact verification which represents the critical matter for this thesis' work. As the first and the last concept come as no-brainer, jeopardy question generation may seem new to most. This is simply the task of guessing an entity from a statement or description – an example might be: "there's a city which is known to be the cradle of civilization and it hosts millions of visitors every year" – in this case, RAG will be tasked to guess Rome as a candidate and to produce the response without knowing the response but identifying it based on the context. All the experiments have demonstrated that RAG performed extremely well, showing that it hallucinates less in Question Answering, was much more factual than BART in Jeopardy Question Generation with BARD being factual only on 7.1% of cases vs the impressive 42.7% of the RAG system. In Fact-verification benchmarks showed that RAG has very similar performance to complex and tailored state-of-the-art solutions – the results are impressive considering that

it was only supplied with the claims and not with the True or False label retrieving its own evidence. This paper unlocked the potential of RAG as it presents two major advantages over standard LLMs systems, especially for Fact-verification: Up-to-date information and high reliability. At this stage, you may wonder how the system can retrieve the documents and texts' information – how does it make it happen? The answer lies in two processes: Indexing and Chunking. Indexing is the process of acquiring data from different sources and building an index to enhance retrieval speed and efficiency so that data is easily searchable from a large dataset. The retriever usually uses indexes to identify the content that matches the user query and it's easy to see why this process is fundamental for RAG. This is done by cleaning and converting formats such as pdf, html, Markdown, etc... into plain text. Chunking then involves dividing the text into smaller chunks of corpus. This is crucial as language models typically have a limited context window and hence, a limit on the context they can handle (Gao et al., 2023). As this section explored the original RAG architecture, it is important to say that there exist many frameworks that greatly improved its performance. For instance, Advanced RAG has been developed to address shortcomings of Naive RAG such as Retrieval quality (In Naive RAG the content does not match well with the user query) and quality of response generation where irrelevance is the main bottleneck. Advanced RAG optimized Data Indexing by enhancing data granularity and Index Structures, both aimed at enlarging and perfecting contextual information in the pre-retrieval process. After embedding, Post-retrieval strategies are applied once the context is identified. Additional processing includes reranking the relevant documents, compressing prompts to reduce noise and the overall context length. Advanced techniques like recursive retrieval are used to tackle some of Naïve RAG issues - it involves capturing key semantic meanings before the embedding from smaller chunks and give Language models larger blocks with more context – this helps strike the balance between efficiency and contextually rich responses.

2.7 Retrieval Augmented Generation Main Advantages and Shortcomings

RAG allows users to verify the trustworthiness of information, providing a layer of transparency and reducing the critical issue of hallucinations. RAG

systems show to be particularly useful when specialized in fixed domains – this may include law and legal domains where regulations and laws are often difficult to find and interpret and general LLMs might lack their detailed content; The financial sector is another one: with the proliferation of fake news that was described before as extremely harmful for the global economy, RAG provides an easy way to improve LLMs performance and deal with financial data. This data is particularly ideal for RAG systems as most of it is usually structured and easy to index, structure and interrogate, especially when dealing with financial news. Although these systems mostly present advantages to count on, there are obviously some drawbacks impacting its broad adoption in different contexts. Long context is still a major issue in this architecture, as LLMs are heavily constrained by their context windows; if this is too short, the model won't capture enough information and with too much context, it might lose information when navigating in long and semantic-rich windows (Zetterlund, 2023). Robustness is another issue strictly linked to retrieval, as irrelevant noise may appear and the retriever might capture irrelevant or even misinformation instead of carefully selecting appropriate content. After all, no information is better than misinformation and these models must improve on this. Lastly, another major issue is when RAG tackles multi-reasoning questions that require complex reasoning and inference across multiple sources, resulting in incomplete or incorrect answers produced by the generator (Zetterlund, T.).

2.8 RAG Variations and a Gentle Introduction to Agentic RAG

As we outlined the main drawbacks of Traditional RAG approaches, we now introduce some proposed alternatives that aim at tackling what was not properly addressed. In the last chapter, we briefly discussed the potential advantages of Advanced RAG techniques and how they could help in the Indexing, Retrieval, and Generative parts of RAG. However, an introduction to state-of-the-art RAG techniques must be carried out to make this research relevant and enhance the capabilities of our system focused on limiting LLM hallucinations in finance-related Q&A tasks.

2.8.1 Self-RAG

The first architecture is Self-RAG, introduced in 2023 by IBM and Washington University scholars. Self-Reflective Retrieval-augmented Generation improves LLM's generation ability to provide factual accuracy; it does that by "reflecting" on generated reflection tokens which are intermediary tokens which help the system decide whether retrieval is necessary and to gauge the quality of retrieved passages (called the self-reflection step) so that it can be augmented if useful for generation. Once this is done, it processes these multiple passages assessing their relevance and generating corresponding task outputs. This approach differs from conventional RAG approaches as it does not consistently retrieve a fixed number of documents/content but rather it critically decides what to retrieve. Ultimately, the system trains a LLM using unified reflection token that will serve as the next prediction token. In particular, during the inference step, for every user input x and preceding token y, the model creates a retrieval token to assess the utility of retrieval; if that is not useful, it proceeds with the next output segment, otherwise, the model generates a token to evaluate the retrieved content, the next response segment, and a token that acts as a critic for the response to check if its information is evidenced by the passage. Finally, a last token evaluates the overall utility of the response (Asai et al., 2023).

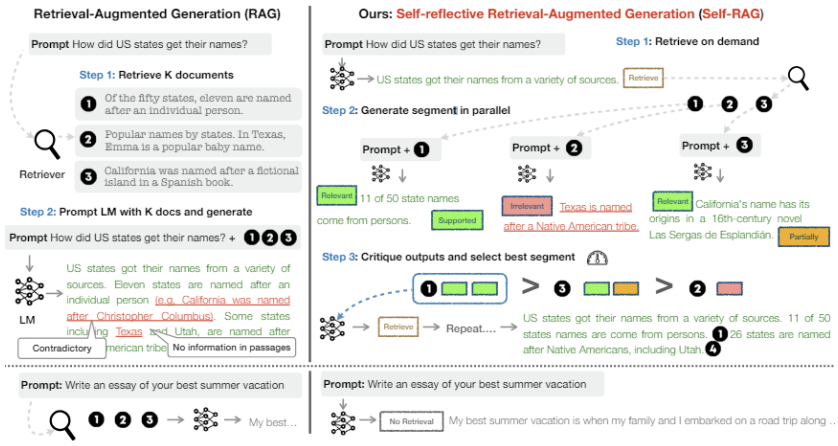

Figure 2.9: Comparison of Naive RAG and Self-RAG architecture (Asai et al., 2023)

2.8.2 Adaptive-RAG

Adaptive-RAG instead is another advanced RAG model which performs particularly well in Question Answering (QA) tasks in LLMs. The architecture was recently discussed in the paper "Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Question Complexity". As the title suggests, this technique involves differentiating between the complexity of user requests and adapting accordingly. The authors claim that some queries require connecting and aggregating multiple documents which are often not answerable with a traditional single-step process of retrieval and response. For this reason, the researchers have created an adaptive framework to pre-define the query complexity using a classifier proposing a method that offers a concrete compromise between iterative LLM Augmentation and single-step methods for simple queries and even no retrieval at all – it does it in the adaptive retrieval a process where a small language model (the Classifier in this case a T5 PLM) trained to classify the complexity of queries assigns labels A, B, or C given q, where A indicates the query q is simple and straightforward to answer, B as moderate complexity when queries might need some iterative retrieval but single-step responses may work fine, and C as complex queries requiring more complex solutions. Based on metrics such as F1 (number of overlapping words between the predicted answer and the ground truth), accuracy (predicted answer contains the ground-truth answer) and response time, Adaptive-RAG overperforms Self-Rag on both Single-hop QA (single search to generate an answer) and Multi-hop QA (Generating an answer based on multiple searches).(Liu et al., 2024).

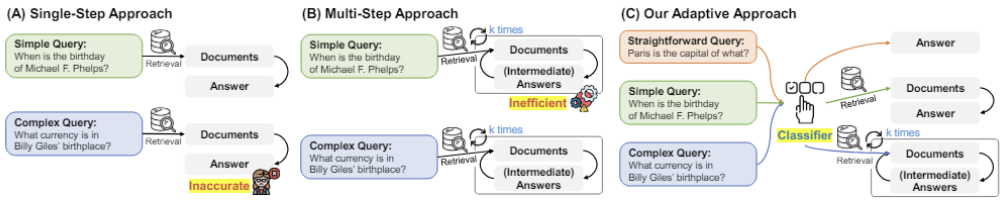

Figure 2.10: The Adaptive-RAG framework differentiating query complexities (Liu et al., 2024)

2.8.3 Corrective-RAG

Corrective-RAG (also known as CRAG) is another innovative architecture that aims to address the inherent shortcomings in the retrieval-augmented generation process. This framework was introduced by researchers focusing on the iterative correction of the generated outputs to improve factual accuracy and coherence. The main idea be Corrective RAG revolves around a feedback loop mechanism where the initial response generated by the model is evaluated and refined through subsequent retrieval and generation cycles. This approach ensures that the final output is more reliable and accurate by iteratively correcting any inaccuracies or inconsistencies present in the initial response.

