MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

A PROPOSAL FOR AN AGENT-BASED FRAMEWORK STRUCTURE FOR FINANCIAL FORECASTING USING LARGE LANGUAGE MODELS

M. MUSTAFA AYDIN

MASTER THESIS
Department of Industrial Engineering

Thesis Supervisor
Asst. Prof. Dr. Semih ÖZEL

ISTANBUL, 2025

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

A PROPOSAL FOR AN AGENT-BASED FRAMEWORK STRUCTURE FOR FINANCIAL FORECASTING USING LARGE LANGUAGE MODELS

M. MUSTAFA AYDIN

(524421016)

MASTER THESIS
Department of Industrial Engineering

Thesis Supervisor
Asst. Prof. Dr. Semih ÖZEL

ISTANBUL, 2025

ACKNOWLEDGEMENT

I gratefully acknowledge my thesis supervisor, Asst. Prof. Dr. Semih ÖZEL, for their invaluable guidance, support, and encouragement throughout this research.

I would also like to extend my heartfelt thanks to Zeynep BÜYÜKPALTA, whose patience, understanding, and constant motivation were invaluable in helping me to overcome the challenges of this thesis.

This work was supported by Marmara University BAP unit under the project code FYL-2025-11614.

August 2025 M. Mustafa AYDIN

INDEX

ACKNOWLEDGEMENT iii
INDEX i
ÖZET iii
ABSTRACT iv
ABBREVIATIONS v
LIST OF FIGURES vi
LIST OF TABLES vii
1. INTRODUCTION 1
2. FINANCIAL MARKET PREDICTION 3
2.1. Efficient Markets Hypothesis and the Predictability Paradox 3
2.2. The Rise of Behavioral Finance and Market Anomalies 4
2.3. Traditional Financial Forecasting Approaches and Their Limitations 4
2.4. Beyond Traditional Methods: The Importance of Unstructured Data 5
3. NATURAL LANGUAGE PROCESSING (NLP) 6
3.1. Text Preprocessing 6
4. MACHINE LEARNING AND DEEP LEARNING MODELS 9
4.1. Traditional Machine Learning Models and Their Role in Financial Prediction 9
4.2. Deep Learning 10
5. LARGE LANGUAGE MODELS 11
5.1. Transformer Architecture: Parallelism and the Self-Attention Mechanism 11
6. AGENT BASED SYSTEMS 13
6.1. Fundamental Characteristics and Architectures of Agents 13
6.2. Multi-Agent Systems (MAS): Power of Specialization and Collaboration 14
6.3. LLM-Based Agents: Large Language Models as Reasoning Engines 14
7. METHODOLOGY AND APPLICATION 15
7.1. Data Collection and Storage 15
7.2. Quantitative Data Stream: Borsa Istanbul Real-Time Price Data 16
7.3. Qualitative Data Stream: KAP Disclosures and General Financial News 16
7.4. Data Storage and Infrastructure Architecture 18
7.5. System Monitoring and Operational Reliability 19
7.6. LLM-Based Information Extraction Module: Generating Meaning from Unstructured Text 20
7.7. Prompt Engineering: Turning an LLM into a Financial Analyst 20
7.8. Improving Output Quality: Few-Shot Learning Approach 21
7.9. Multi-Agent System: Distributed Analysis and Decision Fusion 21

7.10. Orchestrator Agent: Task Routing and Coordination 23
7.11. Expert Analyst Agents: Deep Analysis and Scoring 24
7.12. Fusion Agent: Combining Multiple Perspectives and Final Decision 24
7.13. Hybrid Prediction Model: Merging Qualitative and Quantitative Scores 25
7.14. Backtesting Framework 27
7.15. Simulation Assumptions and Realism 27
7.16. Performance Metrics 28
7.17. Baseline Models for Comparison 28
8. RESULTS AND CASE STUDY 30
8.1. Experimental Setup and Methodology 30
8.2. Case Study: Synthesizing Complex News 34
9. CONCLUSION, FUTURE STUDIES, AND LIMITATIONS 37
REFERENCES 39
APPENDIX 46
RESUME ..... **Error! Bookmark not defined.**

ÖZET

BÜYÜK DİL MODELLERİ KULLANILARAK FİNANSAL TAHMİN İÇİN AJAN TABANLI BİR ÇERÇEVE YAPISI ÖNERİSİ

Finansal piyasa hareketlerini tahmin etmek, sayısız nicel ve nitel faktörden etkilenen piyasaların karmaşık, doğrusal olmayan ve gürültülü doğası nedeniyle önemli bir zorluk teşkil etmektedir. Geleneksel tahmin modelleri, ağırlıklı olarak yapılandırılmış sayısal verilere dayanırken; haber makaleleri, şirket duyuruları ve yasal bildirimler gibi yapılandırılmamış metin verilerinin içinde gömülü olan zengin ve bağlama duyarlı bilgiyi yakalamada genellikle başarısız olmaktadır. Bu makale, Büyük Dil Modellerinin (LLM) analitik gücünü bir Çoklu-Ajan Sistemi (MAS) çerçevesi içinde entegre ederek, hisse senedi piyasası tahmini için özgün ve hibrit bir sistem önermekte ve geliştirmektedir. Bu araştırmanın temel amacı, hem gerçek zamanlı piyasa verilerini hem de yapılandırılmamış metinsel bilgileri sistematik olarak işleyerek Borsa İstanbul'da (BİST) işlem gören hisse senetleri için öngörücü sinyaller üretebilen, uçtan uca bir sistem tasarlamak, uygulamak ve değerlendirmektir. Önerilen metodolojide nicel ve nitel verilerin toplanması, LLM tabanlı bir bilgi çıkarım modülü geliştirilmesi, Çoklu-Ajan Sistemi kullanılarak bu verilerin sentezlenip "Nitel Etki Skorları"na dönüştürülmesi ve gelecekteki hisse yönünün geleneksel teknik göstergelerle birlikte tahminine dayalı öngörü modeline entegre edilmesi gösterilecektir. Bu çalışma, finansal olayların daha derin anlamsal bağlamını yansıtan çok yönlü bir nitel sinyal yaratarak basit duygu analizinin ötesine geçmektedir. Bulguların, Büyük Dil Modellerinin karmaşık karar alma ortamlarındaki tam potansiyelinden yararlanmak için şeffaf ve tekrarlanabilir bir çerçeve sunarak finansal tahmin literatürüne katkıda bulunması beklenmektedir.

ABSTRACT

A PROPOSAL FOR AN AGENT-BASED FRAMEWORK STRUCTURE FOR FINANCIAL FORECASTING USING LARGE LANGUAGE MODELS

Predicting financial market movements poses a significant challenge due to the complex, non-linear, and noisy nature of markets, which are influenced by a countless of quantitative and qualitative factors. Traditional forecasting models primarily rely on structured numerical data, but often fail to capture the rich, context-dependent information embedded in unstructured text sources, such as news articles, corporate announcements, and regulatory filings. This paper proposes and develops a novel, hybrid system for stock market prediction by integrating the analytical power of Large Language Models (LLMs) within a Multi-Agent System (MAS) framework. The primary objective of this research is to design, implement, and evaluate an end-to-end system that processes real-time market data and unstructured textual information to generate predictive signals for stocks listed on Borsa Istanbul (BIST). The proposed methodology involves collecting quantitative and qualitative data, the development of an LLM-based information extraction module, and the synthesis of this data into a "Qualitative Impact Scores" through the use of a Multi-Agent System, which is then integrated into a predictive model alongside traditional technical indicators to forecast future price movements. The study moves beyond simple sentiment analysis by creating a multi-faceted qualitative signal that reflects the deeper semantic context of financial events. The findings are expected to contribute to the financial forecasting literature by presenting a sophisticated, transparent, and reproducible framework that demonstrates the full potential of Large Language Models in complex decision-making environments.

ABBREVIATIONS

ABS	Agent-Based System
AI	Artificial Intelligence
API	Application Programming Interface
BIST	Borsa Istanbul
CNN	Convolutional Neural Networks
DL	Deep Learning
ECS	Elastic Cloud Server
EMA	Exponential Mean Average
EMH	Efficient Market Hypothesis
GDP	Gross Domestic Product
JSON	Java Script Object Notation
KAP	Kamuyu Bilgilendirme Platformu (Turkish: Public Disclosure Platform)
KPI	Key Performance Indicator
LLM	Large Language Model
MACD	Moving Average Convergence/Divergence
MAS	Multi-Agent System
ML	Machine Learning
NLP	Natural Language Processing
RNN	Recurrent Neural Networks
RSI	Relative Strength Index
RSS	Really Simple Syndication
SQL	Structured Query Language
VIOP	Vadeli İşlem ve Opsiyon Piyasası (Turkish: Futures and Option Market)
VWAP	Volume Weighted Average Price

LIST OF FIGURES

Figure 7.1. Data Collection and Preprocessing Module 18
Figure 7.2. Data Storage Module 19
Figure 7.3. Generic Agent Structure 21
Figure 7.4. Multi-Agent System 23
Figure 7.5. Hybrid Prediction Model 26
Figure 8.1. Messages Stored Alert in Grafana 31
Figure 8.2. Strategy Profit&Loss Comparison 33

LIST OF TABLES

Table 8.1. BIST30 Stocks 31
Table 8.2. Analysis Results 32
Table 8.3. Fusion Agent Scores 35

1. INTRODUCTION

The effort to predict future movements of financial markets is one of the most fundamental and difficult problems in both academic research and the finance industry. Markets are inherently complex and dynamic systems shaped by the interaction of numerous macroeconomic, political, firm-specific, and psychological factors. Traditional forecasting models have tried to address this complexity largely by focusing on structured numerical data (prices, volume, balance-sheet items, etc.). However, with increasing digitization and the adoption of AI models, it has become clear that large streams of textual data such as news articles, analyst reports, social media feeds, and official company announcements, form some of the most influential inputs to decision making. Making sense of this flood of unstructured data and extracting the latent predictive value within it has become one of the biggest challenges and opportunities in modern financial analysis.