In its substantial form, it possesses 4 important processes and paradigms that make it so effective:

- **Retrieval Evaluator:** At this initial stage, the module evaluates retrieved documents, scoring their reliability and usefulness. Documents are classified into categories such as Correct, Incorrect, or Ambiguous based on these scores.

- **Corrective Actions:** Based on the confidence scores given by the Retrieval Evaluator, CRAG takes different corrective actions, which are the following: Correct: The document is deemed reliable and usable as is; Incorrect: The document is rejected or replaced; Ambiguous: The document is flagged for further refinement or additional retrieval.

- **Knowledge Refinement:** This process involves breaking down documents into essential information while filtering out irrelevant or incorrect content. The goal is to retain only the most accurate and relevant data.

- **Web Search Integration:** CRAG systems usually integrate large-scale web searches to enhance the amount of useful information, especially when dealing with incomplete or inaccurate corpora. This helps in expanding the knowledge base with more reliable sources.

Essentially, the system aims to ensure the accuracy and relevance of the augmented retrieval data, adapting to multiple varieties of RAG architectures, given its robustness and flexibility in integrating it into already existing workflows (Yan et al., 2024).

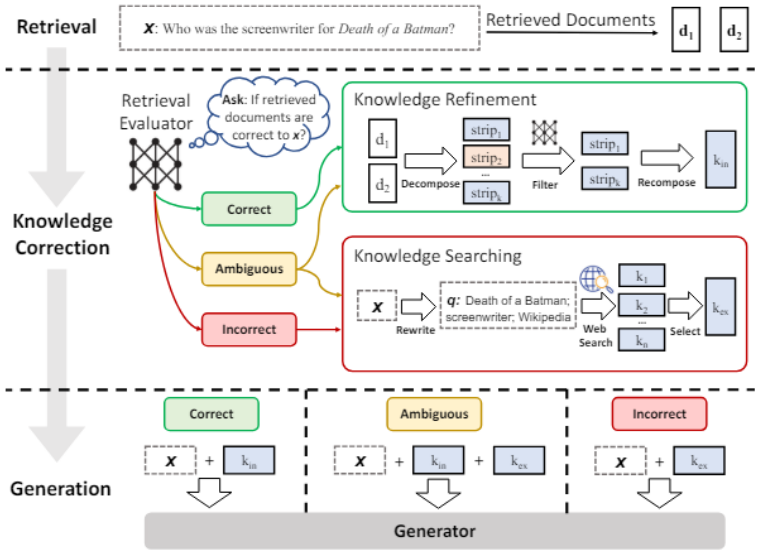

Figure 2.11: Corrective RAG architecture with feedback loop mechanism (Yan et al., 2024)

2.8.4 AI Agents

Although these 3 techniques seem to be extremely promising, our work will be using another approach which is becoming popular not only for Fact-Checking and LLM tasks, but also for wider Generative AI and Automation tasks. Before introducing what this method is about, we must introduce the concept of AI agents. In a nutshell, AI agents are autonomous systems performing specific tasks with no human intervention. They can navigate various environments, making real-time decisions based on dynamic real-time information in a totally independent way. These agents may resemble what standard automation is, but it's more; they can potentially adapt in unknown environments with novel data, and finally, they're able to run computer programs and perform various tasks in it - From browsing the internet

and managing apps to conducting financial transactions and controlling devices, their capabilities are vast and versatile. They work by engaging with LLMs that operate in background and tell what tasks to execute and showcase its understanding of the task they are doing; these make agents plan and utilize external resources using a vast knowledge base (Durante et al., 2024).

Their potential applications range from the physical world (cameras, multi-modal sensors, speech, video, IOT etc.) to the Virtual world (Big Data, LLMs, planning, Inference, reasoning etc.) to more niche tasks such as Robotics Controller and Manufacturing activities. Many researchers claim that these agents may eventually lead to AGI (Artificial General Intelligence) which basically consists of a non-human surpassing humans in most operational and cognitive tasks, which is what companies like OpenAI, DeepMind and Anthropic are trying to achieve (Wikipedia AGI, 2024).

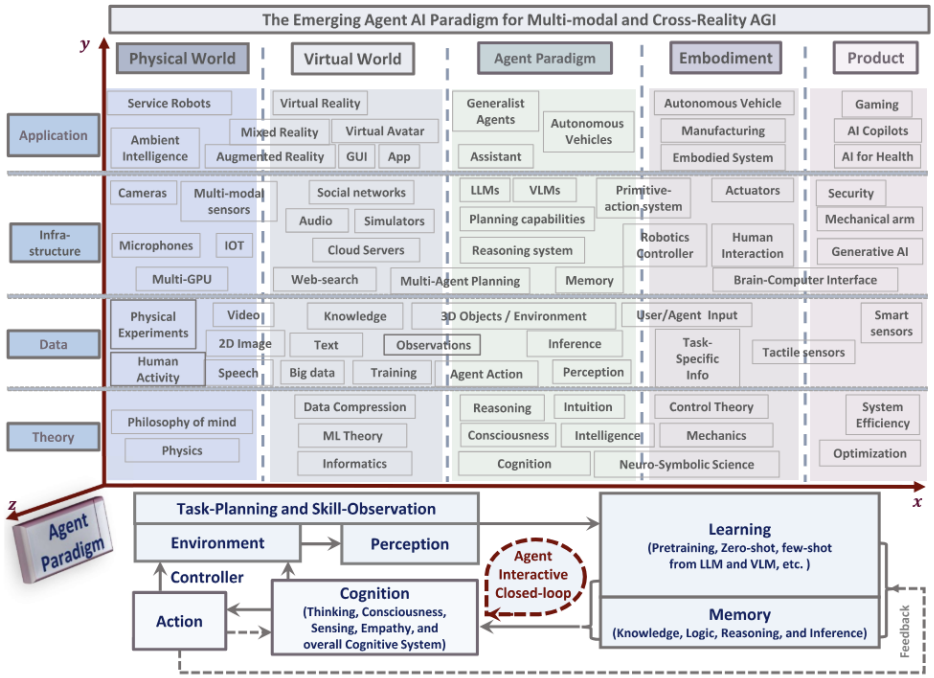

Figure 2.12: Agentic RAG with autonomous agents specializing in specific tasks (Durante et al., 2024)

As stated in the aforementioned paper published by Microsoft and Stanford researchers, AI agents that exclusively rely on pretrained LLMs may incur in hallucinations, since they would have a limited knowledge and hence, unable to understand the complex dynamics of our world.

However, when tackling hallucinations, RAG offers a great method to limit them. By combining RAG and AI Agents, we end up having a solid and adaptable method to prevent misinformation and retrieve suitable content – Agentic RAG is an architecture where agents are plugged in to enhance reasoning and planning prior to selection of RAG pipelines, helping the system in retrieval and/or in reranking and synthetizing retrieved documents or passages based on a user query before sending out answers – this is referred as "Multi-step reasoning", where agents engage in multi-step reasoning, dynamically determining the best sequence of actions to answer a query, rather than a single retrieval-generation pass that is set beforehand. This really improves the overall RAG architecture, since agents can perform additional reasoning tasks and steps, modify their queries based on context, and integrate information from multiple data sources to construct more comprehensive and accurate outputs, effectively taking dynamic decisions and tackle complex questions. There are 3 main reasons why Agentic RAG presents a great opportunity for our case study and in general, for knowledge-based queries in corporations and niche industries: First, Agentic RAG can understand the broader context of a query, asking follow-up questions and refining its understanding based on new information. This allows it to handle complex, multi-part queries that require a nuanced understanding of the context. Second, unlike traditional (naive) RAG, which may only summarize based on the top-k self-rag retrieved documents, agentic RAG can dynamically select and combine information from an extended set of documents to create tailored summaries that address specific user needs. Third, Agentic RAG can perform sophisticated analytics tasks like text-to-SQL conversion, which involves breaking down text queries into structured database queries, enhancing the capabilities of data-driven industries such as finance, which is our domain of interest; this could potentially be very useful when dealing with financial formulae and ratios to retrieve and calculate in large financial data corpora.

In the case of financial analysis, Agentic RAG would enable investors convert unstructured financial text into structured queries to extract meaningful insights from large datasets, uncovering insights from vast datasets and fact check their validity.