Natural Language Processing (NLP) and its most advanced application, Large Language Models (LLMs), offer revolutionary tools to address this challenge. Models such as ChatGPT or LLaMA do not only process text by keywords; they can also understand a sentence's context, nuance, and underlying sentiment. This ability allows analysis of a financial news item's potential impact on stocks or other risky assets (direction, magnitude, time horizon, etc.) with near human-like accuracy. However, financial analysis is not limited to interpreting a single news item; it is a multi-layered process that requires combining different areas of expertise (macroeconomics, sector analysis, company analysis). To simulate this complex decision-making process, this study makes use of the Multi-Agent Systems (MAS) paradigm, which is composed of autonomous software units, each focused on a specific task.

The main objective of this paper is to develop and evaluate an original, hybrid forecasting system for stocks traded on Borsa Istanbul (BIST) that integrates real-time market data, live news streams, and modern AI techniques. To achieve this goal, the Denizbank Algolab platform application will be used to monitor the Borsa Istanbul stock market live. In parallel, textual data will be collected from the Public Disclosure Platform (KAP) and other financial news sources; these texts will be processed by Large Language Models and transformed into structured, meaningful, and quantitative formats. Then, LLM-based intelligent agents will analyze these structured data to produce the final inputs for a mathematical decision mechanism. For converting this input into buy-sell signals, a primary linear regression-based model will be developed.

To meet this objective, the thesis is structured as follows. Following this introductory chapter, the theoretical infrastructure and literature review are established across several chapters. Chapter 2 presents the fundamentals of financial market forecasting. Subsequently, the core technologies underpinning our approach are detailed: Chapter 3 covers Natural Language Processing, Chapter 4 discusses Machine Learning and Deep Learning models, Chapter 5 focuses on Large Language Models, and Chapter 6 explores Agent-Based Systems. In Chapter 7, the system architecture and methodology are detailed, covering all steps from data collection to the design of the agents and the development of the prediction engine. Chapter 8 presents the empirical findings and performance comparisons obtained from testing the system. Finally, in Chapter 9, a general evaluation of the study is made, the results are discussed, and recommendations for future work are offered.

2. FINANCIAL MARKET PREDICTION

The effort to predict the future path of financial asset prices is one of the most fundamental and difficult problems in both academic finance literature and practical investment (Darwish et al., 2025). Financial markets can be described as dynamic and complex adaptive systems where countless macroeconomic indicators, political developments, firm-specific news, and most importantly the rational and irrational expectations of millions of investors come together (Potters et al., 1997). The difficulty at the heart of the forecasting problem is to separate the noise which consists of meaningless and random price fluctuations, from the signal, which is meaningful information that points to future trends. Despite this difficulty, the economic gains from a successful forecast continue to drive uninterrupted research and model-development efforts in this field (Cheng et al., 2022).

2.1. Efficient Markets Hypothesis and the Predictability Paradox

Theoretical debates about the predictability of financial markets are rooted in the Efficient Market Hypothesis (EMH) formulated by Eugene Fama (1965). EMH states that a security's market price fully and instantaneously reflects all available information. If markets are efficient, new information is immediately incorporated into prices, making it impossible for investors to consistently earn abnormal returns. EMH is commonly examined in three forms: weak, semi-strong, and strong, which differ according to the range of information taken into account (G. Cairns, 2014; B. G. Malkiel, 1991).

- **Weak form:** Current stock prices already reflect all historical market information (past prices, trading volumes, etc.). If this assumption holds true, technical analysis that relies solely on past price patterns and statistics should not yield systematic profits. There is no guarantee that a model from the past will work in the future because this information is already included in the price.
- **Semi-strong form:** Prices incorporate not only past market data but all publicly available information (company disclosures to KAP, annual and quarterly reports, dividend announcements, macroeconomic releases, analyst reports, news). In this case, fundamental analysis methods that attempt to determine a company's "intrinsic value" based on publicly available information would also fail to provide abnormal returns, as the market would have already priced this information at the moment a news item is released.

- **Strong form:** This form represents the most stringent version of the hypothesis, asserting that prices reflect all information, both public and private, including "insider information" that has leaked from within the company. This strongest version is difficult to hold in practice due to legal controls and information asymmetries, but it represents a theoretical upper bound.

2.2. The Rise of Behavioral Finance and Market Anomalies

Despite EMH's theoretical elegance and central role in finance, empirical studies and historical episodes show that markets are not always fully efficient. Events such as the 2000 Dot-com bubble and the 2008 Global Financial Crisis demonstrate that prices can deviate substantially from rational expectations (Ball, 2009; B. Malkiel, 2011). These deviations are referred to as market anomalies and contradict the assumptions of the EMH. At this point, the field of Behavioral Finance has emerged, which examines the impact of human psychology on financial decisions. Pioneering work by Kahneman and Tversky (1979) showed that investors are not always rational agents but exhibit systematic cognitive biases. Biases such as overconfidence, herd behavior, loss aversion, and anchoring can cause investors to overreact or underreact to information, make irrational trading decisions, and thus push prices away from their fundamental values (Griffin & Tversky, 1992; Masood, 2024). These behavior-driven inefficiencies create the conditions for predictable patterns in markets, which this study targets as its primary opportunity window.

2.3. Traditional Financial Forecasting Approaches and Their Limitations

Over time, various analysis and forecasting methodologies have been developed to exploit market inefficiencies; these methods fall broadly into two categories: Fundamental and Technical Analysis (Nti et al., 2020).

- **Fundamental Analysis:** Focuses on determining a financial asset's intrinsic value (Gottwald, 1913). Its central assumption is that a company's stock price will converge toward its economic value in the long run. Fundamental analysts examine macroeconomic variables (GDP, inflation, interest rates), industry dynamics (competition, growth potential), and especially firm-level data (balance sheet, income statement, cash flows, leverage ratios, profitability margins) in detail (Nti et al., 2020). If the computed intrinsic value exceeds the current market price, the stock is considered

undervalued (a buy); if lower, overvalued (a sell). Benjamin Graham and David Dodd's Security Analysis is often viewed as the manifesto of this school (Griffioen, 2003).

- **Technical Analysis:** This approach does not attempt to estimate intrinsic value. Instead, based on the principle that "market discounts everything" it uses past price and volume data to forecast future price movements (Griffioen, 2003; Nti et al., 2020). Technical analysts identify chart patterns (e.g., head-and-shoulders, double bottoms/tops) and apply indicators (moving averages, RSI, MACD, etc.) to detect trends, support, and resistance (Kirkpatrick & Dahlquist, 2010; Schwager, 1995). The underlying assumption is that investor psychology produces recurring patterns, and that history tends to repeat itself.

2.4. Beyond Traditional Methods: The Importance of Unstructured Data

Traditional forecasting approaches share a reliance on structured numerical data (prices, volumes, financial statement items). However, 21st-century financial markets are defined by an explosion of information due to digitalization. Massive volumes of unstructured textual data such as public disclosures, national and international financial news, central bank speeches, analyst reports, and social media discussions have the power to shape market sentiment and expectations in real time. A company's investment announcement, a new regulation affecting a sector, or a geopolitical event can cause sudden, large price moves that traditional models may fail to explain (Gandhmal & Kumar, 2019).

Manually extracting the causal relations, context, sentiment, and intent embedded in this textual data is infeasible. The inability of conventional models to process this critical information source creates a major gap in modern financial forecasting (Darwish et al., 2025; Divate et al., 2024). This gap motivates the use of advanced AI techniques such as Natural Language Processing (NLP), Large Language Models (LLMs), and Agent-Based Systems (González et al., 2018) where these models are structured as autonomous decision-makers, both necessary and highly valuable in this study. These technologies can capture semantic depth in text, convert it into quantitative signals, and thus feed traditional forecasting models to produce timelier and more accurate predictions.

3. NATURAL LANGUAGE PROCESSING (NLP)

In the previous section, it was noted that traditional econometric and statistical models used to forecast financial markets rely largely on structured numerical data (prices, volumes, balance-sheet items, etc.), which constitutes a significant limitation. In contrast, one of the most critical factors shaping market dynamics is the continuous flow of unstructured textual data. Mandatory disclosures made by companies through the Public Disclosure Platform (KAP), articles published by financial news agencies, central bank policy texts, analyst reports, and investor discussions on social media together form a vast information pool that instantly affects market sentiment and expectations (Kearney & Liu, 2014). The rich information embedded in this textual data such as causality, context, sentiment, and intent, exceeds human capacity for manual analysis.

At this point, Natural Language Processing (NLP), which sits at the intersection of computer science and artificial intelligence, emerges as an indispensable technology for financial analysis. Broadly defined, NLP is the set of theories, algorithms, and methodologies that enable computers to understand, interpret, process, and generate human language. In finance, NLP aims to automatically process massive text corpora to extract subjective or objective information, classify and summarize it, and ultimately convert it into quantitative, structured signals that can support investment decisions. This section will review the main NLP techniques used in financial text analysis, their evolution, and their limitations in light of foundational works in the literature.

3.1. Text Preprocessing

The field of natural language processing (NLP) has evolved significantly over time under the influence of theoretical innovations, increasing data volumes, and advances in computational resources. In the early period, research focused on rule-based approaches. Grammar-based parsers developed from the 1950s to the 1970s used handcrafted rules and grammars to analyze texts' syntactic structure and produce parse trees (Chomsky, 1957). In these methods, lexicons were commonly applied to identify and extract specific pieces of information from text, such as names, dates, and locations. However, such systems suffered from heavy dependence on extensive handcrafted rules and dictionaries and were unable to handle exceptions and outliers in text. ELIZA, an early chatbot example, relied on the era's parsing and pattern-matching techniques (Lo et al., 2023; Weizenbaum, 1966).

The 1980s and 1990s witnessed the rise of statistical NLP models. Probabilistic methods such as Hidden Markov Models and Maximum Entropy Models were used to model sequential data, learn patterns in text, and make predictions (Berger et al., 1996). Although these models were more flexible than rule-based approaches, they faced challenges in computational efficiency and modeling complex relationships (Lo et al., 2023).

In the late 1990s and early 2000s, hybrid approaches combining lexicon-based and statistical methods emerged (Lo et al., 2023). These approaches analyzed texts using predefined or data-derived word lists. At the same time, topic modeling techniques began to be applied to statistically detect themes in document collections (Blei et al., 2003; Lo et al., 2023). To improve performance in finance, the financial sentiment lexicon developed by Loughran and McDonald (2011) became widely used for classifying sentiment in financial texts (Mishev et al., 2020).

The 2000s and 2010s marked the rise of embedding-based NLP techniques. Word embeddings represented words as high-dimensional continuous-valued vectors, capturing both semantic and syntactic information (Patil et al., 2023). In this period, numerical representations such as term frequency–inverse document frequency (TF–IDF), n-grams, and bag-of-words (Patil et al., 2023) were used alongside predictive methods like Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). These approaches enabled more effective modeling of relationships between words.

In recent years, driven by exponential growth in data and computational infrastructure, large deep learning-based language models have been developed and widely adopted in NLP applications. Initially, Recurrent Neural Networks (RNNs) were developed for sequential data analysis and adapted for NLP tasks; similarly, Convolutional Neural Networks (CNNs) were adopted as an effective approach for text classification (Kim, 2014; Lo et al., 2023). The revolutionary shift in NLP began with the introduction of BERT (Devlin et al., 2019) and the widespread use of the self-attention mechanism in transformer architectures (Lo et al., 2023). During this period, powerful models such as GPT-2 (Radford et al., 2019), RoBERTa (Liu et al., 2019), XLNet (Z. Yang et al., 2020), ALBERT (Lan et al., 2020), DistilBERT (Sanh et al., 2020), XLM (Lample & Conneau, 2019), and BART (Lewis et al., 2019) were developed.

Among domain-specific versions, FinBERT (Y. Yang et al., 2020) stands out for financial text analysis.

In conclusion, traditional NLP methods provided a strong foundation for extracting information from financial texts. However, their inability to capture word meanings that vary by context and to fully model language's complex semantic and syntactic structures pushed research toward the next stage: context-centered Deep Learning Models and Large Language Models. This topic will be discussed in detail in the following sections.

4. MACHINE LEARNING AND DEEP LEARNING MODELS

The previous sections have detailed how unstructured text data, a fundamental factor influencing financial markets, is processed using Natural Language Processing (NLP) techniques and converted into meaningful, structured features. After the text data is transformed into quantitative formats, there is a requirement for powerful algorithmic models that can produce forward-looking predictions by integrating this new source of information with traditional numerical market data (such as price, volume, technical indicators, etc.).

At this point, Machine Learning (ML) and its more sophisticated subfield, Deep Learning (DL), come into play, possessing the ability to autonomously learn patterns, relationships, and complex dynamics from data instead of relying on predefined rigid rules or equations.

Machine learning, defined by Arthur Samuel (1959) as the field of giving computers the ability to learn without being explicitly programmed, offers greater flexibility and modelling power for the complex, noisy, dynamic, and nonlinear relationships inherent in financial markets compared with traditional statistical models (Guresen et al., 2011). Financial data often suffer from low signal-to-noise ratios and non-stationarity, which makes it essential for ML/DL models to effectively handle these challenges. This section will examine the main ML and DL models commonly used in financial forecasting literature, along with their theoretical foundations, strengths, and limitations.

4.1. Traditional Machine Learning Models and Their Role in Financial Prediction

Before the widespread adoption of deep learning, various machine learning algorithms with "shallow" architectures were successfully applied to predict the direction or value of financial asset prices. These models generally had advantages such as requiring less data, being trainable with lower computational power, and providing more interpretable model decisions.

Linear models, despite being among the most basic machine learning models, are still considered strong baselines because of their simplicity and interpretability. Among these, linear regression is one of the simplest forecasting methods and one of the earliest used (Fama, 1965). Later, Support Vector Machines (SVM) (Cortes & Vapnik, 1995), decision trees (Ryzin et al., 1986) and ensemble methods gained popularity for their generalization power. In particular, random forest offers strong and stable classification/regression performance by combining many decision trees in a randomized way (Breiman, 2001). Scalable gradient-boosting

algorithms like XGBoost have been widely adopted both in competition settings and in practical systems (T. Chen & Guestrin, 2016).

4.2. Deep Learning

Deep Learning is a subfield of machine learning that learns complex and hierarchical representations from data using computational models composed of many processing layers (LeCun et al., 2015). The rise of deep learning has significantly expanded the capacity to model the complex, nonlinear relationships in financial data. Especially on large datasets and difficult problems, increasing the number of layers allows learning more abstract and complex features, producing breakthrough performance compared to traditional models. Special deep learning architectures have been developed for sequential data analysis such as financial time series (Sezer et al., 2019). Recurrent Neural Networks (RNNs), unlike traditional feed-forward neural networks, include recurrent connections in the network (Sezer et al., 2019). This recurrent structure allows the output at one time step to be fed as input to the next time step. This property makes RNNs a natural choice for inherently sequential data like language modeling and time series forecasting. However, standard RNNs make learning long-term dependencies nearly impossible when working with long sequences (for example, the effect on today's price of an event weeks or months ago) (Bengio et al., 1994). Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) was developed to solve this critical long-term memory problem of RNNs; it is a specialized and very successful RNN architecture that effectively captures both short and long-term temporal dependencies (LeCun et al., 2015; Sezer et al., 2019). This superior ability has made LSTM one of the most widely used and successful deep learning models in financial time series forecasting, and it has been shown to outperform traditional machine learning models (Fischer & Krauss, 2018; Nelson et al., 2017). As an alternative to LSTM, Gated Recurrent Units (GRU) were developed as an RNN architecture with a simpler gating mechanism (Cho et al., 2014). Having fewer parameters, GRU is often faster to train than LSTM and in some cases, especially with smaller datasets, shows similar or better performance with less tendency to overfit (Cho et al., 2014; Chung et al., 2014). Along with RNNs, Convolutional Neural Networks (CNNs), although originally developed to capture local features in computer vision, have also produced successful results in financial time series analysis (Kim, 2014). The revolution in NLP came later with transformer and self-attention architectures and Large Language Models trained on very large data (Devlin et al., 2019; Vaswani et al., 2017).

5. LARGE LANGUAGE MODELS

The traditional natural language processing (NLP) and machine learning methods reviewed earlier have provided important progress in financial text analysis. Frequency-based models like TF–IDF and static word embedding techniques such as Word2Vec or GloVe were effective at converting texts into machine-readable numeric formats. Similarly, recurrent neural networks like LSTM and GRU could capture context to some degree by modelling the sequential nature of text. However, all these models have limitations when it comes to fully resolving one of language's most fundamental and complex features: dynamic context. For example, static word embedding models cannot distinguish between different meanings of the word *capital*, such as "financial assets" or "city of a country", while RNN-based models struggle to establish distant semantic relations in very long texts. These constraints created the need for methods that can learn context-sensitive representations of language and model long-term dependencies more effectively.

5.1. Transformer Architecture: Parallelism and the Self-Attention Mechanism

With the emergence of the transformer architecture, models using the self-attention mechanism were developed that can deeply contextualize text and preserve semantic relations over much longer inputs (Vaswani et al., 2017). Scaling this architecture gave rise to Large Language Models (LLMs), which are notable not only for detecting semantic relations in financial texts but also for their ability to transfer knowledge across tasks (transfer learning). These models are deep neural networks with billions or even trillions of parameters, pre-trained on massive, internet-scale corpora. The success of the transformer led to different LLM families optimized for various purposes.

For example, BERT learns context from both directions (left-to-right and right-to-left) simultaneously. This bidirectional learning makes BERT highly successful at extracting deep semantic representations of text. In finance, domain-adapted models such as FinBERT, created by fine-tuning BERT on financial texts (analyst reports, news, etc.), achieved state-of-the-art results in financial sentiment analysis and text classification (Araci, 2019; Y. Yang et al., 2020). The GPT family from OpenAI uses only the decoder stack and is optimized for text generation (Brown et al., 2020; Radford et al., 2018, 2019). Especially with GPT-3 and later models (e.g., GPT-4), it was discovered that these models can perform complex novel tasks with very few examples (few-shot) or even without examples (zero-shot) (Brown et al., 2020; OpenAI et al., 2024). This capability, called in-context learning, fundamentally changed the flexibility of

LLMs. They can also convert raw text directly into actionable structured formats such as JSON (Huang et al., 2025). These features are a major advantage in domains like finance, where labelled data is scarce and expensive. Consequently, LLMs enable forecasting systems that can respond quickly to sudden market events (e.g., unexpected interest rate decisions, geopolitical crises). Multimodal prediction models that combine contextual vector representations from news texts with numerical market data further broaden LLM applications in finance (Huang et al., 2025).

However, using LLMs also brings challenges. The factual accuracy of their outputs is not always guaranteed, and they have a tendency to fabricate information, a phenomenon known as "hallucination" (Ji et al., 2023). Therefore, verifying LLM outputs and using them cautiously in financial applications is critically important. This study aims to investigate how the powerful capabilities of LLMs can be harnessed within an agent-based system, discussed in the next section, to build a reliable and autonomous structure for financial forecasting.