Typical Agentic RAG models involve three main steps:

1. **Initial Query Handling:** Starts with a basic understanding of the query, as the AI agent tries to identify the intentions and needs of users.

2. **Iterative Deepening:** Based on the initial output, the agent determines if further information is required, asking additional questions or retrieving more data as needed.

3. **Integration and Response Generation:** Integrates all collected information, applies domain-specific rules, and generates a comprehensive response or analysis.

In the next chapters, we will demystify the potentials of Agentic RAG to enhance fact-checking mechanisms and to provide reliable answers on finance-related topics.

Chapter 3

Methodology

3.1 A mixed design: Combining Self-RAG, Adaptive and Corrective RAG

As we introduced the powerful applications that Agentic RAG may address, a thorough study on how this system can enhance fact-checking and hallucinations mitigation was carried out. After assessing Self-RAG, Corrective-RAG and Adaptive-RAG independently, a good compromise lies in mixing their capabilities to enhance the Agentic RAG architecture. This chapter outlines the methodologies and frameworks employed to boost the performance of Large Language Models (LLMs) in fact-checking financial information using Retrieval-Augmented Generation (RAG). The process encompasses data collection, data preprocessing, the design of the Agentic RAG architecture, and the implementation of all the aforementioned RAG techniques. The goal is to create a robust system capable of providing accurate and reliable financial information by leveraging state-of-the-art technologies in Generative AI. To maximise Agentic RAG revolutionary abilities, these Advanced RAG methods were all utilized in our system, to mitigate the problem of hallucinations in financial-data focused questions on the stock market. The architecture is mostly inspired by a LangChain engineering project, called "Local RAG agent with LLaMA3", which introduced the nuances and powerful tools of LangGraph, a LangChain proprietary framework that allows users to interact with Large Language model by integrating RAG modules in a user-friendly and intuitive way (Langchain RAG, 2024).

3.2 Data Collection: Extracting Financial news data through FNSPID

3.2.1 Data Sources

Data collection and creation are arguably the most crucial part of our experiment. As we outlined in previous sections, data quality and veracity are essential to provide faithful and trustworthy information to investors and financial institutions. Financial datasets are deemed to be extremely precious for financial institutions and investors and hence, refined, and accurate financial datasets have high prices due to their demand. However, after researching multiple financial data providers, we found a completely open-source database called FNSPID. FNSPID provides 2 sets of data: a dataset with over 29.7 million stock prices and 15.7 million time-aligned financial news records for 4,775 S&P500 companies, covering the period from 1999 to 2023, sourced from 4 stock market news websites; specifically, the experiment news data we extracted was collected from Nasdaq, a primary stock exchange, which also provides reliable and verifiable information about stock markets (Dong et al., 2024).

3.2.2 Data Scope

Due to the limited local system capacity and the scope of our project, which is demonstrative and in no way comprehensive, apple stock news from November 2023 to December 2023 have been used in our system. The final dataset consists of 699 Apple news, with all having information about their publication date, the stock discussed in the articles (in our case, Apple) and a summary of these articles. Replacing the full corpus of articles with summaries presents 2 main advantages for our system:

1. **Reduced noise and useful information:** Instead of navigating through often repetitive and useless information, summaries provide concise and critical facts about articles, ignoring noisy and confusing additional information such as articles' introduction and authors' opinions, as well as additional stock ticks and metadata which might confuse and hinder the retrieval of data.

2. **Data size and efficiency:** As FNSPID financial articles were articulated and long, corpus size emerged as a challenging issue in data processing, given that our initial intent was to run and process data locally.

3.2.3 Data Processing and Structure

The data processing part has been straightforward. Once a thorough explorative data analysis was completed using Polars, which mainly implied checking data quality and removing unnecessary features, the apple dataframe was built. An important aspect was to enhance the speed of data processing; working with an 5 GB dataset was unbearable for a local machine; thus, the file was converted into parquet, an extremely efficient storage file type which compressed the initial CSV, making it 10 times smaller in size. The initial dataframe had the following attributes:

Column Name	**Type**
Unnamed: 0 (Index)	Float
Date (Article Publication Date)	String
Article_title	String
Stock_symbol (stock tick)	String
Url (Nasdaq article link)	String
Publisher	String
Author	String
Article (corpus)	String
Lsa_summary (Summary Lsa algorithm)	String
Luhn_summary (Summary Luhn algorithm)	String
Textrank_summary (Summary Textrank)	String
Lexrank_summary (Summary Lexrank)	String

Table 3.1: Table showing column names and their respective types.

As one can immediately notice, there are 4 final columns which may look unfamiliar to some. These refer to article summaries made by 4 summarisation NLP algorithms; to make it short, these techniques utilize different approaches:

- **Latent Semantic Analyzer (LSA):** Based on decomposing the data into low dimensional space. LSA can store the semantics of given text while summarizing.
- **Luhn algorithm:** Based on the frequency method, giving more importance to frequent and contextually important keywords and/or phrases.
- **LexRank:** An unsupervised machine learning-based approach in which we use the TextRank approach to find the summary of our sentences. Using cosine similarity and vector-based algorithms, the minimum cosine distance among various words is calculated, and the more similar words are stored together.
- **TextRank:** Based on PageRank, the famous graph-based algorithm used by Google to rank search results.

Among those techniques, LSA summaries were picked, as their length and contexts were found to be satisfying on a qualitative basis; however, it is very likely that TextRank quality would be higher – however, for our analysis purposes, it is better to prioritise size reduction rather than comprehensiveness (as long as critical information is retained). Eventually, only the Date, Article_title and Lsa_summary were merged into a unique column "Summary" which entails the most relevant information.

3.3 Analytical Framework: Using RAG and LLMs (LangChain, and Llama3 etc.) for Fact-Checking

In this section, the analytical framework used in this study is presented, able to leverage RAG and LLMs in fact-checking financial information. While we develop the framework to try to address the shortcomings of the current ordinary LLMs, in this case, we aim more particularly to reduce the tendency of the LLMs to hallucinate, and inability to get current information since training data has a fixed date for the information cut-off.

Below, we briefly illustrate the components of the Analytical Framework to build the system:

3.3.1 Data Collection and Preparation

- **Dataset:** The dataset used for this study consists of financial news articles related to Apple, extracted from the FNSPID database. This dataset includes over 699 articles summarizing financial events and stock performance of Apple from November 2023 to December 2023.

- **Summarization:** The articles were summarized using the Latent Semantic Analyzer (LSA) to reduce noise and focus on essential information. Summarization helps in efficiently processing the data by removing redundant and less relevant content, which is critical for improving retrieval and generation performance.

3.3.2 Text Embedding and Tokenization

- **Embedding Model Selection:** We utilize the `gte-large-en-v1.5` model from Alibaba-NLP for text embedding. This model, implemented via the `AutoTokenizer` and `AutoModel` classes from the hugging chat `transformers` library, converts text into dense vector representations, capturing semantic relationships between words and phrases. The model was picked as it is lightweight and it is one of the most powerful embedding models for retrieval in MTEB, the most famous benchmark for Embedding models (MTEB, 2024).

- **Embedding Process:** Texts are tokenized and transformed into embeddings using a forward pass through the model. The embeddings are normalized to ensure they lie within a unit range, facilitating efficient similarity searches.

3.3.3 Document Retrieval

- **Indexing:** The financial news articles are indexed using FAISS (Facebook AI Similarity Search), enabling rapid retrieval of relevant documents. The indexing process involves converting each document into its embedding representation and storing these embeddings in the FAISS index. This model enables efficient similarity search and clustering of dense vectors. Indeed, this model is widely used in industry and renowned as one of the most effective ones, and lastly, it is completely open source (Johnson et al. , 2017).

- **Dense Retrieval:** Queries are converted into embeddings and matched against the indexed document embeddings to retrieve the most relevant documents. Dense retrieval leverages the semantic similarity between the query and documents, going beyond simple keyword matching. Specifically, it uses a nearest neighbours' search, where it finds the closest documents (in our case set to $k = 5$) to the user query vector in high-dimensional vector space.

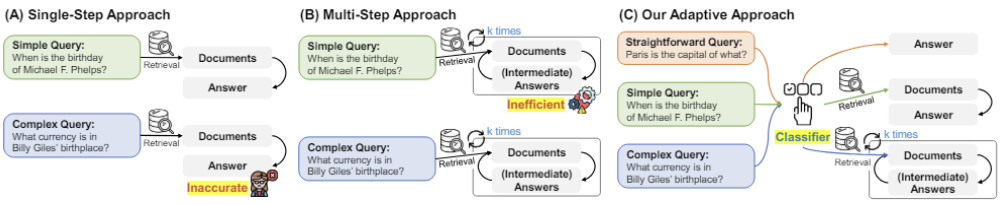

Figure 3.1: The Adaptive-RAG framework differentiating query complexities (Karpukhin et al., 2020)

3.3.4 Grading and Filtering

- **Retrieval Grader:** A custom retrieval grader assesses the relevance of the retrieved documents to the user's query. This grader, built on the ChatGroq language model (llama3 running in a free-to-use computing provider), uses a binary scoring system (`yes` or `no`) to evaluate document relevance based on predefined prompt templates. Essentially, the large language model is given a prompt where it is asked if the context is relevant to answer the question.