6. AGENT BASED SYSTEMS

The previous sections reviewed NLP techniques used to extract meaning from financial text data and the machine learning models that produce forecasts using that information. In this context, especially Large Language Models (LLMs) have introduced revolutionary capabilities in text understanding, reasoning, and information extraction, causing a significant transformation in financial analysis workflows. However, by themselves these models can be limited in planning autonomous actions toward specific goals, adapting to environmental changes, and working integratively with different data and tool sources. At this point, Agent-Based Systems (ABS) and their multi-agent variant, Multi-Agent Systems (MAS), provide a strong and flexible theoretical framework for modern AI architectures by combining the linguistic intelligence of LLMs with autonomous decision-making and interaction capabilities.

An agent, in the broadest sense, is a system that can perceive its environment, adapt to changing conditions, and set and pursue goals autonomously (Russell & Norvig, 2003). In the context of software agents, an agent is an autonomous computer program that exhibits proactive and reactive behaviors to reach its goals. This section will examine the basic properties and architectures of agent-based systems and, especially considering the rise of LLMs, how they form a new and powerful paradigm to solve complex tasks such as financial forecasting.

6.1. Fundamental Characteristics and Architectures of Agents

To be qualified as an “agent”, a software piece generally should have certain properties. Commonly accepted core traits in the literature are autonomy, social ability, reactivity, and proactivity (C. Gao et al., 2023; Wooldridge & Jennings, 1995). Agents are required to be autonomous, implying that they can function independently of human intervention or external system control while maintaining authority over their internal states and actions. Agents are expected to demonstrate social capabilities by interacting, cooperating, and negotiating with one another. They must be able to perceive their environment, such as fluctuations in market data or the emergence of new information, and respond promptly to these changes. In addition to reacting to the environment agents should also act proactively, exhibiting goal-directed and opportunistic behavior in pursuit of their objectives.

6.2. Multi-Agent Systems (MAS): Power of Specialization and Collaboration

Problems that are too large or complex for a single agent are usually solved by Multi-Agent Systems. MAS is a system of autonomous agents interacting to achieve a common goal or to fulfil their individual goals (Jennings et al., 1998; Weiss, 1999). The MAS paradigm's basic philosophy is "divide and conquer": a complex problem is split into smaller, manageable sub-tasks, and each sub-task is assigned to an agent specialized in that area. In financial analysis this approach has large potential. Thus, a general and monolithic "financial analysis" task can be decomposed into sub-tasks assigned to expert agents that use different data sources and analysis methods, enabling realistic simulation of market responses (Hommes, 2005). Such a modular design provides significant flexibility and scalability because adding a new analysis type (for example, social-media sentiment analysis) can be done by introducing a new expert agent without breaking the existing system. Moreover, overall system robustness increases: if one agent fails for any reason, other agents can continue their tasks and the problem can still be at least partially solved.

6.3. LLM-Based Agents: Large Language Models as Reasoning Engines

In traditional MAS-based financial modelling work, agents' decision mechanisms were often limited to rule-based or simple statistical models. The rise of LLMs opened a new paradigm: these models can now be used not only as text processors but as reasoning engines (J. C.-Y. Chen et al., 2024). An LLM-based agent can autonomously determine, in natural language and via reasoning, which steps to take to reach a goal, which tools to use, and how to interpret its results (S. Gao et al., 2024). For these reasons, LLM-agent based financial forecasting systems have been developed (Li et al., 2024; Xiao et al., 2025).

This study uses that innovative agent paradigm to develop a multi-agent system composed of LLM-based agents with different expertise (macroeconomics, sector, fundamental analysis, technical analysis) and aims to investigate whether this system produces more accurate, explainable, and robust results in financial forecasting compared to traditional monolithic approaches.

7. METHODOLOGY AND APPLICATION

In the preceding chapters the challenges in predicting financial markets, the limitations of traditional approaches, and the potential offered by technologies such as Natural Language Processing (NLP), Machine Learning (ML), and Deep Learning (DL) to overcome these limitations have been investigated in a comprehensive manner. Especially, the revolutionary capabilities of Large Language Models (LLM) in understanding, reasoning, and extracting information from text, and the modular structure of Agent-Based Systems (ABS) that provides autonomy, specialization, and cooperation, present a new and powerful paradigm for financial analysis.

The theoretical framework presented in the literature review forms the foundation of this study. However, for this theoretical potential to be transformed into a practical value, it is necessary that these different technologies are integrated within an end-to-end system architecture that works in a compatible way with each other. This chapter presents the methodological roadmap detailing how the study was conducted. Our aim is not only to test the performance of individual models, but also to develop a holistic system that includes data collection and processing, LLM-based qualitative analysis, multi-agent-based decision synthesis, and finally the creation of a hybrid prediction model.

First, this section introduces the overall architecture of the proposed "Agent-Based Financial Prediction System Supported by Large Language Models" and the data flows between its components. Then all modules and subsystems that compose the system are described both at a high level and in detail. For each component, the specific algorithms, data sources, engineering approaches, and technological tools to be used are specified. This systematic and modular approach aims to ensure the reproducibility of the project and to evaluate the effectiveness of the proposed system in an objective and measurable way.

7.1. Data Collection and Storage

The success of the Agent-Based Financial Forecasting System to be developed depends primarily on the quality, timeliness, and coverage of the data it is fed. In order to be able to analyze the market dynamics in a holistic manner, the system requires two main types of data sources: both quantitative (numerical) and qualitative (textual). This section describes the technical details of the high-performance, continuously available data collection and storage

infrastructure designed to provide these data flows, the technologies used, and the rationale behind those choices. The system's data backbone consists of real-time market data from Borsa Istanbul (BIST), disclosures from the Public Disclosure Platform (KAP), and general financial news feeds. For all these operations to be conducted in an uninterrupted and reliable manner, a remote server infrastructure located on Huawei Cloud has been preferred.

7.2. Quantitative Data Stream: Borsa Istanbul Real-Time Price Data

The most fundamental input for financial prediction models is the instant price, order book, and volume data which reflect the supply and demand dynamics in the market. This study aims to collect real-time and high-frequency data for stocks traded on Borsa Istanbul, particularly those included in the BIST30 index. To minimize latency and ensure data integrity during market data collection, the WebSocket API provided by the Denizbank Algolab platform was preferred. Contrary to the traditional HTTP request-response model, the WebSocket protocol establishes a persistent, bidirectional communication channel between the server and the client. By this means, every price or volume change (tick data) that occurs in the market is instantly "pushed" to the client by the server. This approach guarantees the data timeliness, which is critical especially for short-term analyses, by enabling the system to react to market events in almost real-time. For the development of the data collection client, the Rust programming language, which stands out with its memory safety, high performance, and concurrency capabilities, was chosen. Financial data streams are high-volume and critically important data that can contain thousands of messages per second. Rust's "zero-cost abstractions" and its ownership model, which guarantees memory safety at compile-time, eliminate risks such as memory leaks or undefined behavior that could arise in such a system. By using asynchronous runtime libraries like tokio, it has been made possible to efficiently manage numerous WebSocket connections on a single thread and to process the incoming data stream without suffering performance loss. This client developed with Rust will parse the incoming raw WebSocket messages, convert them into the relevant OHLCV (Open, High, Low, Close, Volume) data, and transmit them to the database for storage.

7.3. Qualitative Data Stream: KAP Disclosures and General Financial News

The foundation of the system's qualitative analysis capability is constituted by textual data about companies and markets. For this purpose, data will be collected periodically from two main sources. The Public Disclosure Platform (KAP) is the official platform in Türkiye where publicly-held companies are obligated to announce to the public all significant developments

that could affect investors (financial reports, capital increases, major contracts, management changes, etc.). These notifications are the most important sources of information with high reliability and are directly related to the company. In this study, a program will be developed to pull KAP notifications and news from the internet. A program, scheduled to run at certain intervals (for example, every minute), will visit the latest notifications or relevant company pages on KAP, and will pull the titles of new notifications, their publication dates, the related companies, and the full content of the notification text. By checking the unique ID of each pulled notification, the making of duplicate records in the database will be prevented. While KAP notifications are company-focused, there is a need for general news sources to capture more comprehensive information such as macroeconomic developments, sector trends, geopolitical events, and general market sentiment that affect the markets. In this study, the RSS feeds of some news channels operating in Türkiye were determined as the news source due to their ease of implementation and the wide scope they offer. By using RSS feeds, the most current news headlines, summaries, sources, and links will be periodically pulled. This method simplifies the development process as it offers access to the news of different sources through RSS feeds that have a certain standard, instead of developing separate web scrapers for dozens of individual news sites. This module, which includes the steps of collecting quantitative and qualitative data from different sources and converting them into a standard format, is visualized in Figure 7.1.