- **Hallucination Grader:** This component checks the factual accuracy of the generated answers by comparing them against the retrieved documents. It ensures that the answers are grounded in the provided context, reducing the risk of hallucinations. This is a paramount component of our pipeline, acting as a final checker for answers' reliability and trustworthiness.

- **Answer Grader:** This grader evaluates the utility of the generated answers, determining if they adequately address the user's queries.

3.3.5 Answer Generation

- **RAG Chain:** The RAG chain integrates retrieved documents as context for generating answers. This process involves formatting the retrieved documents and feeding them into a language model (Llama3) along with the user query. The language model generates concise, contextually relevant answers.

3.3.6 Workflow and Conditional Logic

- **State Management:** The system's state, including the question, retrieved documents, generated answers, and search status, is managed using a `TypedDict` to ensure type safety and clarity.

- **Node Definitions:** Key processing steps are defined as nodes, including retrieval, generation, grading, and web search. Each node is responsible for a specific part of the workflow.

- **Conditional Routing:** The system uses conditional logic to route the processing flow based on the outcomes of various steps. For instance,

if retrieved documents are not relevant, the system can trigger a web search; if multiple searches fail to find relevant documents, a default response is generated.

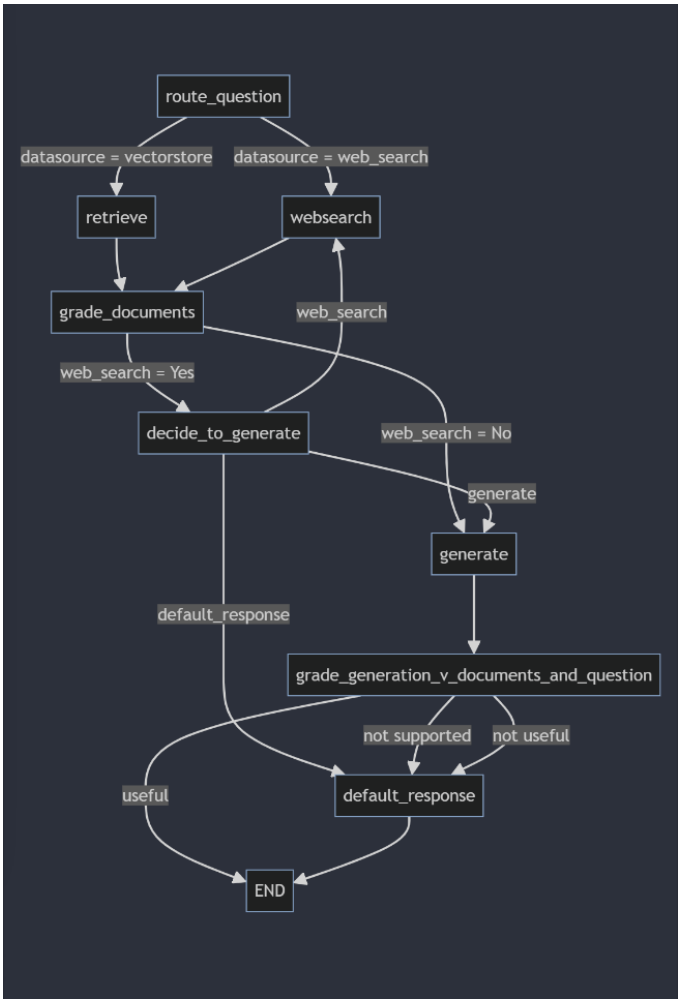

Figure 3.2: An illustration of how our conditional workflow works. Different conditions, which are output of nodes' functions, determine which steps to take.

3.3.7 Functions and Nodes Explanation

Below, a walk-through the application key features is shown, explaining the functions' role and what nodes are set to do:

- **Functions (Agentic modules):** These nodes are the steps the user's query can be subject to. They're the foundation of the whole system, routing and taking decisions on the next steps and how to proceed in the workflow.

  1. `route_question`:
     - Routes to `websearch` if the datasource if the answer is deemed to be answered via a websearch (for this task, we use Tavily, a user-friendly and open-source search engine optimized for RAG applications).
     - Routes to `retrieve` if the datasource is `"vectorstore"`.
  2. `websearch`:
     - Performs a web search and then routes to `grade_documents`.
  3. `retrieve`:
     - Retrieves documents based on the question and then routes to `grade_documents`.
  4. `grade_documents`:
     - If any document is not relevant, sets `web_search` to `"Yes"`.
     - If `web_search` is `"Yes"`, routes to `decide_to_generate`.
     - If `web_search` is `"No"`, routes to `generate`.
  5. `decide_to_generate`:
     - If `web_search` is `"Yes"`, routes to `websearch` again.
     - Otherwise, routes to `generate` or `default_response` (an answer saying that the model can't answer due to a lack of confidence and evidence from retrieved data), based on the provided context and its usefulness in proving an accurate and data-backed answer.
  6. `generate`:
     - Generates an answer and then routes to the next function (see below).

7. `grade_generation_v_documents_and_question`:
   - Checks if the generated answer is grounded in the documents and answers the question.
   - If the answer is not supported, routes to `default_response`.
   - If the answer is useful, routes to `END`.
   - If the answer is not useful, routes to `default_response`.
8. `default_response`:
   - Generates a default response indicating the inability to answer confidently.
   - Routes to `END`.

3.3.8 Steps and Nodes

Nodes are the actual steps where functions get executed.

- `route_question`: This is the starting point that determines whether to route to `websearch` or `retrieve` based on the data source.
- `websearch`: Performs a web search and returns documents.
- `retrieve`: Retrieves documents from our FAISS vector store.
- `grade_documents`: Grades the relevance of retrieved documents.
- `decide_to_generate`: Decides whether to generate an answer or perform another web search based on the relevance of documents.
- `generate`: Generates an answer using retrieved documents.
- `grade_generation_v_documents_and_question`: Checks if the generated answer is grounded in the documents and answers the question.
- `default_response`: Generates a default response indicating the inability to answer confidently.
- `END`: Represents the end of the workflow.

3.3.9 Evaluation and Testing

Back to our framework, the final part involves an evaluation and testing section which comprises 2 fundamental characteristics:

- **Batch Processing:** The framework is tested with a batch of questions, both answerable and non-answerable, to evaluate its performance. Metrics such as accuracy, precision, recall, and F1 score are calculated to quantify the system's effectiveness.
- **Error Handling:** Robust error handling mechanisms are in place to catch and log issues during processing, ensuring that the system can handle unexpected inputs.

3.3.10 Execution and Evaluation

Finally, to execute the system, the workflow is compiled, and questions are processed in batches. Each question triggers a series of actions within the defined workflow, including document retrieval, relevance grading, answer generation, and final grading. The responses are collected and evaluated against expected answers to determine the system's performance metrics. Here is a summary of the key steps in the execution process:

1. Initialize the System: Load models, tokenizer, and data as well as the already processed embeddings of our dataset.
2. Process Each Question: For each question, retrieve documents using FAISS, generate an answer, and grade the answer's relevance and accuracy using Llama3 acting as multiple agents.
3. Evaluate Performance: Calculate metrics based on the system's responses to measure its accuracy, precision, recall, and F1 score.

3.4 Comparative Analysis: Performance of RAG-enhanced LLMs versus traditional LLMs in identifying and correcting misinformation.

In this chapter, the testing and evaluation of our RAG application is comprehensively presented, showing the questions asked to the model and the expected answers to gauge its effectiveness in retrieving and fetching relevant information for accurate answer generation. As briefly mentioned in the last chapter, the model assessment will be based on 4 metrics which are widely used in the industry to evaluate RAG and LLMs performance: accuracy, precision, recall, and F1 score. To set up our evaluation method, a set of questions have been selected, based on the knowledge base content, differing those which could be "answerable" by the model with little or none further external knowledge, and those which were unlikely to be given a solid answer. Below, the set of questions:

3.4.1 Answerable questions:

- "Tell me one Apple's news of December 16, 2023?"
- "Did Apple announce any new products in December 2023?"
- "What is Apple's current market cap as of December 2023?"
- "What did Warren Buffett say about Apple in his latest statement?"
- "How did the Chinese government's policies impact Apple in December 2023?"
- "What is the status of Apple's Series 9 and Ultra 2 smartwatches in the US?"
- "How did Apple's revenue compare to Amazon's in 2023?"
- "What are the top ETFs holding Apple stocks as of December 2023?"
- "What was the Zacks recommendation for Apple in December 2023?"

- "What impact did China's ban on iPhones have on Apple shares in December 2023?"
- "What are the details of Apple's partnership with Goldman Sachs in December 2023?"
- "How did Nvidia's revenue growth compare to Apple's in recent years?"
- "What was Apple's response to the ITC's decision on its smartwatches in December 2023?"
- "How did Apple's stock price move after the Federal Reserve's announcement in December 2023?"
- "What did brokerage recommendations say about investing in Apple in December 2023?"
- "How much of Berkshire Hathaway's portfolio is invested in Apple as of December 2023?"
- "What is the significance of Apple in various ETFs as of December 2023?"
- "What were Apple's revenue estimates for the current fiscal year as of December 2023?"
- "What are the current legal challenges Apple is facing regarding its technology in December 2023?"
- "How did the EU's Digital Markets Act impact Apple in December 2023?"

3.4.2 Unanswerable questions, given the knowledge model limited context (Nov-Dec 2023):

- "What is the latest version of iOS released by Apple in 2024?"
- "Who are the key executives at Apple as of 2024?"
- "What are the latest features of the iPhone 15?"
- "How many stores did Apple open worldwide in 2024?"
- "What new markets is Apple planning to enter in 2024?"
- "How did Apple perform in the latest customer satisfaction survey in 2024?"
- "What new technologies is Apple developing for future products in 2024?"
- "What were the key highlights from Apple's latest WWDC event in 2024?"
- "What new services did Apple launch in 2024?"
- "What is Apple's strategy for expanding its presence in the automotive industry in 2024?"
- "How did Apple contribute to environmental sustainability in 2024?"
- "What is the latest market share of Apple in the smartphone industry in 2024?"
- "What are the main features of the Apple Vision Pro?"
- "How many patents did Apple file in 2024?"
- "What partnerships did Apple form with other tech companies in 2024?"
- "How has Apple adapted its business model in response to global economic changes in 2024?"
- "What philanthropic activities did Apple undertake in 2024?"
- "What is Apple's stance on the latest data privacy regulations in 2024?"

- "What are the projected sales figures for the upcoming iPad models in 2024?"
- "How has Apple's R&D expenditure changed over the past five years?"

Given the answers, our task is a classification evaluation. When assessing the model, we convert the questions into Boolean values, corresponding to their answerability, that is, answerable questions are classified as "TRUE" and the others as "FALSE". This ensures that we can easily compute our metrics and compare them against our baseline of "expected answers" with the first 20 being "TRUE" and the rest "FALSE". Since we stressed the importance of avoiding hallucinations in our case study, 2 further metrics must be included to assess it. We can roughly measure, especially on unanswerable questions, the Negative predictive value and the true negative value which are defined as follows:

Negative Predictive Value

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}}$$

where TN is true negatives and FN indicates the number of false negatives. This measure exactly corresponds to what precision is for true observations – it reveals how any false predicted observations we got correct out of all that we predict to be negative. The higher this measure is, the more solid and trustworthy our negative predictions are.

True Negative Rate

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

where TN stands for True negative, whereas FP for False positives. This metric simply tells us how many negatives we have predicted out of all actual negatives. The higher TNR, the higher number of false observations we can spot and not leave behind.

Accuracy	Predictions/ Classifications	$\frac{\text{Correct}}{\text{Correct + Incorrect}}$
Precision	Predictions/ Classifications	$\frac{\text{True Positive}}{\text{True Positive + False Positive}}$
Recall	Predictions/ Classifications	$\frac{\text{True Positive}}{\text{True Positive + False Negative}}$
F1	Predictions/ Classifications	$\frac{\text{2 * True Positive}}{\text{True Positive + 0.5 (False Positive + False Negative)}}$

Figure 3.3: In the graphic above, the other metrics are displayed (Martin, 2022). As we said, precision is exactly what NPV is for negatives, and recall has the same role of the True Negative Rate (TNR).

A different explanation must be done for F1 score: it is the harmonic mean of the precision and recall. It thus symmetrically represents both precision and recall in one metric (F1 Score, 2024). As we ran the workflow and asked questions to the system, the results are visible in the next page:

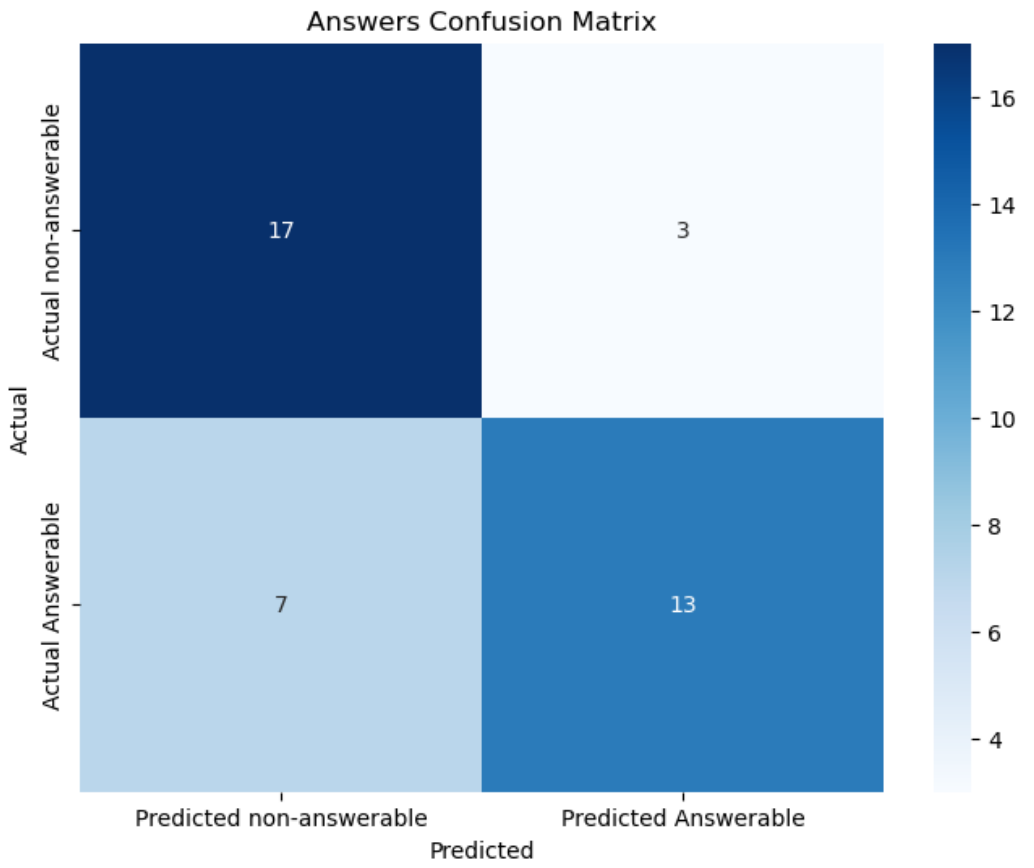

Figure 3.4: The Confusion matrix reveals the goodness of our system in reducing hallucinations.

Looking at the left diagonal we see that it correctly answered to 30 out of 40 questions, a good result considering the limitations of our RAG application. It is particularly compelling to see how good the system was in not giving answers which were not backed by news. Indeed, out of the 3 errors, 2 were spotted to be likely answerable after analysing the model logs: The question "What are the latest features of the iPhone 15?" was answered thanks to 2 web search retrievals made that were found to be true by the hallucination grader and the decide_to_generate modules.

Instead, the query "What new technologies is Apple developing for future products in 2024?" arrived at the last state "generate" and the system gave this answer: "I don't know. The provided context does not mention specific new technologies Apple is developing for future products in 2024." In all other cases, after questions passed the other checks, the generate module always answered with specific and concise answers, but on this occasion, it ultimately decided that the retrieved context was not enough to respond to this query – this reinforces once again, the idea that AI agents in RAG applications can really help further assess the reliability of answers, and auto-detect its previous "mistakes". It is important to note that this answer was labelled as "TRUE" and hence, answerable, simply because our system labelled answers as "FALSE" when the `default_response` was given (that is when, in the `decide_to_generate` module after attempting web searches, the model decided that there was at least one relevant document out of the 5 retrieved and generation was doable since it was relevant to the query). The question ""How many stores did Apple open worldwide in 2024?" was incorrectly answered, since it's theoretically wrong answering a question without providing its "as of" date: "According to the provided context, Apple operates over 530 retail stores in at least 25 countries as of 2024."

However, it is also crucial to note that the model has not performed extremely well on answerable questions. Out of 20, 7 were said to be unanswerable, although they were, looking at the document: this may be due to multiple factors such as wrong retrievals or lack of confidence in giving a definitive answer based on the retrieved context.