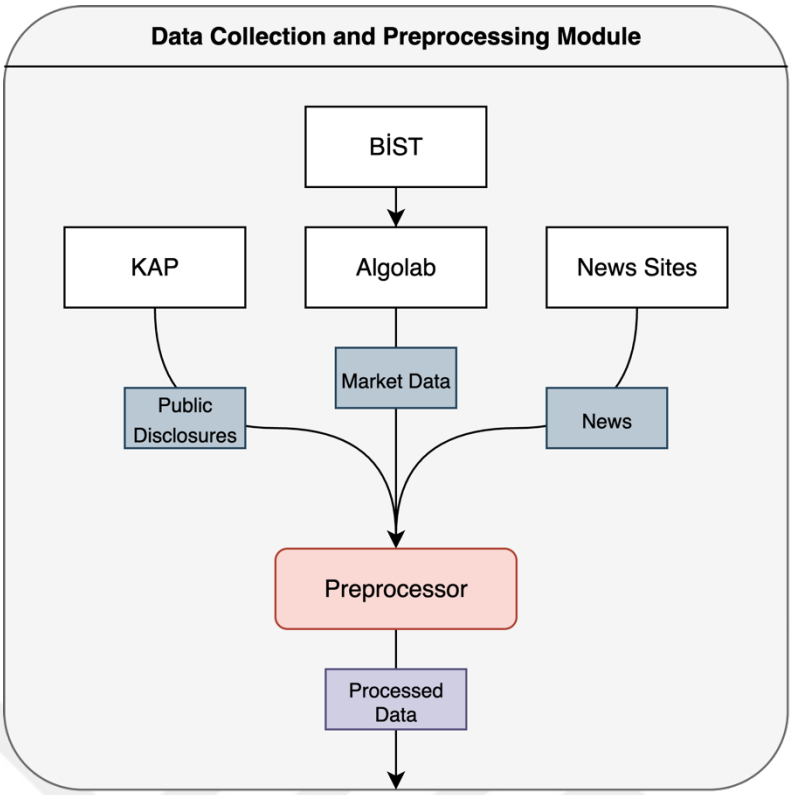

Figure 7.1. Data Collection and Preprocessing Module

7.4. Data Storage and Infrastructure Architecture

The storage of all these different types of collected data in a manner that is consistent, reliable, and can be queried with high performance is critical for the overall performance of the system. As the data storage solution, TimescaleDB, which is an open-source time-series database that runs as an extension on PostgreSQL, has been preferred. Since TimescaleDB automatically partitions data according to the timestamp into smaller chunks called hypertables, it enables queries based on time intervals to run much faster compared to standard PostgreSQL tables. As it is a PostgreSQL extension, it benefits from the full power and flexibility of the SQL language. The database schema will be designed in such a way as to associate the quantitative and qualitative data. For example, by creating a '*stocks*' table and a '*news_data*' table which both have '*timestamp*' columns, it will be possible to easily query what the price movements were at the moment the news was published. This data flow is illustrated in Figure 7.2.

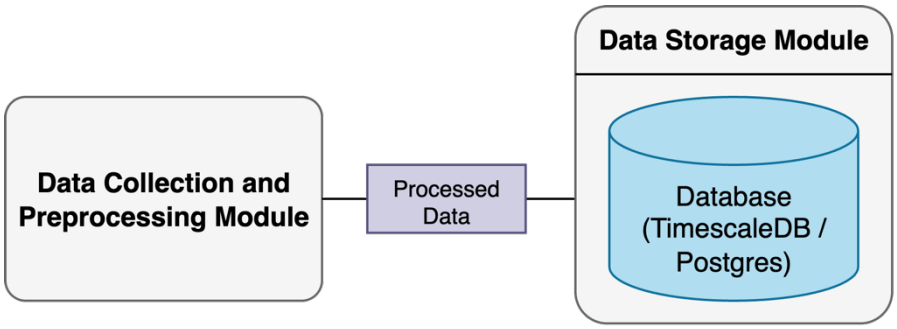

Figure 7.2. Data Storage Module

For all this data collection software and the database to be able to operate 24/7 without interruption, it was decided to use a cloud computing infrastructure instead of a physical local machine. For this purpose, considering the advantages of geographical proximity and local support, a virtual server (Elastic Cloud Server, ECS) will be configured on Huawei Cloud's data center in Istanbul. This approach will guarantee that the system reliably fulfills its data collection task by providing high availability, scalability, and fault tolerance against hardware failures.

7.5. System Monitoring and Operational Reliability

The continuous and reliable operation of the data collection, analysis, and storage components is essential to the integrity of the entire research system. To ensure operational stability and provide real-time insight into the system's health, a dedicated monitoring and alerting infrastructure has been implemented. This infrastructure is built upon two industry-standard open-source tools: Prometheus for metrics collection and Grafana for visualization and dashboarding.

- **Metrics Collection with Prometheus:** Each component within the system architecture including the Algolab client, the scripts for KAP and news scraping, and the LLM interaction module is instrumented to expose key performance indicators (KPIs) as time-series metrics. These metrics include data processing rates (e.g., ticks per second), API response latencies, database query performance, and application error rates. Prometheus periodically scrapes these metrics and stores them for later inspection.
- **Visualization and Alerting with Grafana and Telegram:** Grafana is connected to the Prometheus data source to create a comprehensive operational dashboard, providing a centralized, visual overview of the entire system's status in real-time. To move from passive monitoring to proactive incident response, alert rules are defined within

Grafana. These rules trigger alerts if critical thresholds are breached. For instance, if the WebSocket connection to the data provider is lost, if the rate of KAP notifications drops to zero for an extended period, if the LLM API error rate exceeds a predefined limit or, significant money loss happened in a period alerts trigger. Upon triggering an alert, a notification is sent directly to a designated Telegram channel. This mechanism ensures that any potential system failures are immediately communicated, allowing for rapid intervention and minimizing data loss or system downtime.

7.6. LLM-Based Information Extraction Module: Generating Meaning from Unstructured Text

The data collection infrastructure described in the previous section continuously supplies raw textual data to the system (KAP disclosures and financial news). However, these texts are unstructured and not directly interpretable by machine learning models. The primary goal of this module is to transform the raw text into an analyzable, consistent, and information-rich structured format (JSON) by leveraging the advanced comprehension and reasoning capabilities of Large Language Models (LLMs). This process means "extracting" the hidden information in the text and converting it into a meaningful input for the subsequent agent-based analysis stage. For this critical task, the DeepSeek model was chosen as a balance between performance and cost effectiveness; optionally, locally runnable open LLMs such as LLaMA may be used.

7.7. Prompt Engineering: Turning an LLM into a Financial Analyst

The effectiveness of LLMs depends largely on the quality and clarity of the instructions (prompts) they receive. "Prompt engineering" is the craft of designing and optimizing the instruction text to obtain accurate and consistent outputs from an LLM. In this study, a carefully structured prompt template will be developed to analyze each news item and extract information according to a predefined JSON schema. First, the model will be assigned a role and identity, so it operates in the desired context. For example: "You are a senior financial analyst specialized in Borsa Istanbul. Your task is to analyze the text below and extract key information required for an investor's decision in a structured JSON format." Next, the task description and the output schema will be shown to the model, and each field in the schema will be explained in detail (what it means and how to fill it). Finally, the text to be processed will be appended to the end of the template.

7.8. Improving Output Quality: Few-Shot Learning Approach

Although LLMs are successful in "zero-shot" (without seeing any examples) tasks, their performance increases significantly when they are supported with a few examples (few-shot). In order to make the model's output more consistent and accurate, 2-3 example input-output pairs will be added to the prompt template. This helps the LLM to better understand the nuances of the task. This approach, by enabling the model to "align with" the desired output format and reasoning style, will increase the reliability of the project's most critical module. The structure of a generic LLM-based agent, which is fed with sample data and a system prompt and converts the qualitative data into a structured JSON output, is visualized in Figure 7.3.

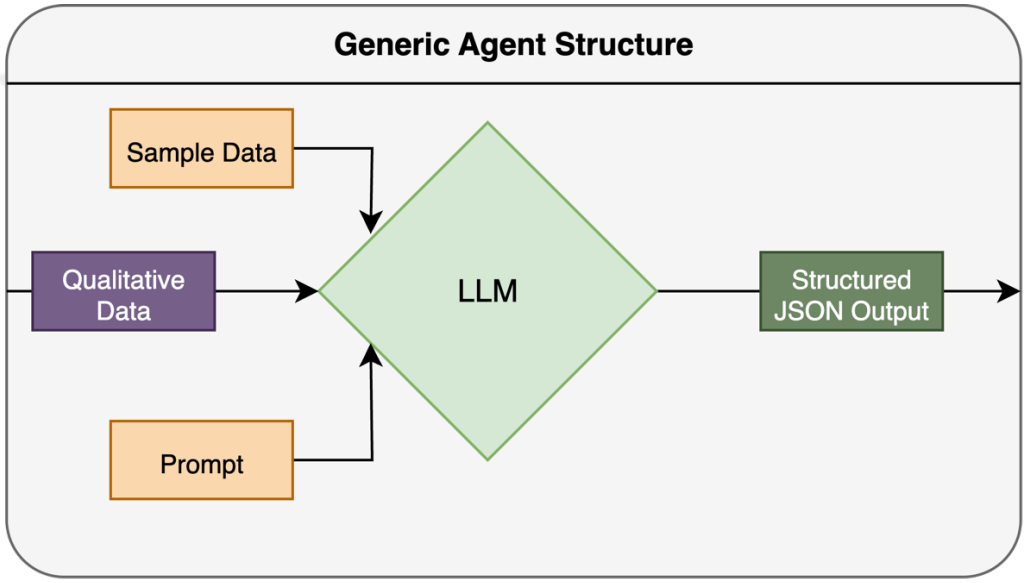

Figure 7.3. Generic Agent Structure

7.9. Multi-Agent System: Distributed Analysis and Decision Fusion

The previous module has taken a critical first step by transforming the raw text data into a rich, structured JSON object by means of the Large Language Model (LLM). However, this JSON object is still raw information that melts the different dimensions of a news item (company-specific impact, sectoral impact, macroeconomic context, etc.) in a single pot. A complex investment decision requires the separate evaluation of these different dimensions and their subsequent combination by passing them through a logical filter. In this study, to perform this complex analysis and synthesis task, a modular approach inspired by the Multi-Agent System (MAS) paradigm has been adopted. This module has been designed as a structure that simulates the fundamental philosophy of the agent paradigm, that is, task distribution, specialization, and

cooperation. The aim, contrary to a single monolithic rule set, is to create autonomous logic units (agents) representing the different layers of the analysis and to combine the outputs of these units for a final decision. This approach makes the logic of the system more modular, understandable, and developable in the future. The system consists of five logical components: an Orchestrator Agent, three different Expert Analyst Agents, and a Fusion Agent. The architectural structure of this Multi-Agent System, which is the most critical component of the developed system, and the interaction between its components are shown schematically in Figure 7.4.

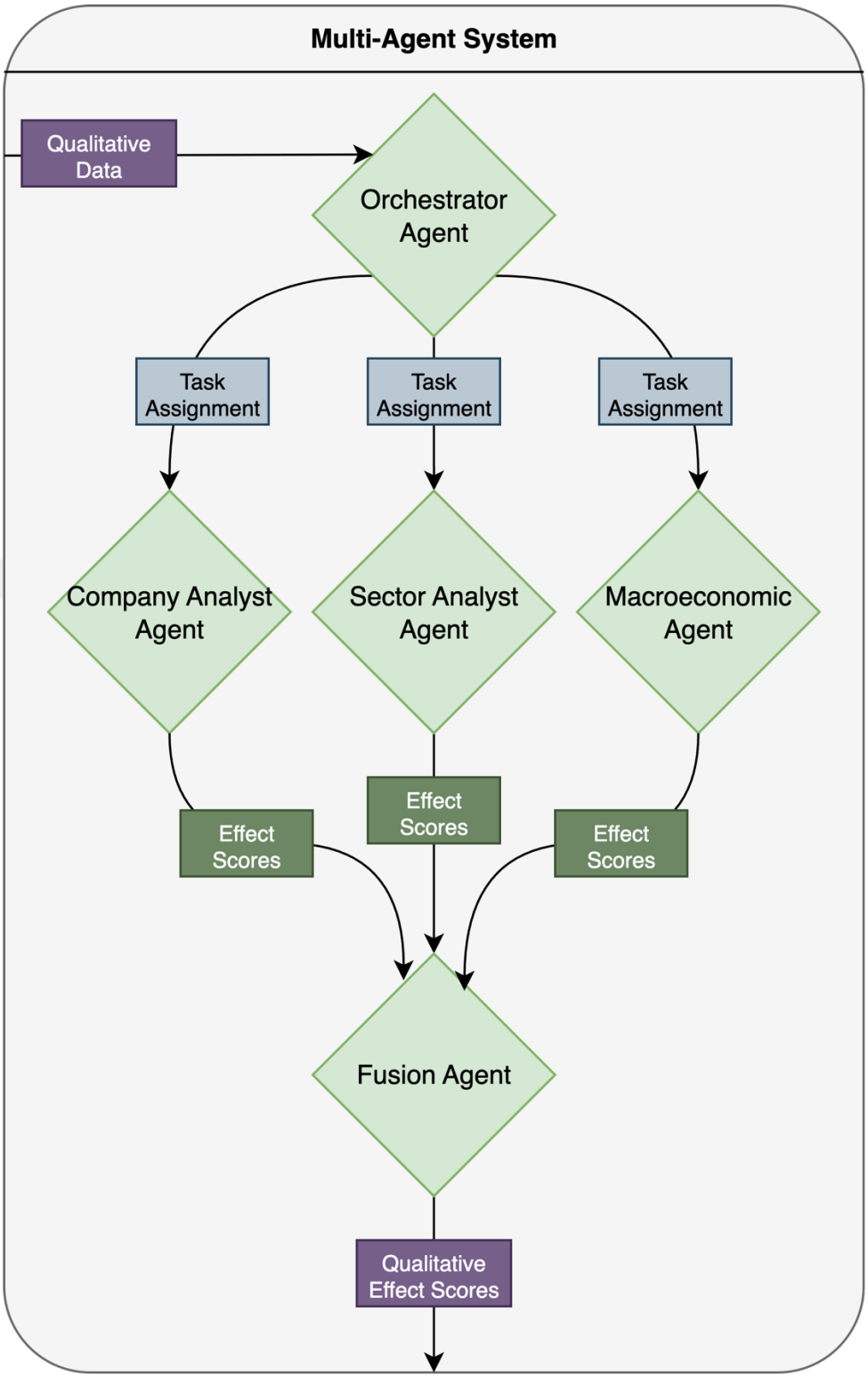

Figure 7.4. Multi-Agent System

7.10. Orchestrator Agent: Task Routing and Coordination

The Orchestrator Agent acts as the system's central nervous system. Its primary task is to take the unstructured text output from the Data Collection and Preprocessing Module, perform an

initial screening, and route the item to the appropriate specialist agents for deeper analysis. The Orchestrator superficially examines the raw news texts. If the news contains an event directly related to a company, such as “Balance Sheet”, “Contract”, “Investment” or “Penalty”, it delegates the task to the Company Analyst Agent. If it concerns a sector, such as “Regulation”, “Incentive”, “Raw Material Prices”, it directs the task to the Sector Analyst Agent. If it concerns general market conditions, such as “Macroeconomy”, “Interest Rate Decision”, “Inflation Data”, it forwards the task to the Macroeconomic Agent. If a news item concerns more than one category (for example, a company's large investment affects both the company itself and the sector), the Orchestrator sends the task to all relevant agents simultaneously.

7.11. Expert Analyst Agents: Deep Analysis and Scoring

The expert analyst agents, by focusing on a specific area of expertise, analyze the task coming from the Orchestrator in a detailed manner and produce a score from their own perspectives. The company analyst agent evaluates the direct impact of an event on a single company. It calculates company-specific impact scores by analyzing the raw text. Here, the score of the agent’s own analysis, the magnitude of the impact, the short-medium-long term effect, and the confidence interval are specified. The sector analyst agent focuses on the impact of a news item or event on an entire sector rather than a single company. For example, news of an import tax imposed on the steel sector is evaluated as a positive impact for all companies in the sector, and this agent’s score becomes highly positive. The macroeconomic agent, on the other hand, looking from the broadest perspective, evaluates the impact of an event on general market sentiment and risk appetite, and in this respect, it analyzes news such as central bank interest rate decisions, inflation data, and geopolitical tensions. For example, an unexpected interest rate hike is generally considered a negative impact in the short term for all stocks on the stock exchange, and this agent’s score becomes negative.

7.12. Fusion Agent: Combining Multiple Perspectives and Final Decision

The Fusion Agent is the final decision-making mechanism of the system. It takes the individual scores coming from the different expert analyst agents and, by combining them within a logical framework, produces a short, medium, and long-term “Qualitative Impact Scores” specific to different companies and sectors for the relevant news event. This scheme determines the priorities according to the nature of the analysis. LLM Prompt of Fusion Agent is given in Appendix.

7.13. Hybrid Prediction Model: Merging Qualitative and Quantitative Scores

The primary output of the Fusion Agent is a set of "Qualitative Impact Scores" which represent the strategic, event-driven outlook for a stock. However, this directional signal does not specify an optimal entry or exit price. To bridge this gap, a hybrid model is employed that combines the qualitative "what to trade" signal with a simple quantitative "when to trade" model for execution timing.

Quantitative Model is a simple Linear Regression Channel calculated for each stock. For a given rolling window of recent price data (e.g., the last data from 5-minute window), a linear regression line is fitted to identify the short-term trend. Upper and lower bands are then calculated based on one standard deviation from this regression line. This channel represents the expected trading range of the stock, assuming its recent trend continues.

The execution logic is triggered only when both the qualitative and quantitative signals align, creating a high-conviction entry point. The signal merging process is as follows:

- **Signal Generation:** The Fusion Agent generates a new set of Qualitative Impact Scores based on a news event, providing a '*short_term_impact*' score (ranging from -1.0 to +1.0) for an affected stock.
- **Execution Filter:** This score is treated as a strategic directive. The system then consults the quantitative Linear Regression Channel for tactical timing.
    - **Buy Condition:** A buy is considered if the '*short_term_impact*' score is strongly positive (e.g., > 0.7) AND the stock's current price is trading near or below the lower band of its Linear Regression Channel. This logic aims to act on positive news only when the asset is at a tactical discount relative to its recent trend.
    - **Sell Condition:** A sell is considered if the '*short_term_impact*' score is strongly negative (e.g., < -0.7) AND the stock's current price is trading near or above the upper band of its Linear Regression Channel. This logic aims to act on negative news when the asset's price is at a tactical premium.
- **Position Management:** Once a position is reached to the target price based on this hybrid signal, the quantitative model is no longer used for that trade. The target price or exit strategy would then be based on subsequent qualitative analysis or predefined risk management rules (e.g., stop-loss or take-profit levels).