3.4.3 Results

Below, all the metrics are shown:

Metric	**Value**
Accuracy	0.75
Precision	0.8125
Recall	0.65
F1 Score	0.72
Average Response Time	12 seconds

Table 3.2: Overall Performance (without including the 2 actual "TRUE")

Metric	**Value**
Accuracy	0.65
Precision	1
Recall	0.65
F1 Score	0.79

Table 3.3: Metrics on answerable questions (without including the 2 actual "TRUE")

Metric	**Value**
Accuracy	0.95
NPV	1
TNR	0.95

Table 3.4: Metrics on unanswerable questions (including the 2 actual "TRUE")

3.5 Case Study: Application in stock market news

As we described our system, a brief recap on our case study is necessary. In this case study, we focused on applying Retrieval-Augmented Generation (RAG) enhanced Large Language Models (LLMs) to fact-check and verify stock market news related to Apple during November and December 2023. By leveraging a custom Agentic RAG, we aimed to mitigate the issue of hallucinations and ensure that the information generated by the LLMs was accurate, timely, and relevant. We employed various summarization algorithms to compress these articles into concise summaries, reducing noise and emphasizing critical information. The summarized dataset was then embedded using the `gte-large-en-v1.5` model and indexed with FAISS to enable efficient document retrieval. Our mixed Agentic-RAG framework involved multiple steps, including document retrieval, relevance grading, answer generation, and hallucination grading. The use of AI agents for dynamic query handling and multi-step reasoning further improved the system's ability to provide accurate and contextually relevant answers. During testing, we evaluated the system's performance on a set of both answerable and unanswerable questions about Apple stock market news. The results showed that the RAG-enhanced LLMs significantly reduced hallucinations and provided accurate answers backed by retrieved documents. The system's accuracy, precision, recall, and F1 score metrics demonstrated its effectiveness in fact-checking financial news, with success in identifying unanswerable questions and avoiding false positives. This application is purely demonstrative but shows us the potential of Agentic RAG in fact-checking: investors could build a personal LLM with their data to verify its credibility and produce rich and relevant information without relying fully on third media parties and general knowledge LLMs – this solution is tailored to stock news which are, by definition, great drivers for the stock market and can drastically impact on its behaviour over time. This tool, if continuously improved and integrated by large and reliable data, can also function as a foundation for proprietary LLMs for media companies; Financial data and news providers could build this system to provide users with an easy and user-friendly platform to fetch important news, as an alternative to traditional search engines which lack the ability to actively interact with users.

3.6 Insights and Implications of our Findings

The findings from this study bring up interesting insights and implications for the application of RAG-enhanced LLMs in tackling financial misinformation:

1. **Reduction of Hallucinations:** The integration of RAG can significantly mitigate the issue of hallucinations in LLM-generated responses. By grounding answers in retrieved documents, the system ensures that the information provided is factually accurate and verifiable.

2. **Enhanced Accuracy and Relevance:** The use of advanced retrieval and grading mechanisms improves the relevance and accuracy of generated answers. The ability to dynamically retrieve and synthesize information from multiple sources allows the system to handle complex and nuanced queries more effectively.

3. **Importance of Data Quality:** The quality and relevance of the dataset are crucial for the success of RAG systems. Summarizing articles and focusing on essential information reduces noise and enhances retrieval efficiency, leading to more accurate and contextually appropriate answers.

4. **Versatility and Adaptability:** The combination of Self-RAG, Adaptive-RAG, and Corrective-RAG techniques, along with AI agents, demonstrates the system's versatility in handling different types of queries and adapting to varying levels of complexity. This adaptability is essential for addressing the diverse needs of financial journalism and analysis.

5. **Implications for Financial Journalism:** The application of RAG-enhanced LLMs in financial journalism can significantly improve the reliability and trustworthiness of financial news. Journalists can leverage these systems to quickly verify information, reduce the spread of misinformation, and provide more accurate reporting.

3.7 Technical Challenges: Addressing RAG and LLM limitations

3.7.1 Handling long-context documents:

Implementing RAG-enhanced LLMs like this poses several challenges. A major issue that we intentionally circumvented is handling long-context documents. LLMs suffer from the limitation of their context window, which can eventually lead to incomplete retrieval and generation by agents. When speaking of chunking and indexing, we must outline how critical splitting long documents into manageable chunks for indexing and retrieval is as it can result in loss of context or coherence; that is arguably one of the major factors that worsened performance on answerable questions. Additionally, addressing queries that require information spanning multiple documents (multi-hop reasoning) remains a significant challenge. Ensuring that the system can accurately integrate information from different documents or web searches to form a coherent and accurate response is difficult and often computationally intensive.

3.7.2 Robustness of Retrieval mechanisms:

This is arguably the other major factor that made this experiment fail in detecting relevant context for queries. To limit processing and answering time we relied on a reliable and simple FAISS framework which allows builders to retrieve information in an intuitive way. However, near neighbour search as well as other metrics used for similarity might not always ideal as data shape and form change. For example, we set $k = 5$ to speed up to workflow, but enhancing mechanisms such as re-ranking can substantially improve the system, again, at the cost of computational resources. Also, ensuring that the retrieved documents are relevant and accurate is crucial. Dense retrieval models like this can sometimes return irrelevant or low-quality results, which can negatively impact the final output. Developing robust filtering and grading mechanisms to handle this issue is necessary but challenging if one wants to maintain this application running. Finally, keeping the index updated with the latest information is crucial, especially in fast-moving fields like finance. However, continuously updating and maintaining large indexes requires significant resources and careful management.

3.7.3 Costs and decision-making implications of deploying an enhanced LLM system:

This scenario is specifically true in our use case: running a system like this and making it scalable is costly. We leveraged the capabilities of Groq, a platform that gives access to optimized LLM computational resources (i.e. GPUs); its usage is limited by a maximum token processing capacity which can greatly limit the context window and hence, the possibility for these systems to work with complex and detailed corpora like financial news. Our proposal is far from perfect: Managing a constant flow of real-time financial data requires an important focus on data quality and timeliness which require expertise and time, the famous quote "garbage in, garbage out" applies also to RAG system; Both the retrieval and generation processes in RAG-enhanced systems are computationally demanding. This includes the initial training and fine-tuning of models (which we haven't done but it's often necessary to specialize the LLM agents) as well as real-time query processing. In this experiment only the final outcome was described but many failed attempts thwarted our way, including limitation in data processing due to capacity constraints and finding good embedding models, storing frameworks and retrieval techniques, as some were not suitable for our type of data. Ultimately, choosing the right frameworks and tools to build RAG-enhanced LLM can be tough and time-consuming, and that is why it is considered one of the hardest technical challenges faced.

3.7.4 Ethical Considerations: Bias, Privacy, and Transparency in Automated Fact-Checking

LLMs and retrieval models can inherit biases present in their training data. This can lead to biased or skewed outputs, which is particularly concerning in the context of financial information where impartiality is crucial. Addressing and mitigating bias requires careful selection and preprocessing of training data, as well as implementing bias detection and correction mechanisms within the models; this task is particularly difficult as users should rely on data and news publishers whereas publishers themselves would have to make sure to refine their data quality and timeliness data pipelines as well as handling potential sensitive financial information, which requires strict privacy measures to ensure that proprietary or confidential data is not exposed or misused and as said in the initial chapters, can provoke unexpected finan-

cial loss and even worse, markets' anomalies. To conclude, accountability and explainability are other 2 major issues to tackle. Providing transparency in how the system retrieves and generates answers is crucial for building trust and improve its functioning. This includes explaining the sources of retrieved documents and the rationale behind generated responses which once again, depend on AI agents which are LLMs that themselves are exposed to hallucinations. Finally, ensuring accountability involves tracking the system's decision-making process and being able to audit and review outputs, especially in cases where incorrect or misleading information is produced (Dai et al., 2023).

3.7.5 Limitations of the Study: Scope and Methodological Constraints

Once again, it is paramount to repeat that our study does not represent a comprehensive solution for fact-checking but rather, it intends to discuss the potential improvements that Retrieval-Augmented generation can have on knowledge-based queries on financial news and general data. Indeed, several limitations characterize our RAG-enhanced LLM:

- **Scope of the Dataset, Timeframe and Summarization Quality:** Limited to Apple, the study focused exclusively on financial news related to this company, which may limit the generalizability of the findings to other companies or sectors different to the stock one. Different companies and industries may present unique challenges that were not addressed in this study, such as the predominance of specific unstructured data or lack of textual data, which represents the most suitable data type for RAG system that largely leverage semantic search techniques. The dataset was limited to news articles from November and December 2023. This narrow timeframe may not capture longer-term trends and patterns that could affect the system's performance, as well as not revealing how it behaves when exposed to large contextual data which makes retrieval harder. Moreover, the use of specific summarization algorithms (in this case, Latent Semantic Analyzer) may influence the quality and relevance of the summaries. Alternative summarization techniques might yield better results.