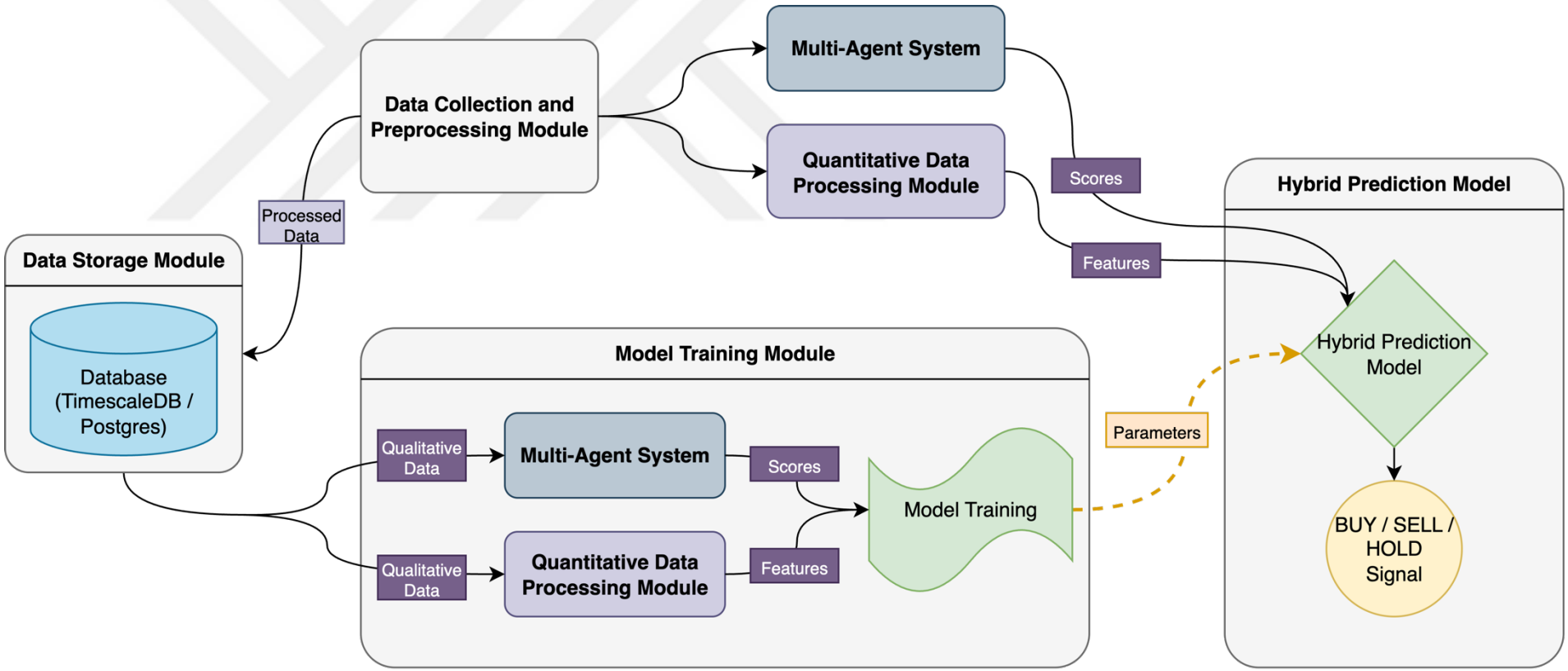

Figure 7.5. Hybrid Prediction Model

This hybrid approach systematically combines the forward-looking, context-aware analysis of the LLM agents with the price-based discipline of a quantitative model, aiming to improve the timing of entries based on the qualitative signals. The graph showing the entirety of this developed system and the interaction between the modules is given in Figure 7.5.

7.14. Backtesting Framework

To objectively assess the economic viability and predictive power of the proposed system, a robust backtesting framework will be developed. The primary objective of this framework is to simulate the execution of the hybrid model on historical data, providing a quantitative evaluation of its performance under realistic market conditions. This evaluation is not only a test of the system's profitability but also a critical analysis of its risk characteristics and its superiority over simpler, alternative strategies.

To prevent overfitting and ensure generalization to unseen data, the historical dataset will be strictly split into two non-overlapping periods: an in-sample training period that precedes a separate out-of-sample evaluation period:

- **In-Sample Training Period:** This period will be used for model calibration, parameter tuning (e.g., determining the optimal lookback window for the linear regression channel), and initial observation.
- **Out-of-Sample Evaluation Period:** The final, definitive performance evaluation will be conducted on this period. The model will run on this data only once, simulating real-world performance as if the system were trading live.

The simulation will be performed using an event-driven backtesting engine that processes data chronologically, one timestamp at a time. This is crucial for avoiding lookahead bias, as it ensures that any trading decision is made only with information that would have been available at that precise moment in time.

7.15. Simulation Assumptions and Realism

To ensure the simulation results are as realistic as possible, the following factors will be incorporated:

- **Transaction Costs:** A commission fee will be deducted for every simulated buy and sell order to account for brokerage costs. A realistic rate for the BIST (e.g., 0.05% of the transaction value) will be applied.
- **Slippage:** To model the difference between the expected trade price and the actual execution price, a fixed slippage cost will be applied to each trade entry and exit.
- **Position Sizing:** A fixed fractional position sizing rule will be employed to manage risk. For each trade signal, a constant percentage of the current portfolio equity (e.g., 5%) will be allocated.

7.16. Performance Metrics

The system's performance will be evaluated using a comprehensive suite of metrics that cover profitability and risk metrics:

- Overall Profitability Metrics:
    - **Cumulative Return:** The total percentage gain or loss of the portfolio over the entire out-of-sample period.
    - **Annualized Return:** The geometric average amount of money earned by the portfolio per year.
- Risk Metrics:
    - **Sharpe Ratio:** Measures the excess return per unit of total risk (volatility). This is the primary metric for evaluating risk-adjusted performance.
    - **Maximum Drawdown**: The largest peak-to-trough decline in portfolio value, representing the worst-case loss from a single high point.
    - **Volatility:** The standard deviation of the portfolio's daily or weekly returns, indicating the strategy's price fluctuation.

7.17. Baseline Models for Comparison

To demonstrate the unique value contributed by the integrated, multi-agent hybrid system, its performance will be benchmarked against several simpler models. This comparative analysis is essential to prove that the system's complexity is justified.

- **Passive Buy-and-Hold:** The performance of a passive investment in a BIST30 index tracking fund (e.g., XU030). This represents the opportunity cost of not investing in the market index.

- **Quantitative Model:** A strategy that trades solely based on the Linear Regression Channel without any input from the LLM agents (e.g., buying at the lower band and selling at the upper band). This will isolate and measure the value added by the qualitative analysis.

By comparing the full hybrid model against these two baselines, this study will be able to draw clear conclusions about the specific contributions of its qualitative, quantitative, and integrated components.

8. RESULTS AND CASE STUDY

In this chapter, the experimental results and financial performance of the developed “Agent-Based Financial Prediction System Supported by Large Language Models” are analyzed through a comprehensive backtesting process. The analyses are focused on proving that the “Qualitative Impact Score”, which is the main innovation of the system, provides a statistically significant and economically valuable predictive power in comparison to a model based solely on technical indicators.

8.1. Experimental Setup and Methodology

To ensure reliability and reproducibility of the backtests, news and price data between 1 March 2025 and 15 August 2025 were recorded in the database in real time. Market data were collected from the Algolab API, while news data were obtained from KAP disclosures and Habertürk RSS feeds. Subsequently, the recorded dataset was split so that observations before 15 July 2025 were used as training data, and those from 15 July 2025 to 15 August 2025 were reserved for testing. In this way, by storing the data instantly, it was aimed to reduce errors in the data acquisition process and look-ahead bias was prevented by using only past data in the training. In order to reflect realistic market conditions, commission and slippage were included in the calculations as a cost for every buy and sell transaction. The BIST30 companies examined in the study, for which market data were collected and information extraction was performed via news streams, are given in Table 8.1.

Also, Prometheus and Grafana instances are provisioned in a separate machine to monitor Data Collection Module. A visualization of an alert created in Grafana can be seen in Figure 8.1.

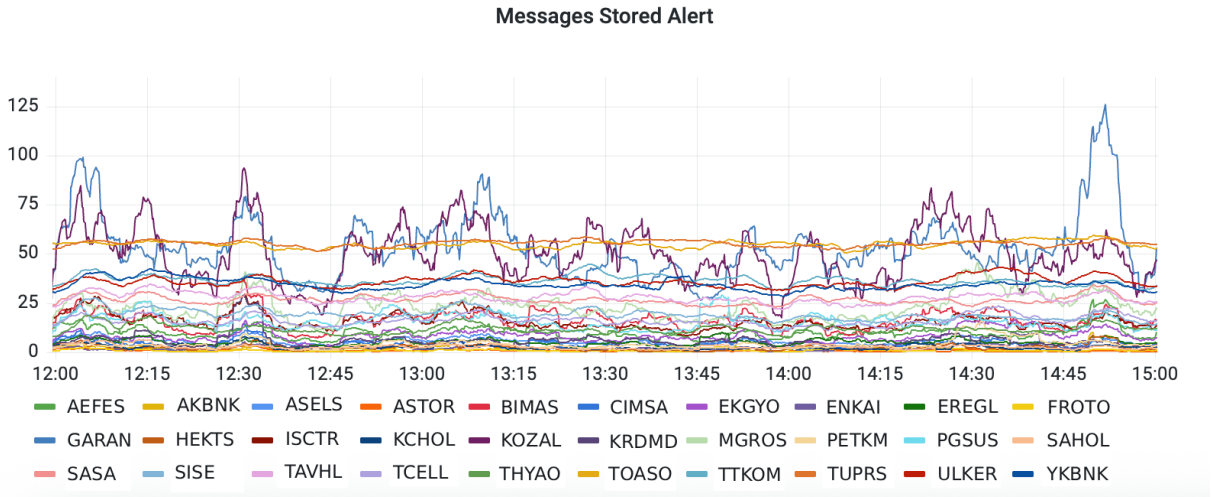

Figure 8.1. Messages Stored Alert in Grafana

Table 8.1. BIST30 Stocks

Symbol	Company	Symbol	Company	Symbol	Company
AEFES	Anadolu Efes	GARAN	Garanti BBVA	SASA	SASA Polyester
AKBNK	Akbank	HEKTS	Hektaş	SISE	Şişecam
ASELS	Aselsan	ISCTR	İş Bankası (C)	TAVHL	TAV Havalimanları
ASTOR	Astor Enerji	KCHOL	Koç Holding	TCELL	Turkcell
BIMAS	BİM Mağazaları	KOZAL	Koza Altın	THYAO	Türk Hava Yolları
CIMSA	Çimsa Çimento	KRDMD	Kardemir (D)	TOASO	Tofaş Oto Fabrika
EKGYO	Emlak Konut GYO	MGROS	Migros	TTKOM	Türk Telekom
ENKAI	Enka İnşaat	PETKM	Petkim	TUPRS	Tüpraş
EREGL	Ereğli Demir Çelik	PGSUS	Pegasus	ULKER	Ülker
FROTO	Ford Otosan	SAHOL	Sabancı Holding	YKBNK	Yapı Kredi Bankası

Three primary strategies were defined for comparison. First, the passive BIST30 index was used as a benchmark to measure market performance. Second, a Quantitative Model based on linear regression channel was developed to exploit short-term price movements; this high-frequency model targets short-term volatility. Third, the Hybrid Model, which is the core of this study, operates on a different philosophy. When a new news item arrives for a stock, the system sets a “target position” for that stock based on the generated Qualitative Impact Score. Afterwards, for the purpose of minimizing the market impact, it uses a “slow execution” algorithm that reaches this target position with small and regular intervals (with orders of 500 Turkish liras every 10 seconds) integrating linear regression channel.