- **Embedding Model Selection:** The choice of Alibaba's embedding model (`gte-large-en-v1.5`) massively affects the retrieval process. While it was chosen for its performance and efficiency, other embedding models could potentially improve retrieval accuracy and relevance. We have not carried out exhaustive research and testing to define which models worked best, as we tried only 2 alternatives.

- **Focus on Classification Metrics:** The study primarily used accuracy, precision, recall, and F1 score as evaluation metrics. While these metrics are important, they do not capture all aspects of system performance – doing so does not permit further model assessment, as it is quite limited in scope – semantic similarity and automated keywords querying are 2 ideas that could help.

- **Limited User Feedback and manual annotations:** The evaluation did not include user feedback or real-world testing, which could provide valuable insights into the system's usability and effectiveness in practical applications. Most importantly, the classification of questions as answerable or unanswerable was based on manual annotations, which may introduce our subjectivity. Automated or more systematic approaches could improve the reliability of these classifications.

- **Static Testing Environment:** Speaking of methodological constraints, the model testing was conducted in a controlled and local environment, which may not fully replicate the dynamic and variable conditions of real-world applications that typically run on cloud architecture and are subject to data drifts and continuous monitoring practices.

Chapter 4

Conclusion and Recommendations

The experiment demonstrated that Retrieval-Augmented Generation (RAG)-enhanced Large Language Models (LLMs) can significantly improve the accuracy and reliability of financial information, particularly in mitigating the issue of hallucinations. By integrating complex agentic retrieval mechanisms and grading processes, the system was able to provide factually correct and contextually relevant answers to queries about stock market news related to Apple. The use of RAG notably decreased the occurrence of hallucinations, ensuring that generated responses were grounded in documents; albeit a proper comparison with different LLMs is not shown, we can still speculate with some numbers. Given that our model is specifically tailored to this data and does not possess the characteristics of general and multi-purpose LLMs, and hence, it is not a grounded and fair comparison, it is worth-noting that our system presents a hallucination rate which is similar to GPT-4 turbo (2.5% vs 5%, which is our TNR) and outperforms larger and more complex models like Gemini Pro, Llama2 with 70 billion parameters and even Llama3 8B, which was our foundational model (Hallucination Leaderboard, 2024).

By manually checking answers and using our metrics, we understand that the system provided high-quality answers with improved accuracy, precision, and relevance, especially for questions related to recent financial news. This is likely due to summarization: Summarizing long articles into concise summaries improved the efficiency of the retrieval process and the overall system performance.

4.0.1 Contributions to the field: Theoretical and Practical Implications

The findings from this study contribute to both the theoretical understanding and practical applications of RAG-enhanced LLMs in the field of financial journalism and information:

- **Mixed RAG approach:** The study highlights the benefits of combining Self-RAG, Adaptive-RAG, and Corrective-RAG approaches to enhance the performance of LLMs in specific domains.
- **Enhanced Understanding of RAG:** The study provides valuable insights into the effectiveness of RAG in addressing common issues with LLMs, such as hallucinations and outdated information; although this might be taken for granted, ensuring that this is the case is key to make LLMs more reliable.
- **Framework for Future Research:** The methodology and framework used in this study can serve as a foundation for future research exploring the integration of RAG with other advanced AI techniques not used that would improve the architecture - fine-tuning the foundational LLM model can be a compelling addition as well as provide complex AI agents with more detailed instructions, that might even be tailored based on the users' preferences.

4.0.2 Contribution to financial information-based fields and practical Implications

This kind of architecture and workflow could be beneficial for several financial information-based fields, including financial journalism, where the system can be used by financial journalists to quickly verify and fact-check news, ensuring the spread of accurate information; on the other hand, retail investors and financial analysts can leverage RAG-enhanced applications to get reliable financial insights, helping in better decision-making.

However, we must include some recommendations and potential good practices, especially for Journalists and Technologists. For journalists, it is reasonable to mention these 3 points:

1. **Integrate RAG-Enhanced Systems:** Journalists may make use of RAG-enhanced LLMs to fact-check and verify financial news, ensuring accuracy and reliability in reporting and news spread.

2. **Continuous Training and Updates:** Although the application is appealing, updating the datasets and models is necessary - this can do the trick when the model needs to fetch the latest financial trends and news, maintaining the relevance of the information provided with users.

3. **Transparency and Accountability:** These 2 aspects are a big topic in Explainable AI these days. Maintaining transparency in the use of AI tools, including documenting sources and processes used in fact-checking are aspects that cannot be overlooked and must be included to make the system accountable and transparent.

For Technologists and "more-aware" users that may want to dig into the technicalities and the building process of these architectures:

- **Develop Robust Retrieval Mechanisms:** A focus on improving the robustness of retrieval mechanisms to handle diverse and complex queries effectively is needed, as further testing can really bring benefits to the model - training AI agents on high-quality and relevant data can surely make a difference too.

- **Optimize Computational Resources:** They should focus in optimizing computational resources to enhance the scalability and efficiency of RAG-enhanced systems, which may be expensive and tedious to manage.

- **Enhance User Interfaces:** First and foremost, users always come first. Developing user-friendly and centred interfaces would allow easy interaction with RAG-enhanced systems, facilitating their adoption by non-technical users. A good example is ollama webui, an open-source project which allows users to use and interrogate open-source LLMs interfaces for free.

4.0.3 Future Research: Potential Areas for Further Investigation

In the final part of this work, we discuss about the most improvable sections of our model, providing some tips and points of discussion for further development, recalling the main challenges faced in this experiment. First, extending the research and the scope of the data (the knowledge base) would be crucial to let the model learn more on other companies, sectors, and financial instruments, testing the generalizability of RAG-enhanced systems on miscellaneous financial sectors which may be very specialized and require personalized approaches. Perhaps, the main limitation in this system approach lies into retrieval and long-context handling: Improving Retrieval and Generation techniques would exponentially increase the capability of this Agentic RAG LLM model to answer factually and avoid missing out on information existing in the knowledge base – as we have seen, in our analysis and metrics, the production of answerable questions was the "worst" result of our experiment, with 7 out of 20 questions which the model claimed to not have enough context to answer. Exploring knowledge graph can be worthwhile too. A knowledge graph is a knowledge base that uses a graph-structured data model or topology to represent and operate on data. Knowledge graphs are often used to store interlinked descriptions of entities — objects, events, situations, or abstract concepts – while also encoding the semantics or relationships underlying these entities, as they power most of the best data-intensive search engines and social networks such as Google, LinkedIn and Facebook among others (Knowledge Graph, 2024). However, it is arguably even more important to enhance long-Context Handling: developing advanced techniques for handling long-context documents and multi-hop reasoning can revamp and boost retrieval accuracy, while implementing adaptive learning mechanisms such as reinforcement learning that can dynamically update the system based on new information and user feedback.

Finally, the most delicate and probably discussed AI topics speculating on AGI fall into Bias Detection and Correction: Research methods to detect and correct biases in training data and model outputs are to be explored and tested, so the model can ensure fairness and impartiality. Making these happen will likely take time, but a key methodology can accelerate this learning process: focusing on user-centric evaluations and testing. LLMs builders must consider users' feedback in the evaluation and improvement of RAG-enhanced systems - meeting the practical needs of end-users is surely beneficial while conducting real-world testing and case studies to validate the effectiveness and usability of RAG-enhanced systems in practical scenarios.

In conclusion, this study underscored the potential of RAG-enhanced LLMs to improve the way financial information is verified and disseminated. By addressing current challenges and exploring future research directions, these systems can be further refined and widely adopted, significantly contributing to the integrity and reliability of financial news, which are again, powerful and potentially detrimental, if not properly verified.

Bibliography

N. Strauß. (2018). Financial journalism in today's high-frequency news and information era. Journal of Financial Reporting. Retrieved from https://journals.sagepub.com/doi/pdf/10.1177/1464884917753556

Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, & Toutanova, Kristina. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805. Retrieved from https://arxiv.org/pdf/1810.04805.

Khurana. (2022). Natural language processing: state of the art, current trends and challenges. Retrieved from https://link.springer.com/article/10.1007/s11042-022-13428-4.

Pennycook. (2023). Who falls for fake news? The roles of bullshit receptivity, overclaiming, familiarity, and analytic thinking. Retrived from https://onlinelibrary.wiley.com/doi/abs/10.1111/jopy.12476.

Lin. (2023). Young Investors Tend to Fall for Online Financial Misinformation. Retrieved from https://www.planadviser.com/young-investors-tend-fall-online-financial-misinformation/.

Rangapur. (2023). Investigating Online Financial Misinformation and Its Consequences: A Computational Perspective. Retrieved from https://arxiv.org/pdf/2309.12363.

Rathinapandi. (2023). Tokenization vs. Embedding: Understanding the Differences and Their Importance in NLP. Medium. Retrieved from https://geoffrey-geofe.medium.com/tokenization-vs-embedding-understanding-the-differences-and-their-importance-in-nlp-b62718b5964a

Nyandwi. (2023). The Transformer Blueprint: A Holistic Guide to the Transformer Neural Network Architecture. Retrieved from https://deeprevision.github.io/posts/001-transformer/

Raschka. (2023). Understanding and Coding the Self-Attention Mechanism of Large Language Models From Scratch. Retrieved from https://sebastianraschka.com/blog/2023/self-attention-from-scratch.html

Zetterlund. (2023). Overcoming Challenges in Retrieval Augmented Generation (RAG). Retrieved from https://torbjornzetterlund.com/overcoming-challenges-in-retrieval-augmented-generation-rag/#gsc.tab=0.

Han. (2023). A Comprehensive Survey on Vector Database: Storage and Retrieval Technique, Challenge. Retrieved from https://arxiv.org/pdf/2310.11703

Guo et al. (2023). A Survey on Automated Fact-Checking. Retrieved from https://arxiv.org/pdf/2108.11896

Zhao et al. . (2022). Dense Text Retrieval based on Pretrained Language Models: A Survey. Retrieved from https://arxiv.org/pdf/2211.14876.

Paltiel. (2023). What is Similarity Search? [Definition and Use Cases]. Retrieved from https://blog.hyper-space.io/what-is-similarity-search-definition-and-use-cases.

IBM. (2023). Retrieval Augmented Generation: Enhancing Language Models with External Knowledge. IBM Research Papers. Retrieved from https://research.ibm.com/blog/retrieval-augmented-generation-RAG.

Lewis, P. et al. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. Advances in Neural Information Processing Systems, 33, 9459-9474. Retrieved from https://arxiv.org/pdf/2005.11401.

Pennycook, G., & Rand, D. G. (2023). The Improbability of Fake News Impact on Financial Literacy. Journal of Economic Psychology. Retrieved from https://onlinelibrary.wiley.com/doi/full/10.1111/jopy.12476

Reuters. (2018). Understanding the Promise and Limits of Automated Fact-Checking. Retrieved from https://reutersinstitute.politics.ox.ac.uk/sites/default/files/2018-02/graves_factsheet_180226%20FINAL.pdf

Sage. (2023). Checking the Fact-Checkers: The Role of Source Type, Perceived Credibility, and Individual Differences in Fact-Checking Effectiveness Journal of Communication Research. Retrieved from https://journals.sagepub.com/doi/10.1177/00936502231206419

Cavazos. (2023). The Economic Impact of Financial Misinformation. University of Baltimore Economic Research Report. Retrieved from https://www.ubalt.edu/news/news-releases.cfm?id=3425

Washington Post. (2023). Washington Post Fact-Checker Methodology. Washington Post Research. Retrieved from https://www.washingtonpost.com/politics/2019/01/07/about-fact-checker/

Wikipedia. (2023). Bandwagon Effect. Retrieved from https://en.wikipedia.org/wiki/Bandwagon_effect.

Asai et al. (2023). SELF-RAG: Learning to retrieve, generate, and critique through self-reflection. Retrieved from https://arxiv.org/pdf/2310.11511.

Durante et al. (2024). Agent-Based Systems in Retrieval-Augmented Generation. arXiv preprint arXiv:2401.03568. Retrieved from https://arxiv.org/pdf/2401.03568.

Zetterlund, T. (2023). Overcoming Challenges in Retrieval Augmented Generation (RAG). Retrieved from https://torbjornzetterlund.com/overcoming-challenges-in-retrieval-augmented-generation-rag/#gsc.tab=0.

Wikipedia. (2024). Artificial General Intelligence. Retrieved from https://en.wikipedia.org/wiki/Artificial_general_intelligence.

Z.Dong & X.Fan. (2024). FNSPID: A Comprehensive Financial News Dataset in Time Series. Retrieved from https://arxiv.org/html/2402.06698v1

Langchain. (2024). LangGraph RAG Agent with Llama 3. Retrieved from https://github.com/langchain-ai/langgraph/blob/main/examples/rag/langgraph_rag_agent_llama3_local.ipynb.

Hugging Face. (2023). Massive Text Embedding Benchmark (MTEB). Retrieved from https://huggingface.co/blog/mteb.

Johnson, J., Douze, M., & Jegou, H. (2017). Billion-scale similarity search with GPUs. arXiv preprint arXiv:1702.08734. Retrieved from https://arxiv.org/pdf/1702.08734.

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., ... & Yih, W. T. (2020). Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906. Retrieved from https://arxiv.org/pdf/2004.04906.

Martin. (2022). Unlocking the potential of deep learning for marine ecology: Overview, applications, and outlook https://www.researchgate.net/figure/Evaluation-metrics-accuracy-precision-recall-F-score-and-Intersection-over-Union_fig2_358029719.

Wikipedia. (2024). F1 Score. Retrieved from https://en.wikipedia.org/wiki/F-score.

Dai et al. (2023). Unifying Bias and Unfairness in Information Retrieval: A Survey of Challenges and Opportunities with Large Language Models. arXiv preprint arXiv:2404.11457. Retrieved from https://arxiv.org/pdf/2404.11457.

Wikipedia. (2024). Knowledge Graph. Retrieved from https://en.wikipedia.org/wiki/Knowledge_graph.

Vectara. (2024). Hallucination Leaderboard for LLMs. Retrieved from https://github.com/vectara/hallucination-leaderboard.

Guo, Z., Schlichtkrull, M., & Vlachos, A. (2021). A survey on Automated Fact-checking. arXiv preprint arXiv:2108.11896. Retrieved from https://arxiv.org/pdf/2108.11896.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. arXiv preprint arXiv:1706.03762. Retrieved from https://arxiv.org/pdf/1706.03762.

ChatGPT. (2024). Example of GPT-4 hallucination. Retrieved from https://chat.openai.com/share/d57f9847-e58d-4f19-9855-da91be64d31c.

Liu, X., Yao, Y., Suyi, L., & others. (2024). Large Language Models Are Human-Level Prompt Engineers. arXiv preprint arXiv:2403.14403. Retrieved from https://arxiv.org/pdf/2403.14403.

Yan, S.-Q., Gu, J.-C., Zhu, Y., & Ling, Z.-H. (2024). Corrective Retrieval Augmented Generation. arXiv preprint arXiv:2401.15884. Retrieved from https://arxiv.org/pdf/2401.15884.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, & Gideon Mann. (2023). BloombergGPT: A Large Language Model for Finance. arXiv preprint arXiv:2303.17564. Retrieved from https://arxiv.org/pdf/2303.17564.

The Emergence of the Robo-Advisor. Retrieved from https://www.researchgate.net/publication/336797236_The_Emergence_of_the_Robo-Advisor.

Rangapur, Aman, Wang, Haoran, & Shu, Kai. (2023). Investigating Online Financial Misinformation and Its Consequences: A Computational Perspective. arXiv preprint arXiv:2309.12363. Retrieved from https://arxiv.org/pdf/2309.12363.pdf.

Ballotpedia. (2023). The methodologies of fact-checking. Retrieved from https://ballotpedia.org/The_methodologies_of_fact-checking.

Thorne, James, Vlachos, Andreas, Christodoulopoulos, Christos, & Mittal, Arpit. (2018). FEVER: a large-scale dataset for Fact Extraction and VERification. arXiv preprint arXiv:1803.05355. Retrieved from https://arxiv.org/pdf/1803.05355v3.pdf.

Naveed, Humza, Khan, Asad Ullah, Qiu, Shi, Saqib, Muhammad, Anwar, Saeed, Usman, Muhammad, Akhtar, Naveed, Barnes, Nick, & Mian, Ajmal. (2023). A Comprehensive Overview of Large Language Models. arXiv preprint arXiv:2307.06435. Retrieved from https://arxiv.org/pdf/2307.06435.pdf.

Ovadia, Oded, Brief, Menachem, Mishaeli, Moshik, & Elisha, Oren. (2023). Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLMs. arXiv preprint arXiv:2312.05934. Retrieved from https://arxiv.org/pdf/2312.05934.pdf.

Gao, Yunfan, Xiong, Yun, Gao, Xinyu, Jia, Kangxiang, Pan, Jinliu, Bi, Yuxi, Dai, Yi, Sun, Jiawei, Guo, Qianyu, Wang, Meng, & Wang, Haofen. (2023). Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv preprint arXiv:2312.10997. Retrieved from https://export.arxiv.org/pdf/2312.10997v2.