In the Quantitative Model strategy, where linear regression with only technical analysis was used, features such as bid/ask level quantities, VWAP, EMA, instant spread, and transaction volume were extracted using feature engineering. By using these features, a linear regression model that predicts the mid-price 5 seconds later was trained. This model traded with an order size of 500 Turkish liras in each buy-sell decision.

Table 8.2. Analysis Results

Results	Hybrid	Quantitative	BIST30
Initial Value	₺10,000	₺10,000	₺10,000
Final Value	₺11,179	₺9,844	₺10,563
Monthly Return	11.79%	-1.56%	5.64%
Annualized Return	271.5%	16.9%	90.7%
Volatility	7.62%	42%	15.39%
Sharpe Ratio	12.70	-1.07	2.09
Maximum Drawdown	-1.06%	-9.18%	-3.05%

Table 8.2 summarizes the financial outcomes of the strategies at the end of the test period. All strategies started with TRY 10,000 initial capital. The results show a clear advantage for the proposed Hybrid Model over the other approaches. Over the one-month test window the Hybrid Model produced a return of 11.79%, outperforming both the BIST30 index (5.64%) and the Quantitative Model (-1.56%). Annualized return comparisons between the Hybrid model and the BIST30 index show notable differences when considering Turkish lira interest rates, though these figures may be affected by the shortness of the test window and prevailing market dynamics.

A visual representation of these numerical results appears in the profit&loss chart in Figure 8.2. The Quantitative Model's return curve is noisy and volatile due to its high number of trades and the simplicity of the strategy, and trading costs significantly eroded its profitability. By contrast, the Hybrid Model's curve shows a much smoother and steadier upward trend because it only takes positions on high-importance news events. Visually, this confirms the strategy's ability to avoid unnecessary trading costs and manage the risk-return trade-off effectively.

The findings indicate that profitability in financial markets is not merely about predicting the next price move but also about patient, information-driven position management. Although the



Figure 8.2. Strategy Profit&Loss Comparison

high-frequency quantitative model may possess theoretical predictive power, it could not convert that advantage into meaningful profits after market noise and trading costs. The proposed Hybrid Model, instead, isolates itself from instantaneous market noise and focuses on meaningful events that indicate a substantive change; its success derives from correctly measuring a news item's medium-term effect on an asset and taking strategic positions accordingly rather than from predicting five seconds ahead. The Hybrid Model's highest Sharpe ratio confirms that this approach is not only more profitable but also superior in capital preservation and risk management.

8.2. Case Study: Synthesizing Complex News

To provide a real example of the system's analytical process and its ability to handle nuanced information, this section details the analysis of a specific, complex market event during the test period. The event was not a single company-specific news item but a confluence of significant macroeconomic data releases that had divergent effects across different sectors.

On 04 August 2025 14:45, the system processed a series of key economic announcements. The primary positive catalysts were official reports showing Turkish inflation falling to a 44-month low, coupled with data indicating record-breaking monthly exports and strong tourism revenue. However, this positive domestic news was set against a backdrop of ongoing global geopolitical uncertainty and recent domestic regulatory pressures on the banking sector. The challenge was not to simply classify this news as "positive" or "negative", but to determine which sectors and companies are likely to benefit and which are likely to be adversely affected.

Upon receiving the several news items, the Orchestrator Agent routed the information to the relevant specialist agents. The Fusion Agent then received their analyses and synthesized them into a single, coherent strategic view. The rationale generated by the agent reveals a sophisticated, multi-layered analysis that goes far beyond simple sentiment scoring:

Strategic synthesis analyzes conflicting but important macroeconomic and sectoral signals. The main theme is the 'Opportunity and Risk Conflict'... The marked decline in inflation is the most important positive catalyst... Additionally, record levels of exports and tourism revenues support foreign currency inflows... Therefore, the outlook for Transportation (THYAO) and export-oriented Automotive (FROTO) sectors is clear and strongly positive... The Banking and Retail sectors are at the center of this conflict... While the Banking sector stands to benefit most from the decline in inflation, it is also under a series of negative regulatory pressures... In the short term, volatility and pressure are expected in Banking and Retail stocks.

This qualitative reasoning was then translated into a clear, actionable set of quantitative impact scores. Table 8.3 highlights the key short-term outputs for the affected BIST30 stocks:

Table 8.3. Fusion Agent Scores

Metric	THYAO	FROTO	GARAN	BIMAS
Short Term Impact	0.7	0.6	0.1	-0.1
Medium Term Impact	0.6	0.7	0.5	0.4
Long Term Impact	0.5	0.6	0.4	0.3
Confidence Score	0.88	0.90	0.75	0.80
Implied Strategy	Conviction Buy	Conviction Buy	Neutral Avoid	Neutral Avoid

The system correctly identified that the most direct and least conflicted beneficiaries of the news were the transportation sector (THYAO), driven by tourism, and the export-heavy automotive sector (FROTO). Crucially, it recognized the "conflict" in the banking sector (GARAN), assigning it a negligible positive score, effectively flagging it as a trade to avoid despite the positive inflation news.

Based on the high-conviction signal for THYAO (score: +0.7), the Hybrid Model's execution logic was triggered. The system set a new target position to go long on THYAO and began accumulating the stock using its slow-execution algorithm, placing small buy orders to minimize market impact.

In the subsequent two trading days, the market digested the news, and THYAO's stock price increased by approximately 5.1% (from 294 to 309), leading to a profitable trade for the Hybrid Model. Simultaneously, the GARAN stock remained volatile and ended the period nearly flat

(0.04% decrease from 150.92 to 150.20). By generating a nuanced signal, the system not only captured the upside in THYAO but also demonstrated effective risk management by avoiding a potentially misleading and unprofitable trade in GARAN.

This case study demonstrates how the agent-based system successfully moves beyond simplistic keyword analysis to a deeper, context-aware understanding. It demonstrates the model's core value proposition: identifying high-conviction opportunities born from complex events while skillfully navigating the market traps hidden within conflicting information.

9. CONCLUSION, FUTURE STUDIES, AND LIMITATIONS

This study presents a comprehensive solution to one of the fundamental challenges of modern financial markets: bridging the gap between structured numerical data and unstructured textual narratives It was set out with the objective of integrating the semantic richness in news streams, announcements, and reports, which is one of the most powerful catalysts that shape market dynamics yet is generally ignored by traditional quantitative models, into the decision-making processes. Starting from the inadequacy of traditional models in processing this valuable source of information, a hybrid, end-to-end financial prediction system that combines the semantic comprehension power of Large Language Models (LLM) with the modular and specialized structure of Multi-Agent Systems has been successfully designed, developed, and tested.

The developed system adopts a layered approach to raw news text: an Orchestrator Agent routes incoming information to relevant expert agents (Company, Sector, Macroeconomic), and a Fusion Agent converts these disparate expert views into multidimensional strategic scores that that distinguish the company-specific opportunity of a news item from the market conditions it is in. This approach goes beyond simple sentiment analysis and succeeds in capturing the nuanced dynamics underlying a news item.

Experimental results show that the proposed Hybrid Agent-Based Model outperforms both the BIST30 index and a high-frequency quantitative model. In addition to achieving high cumulative returns, the system produced a substantially higher Sharpe Ratio and a lower Maximum Drawdown, indicating not only profitability but also strong performance in risk management and capital preservation. Notably, the system could generate profits from positively scored news even under negative “Market Wind” conditions, demonstrating an ability to separate true signal from market noise and the potential to produce market-independent alpha.

Consequently, this work demonstrates that LLMs can be structured not merely as text-processing tools but as autonomous reasoning engines, and that when deployed within an agent-based architecture they can simulate a human-like panel of experts in the decision-making process. This offers a new paradigm for developing more transparent, explainable, and strategically deep systems in financial technology.

The successful results obtained in this study suggest several opportunities and research directions to advance the field. Several development paths are foreseen to further unlock the system's potential. Firstly, it should be aimed to increase the analytical depth of the model. Instead of general-purpose LLMs, a new model can be created through the fine-tuning of open-source large language models, like LLaMA, with thousands of KAP notifications, analyst reports, and financial news texts specific to Borsa Istanbul. This would yield agents more familiar with financial jargon and local market dynamics, improving accuracy and reducing hallucinations. In parallel, the Fusion Agent's decision mechanism could be enhanced through Reinforcement Learning so the system can learn from the financial outcomes (rewards/penalties) of its positions and dynamically adjust which expert agent's view to weight more under different market regimes, ultimately producing a self-optimizing architecture that adapts to changing conditions.

The expansion of the system to include alternative verified data sources such as analyst reports or the transcripts of company presentations, in addition to official news, will offer the possibility of a more holistic analysis. Along with this, the system can also gain an advantage by using signals that have not yet become official. The expansion of the system to include modules that will perform sentiment analysis on instant sentiment changes on social media platforms like X (Twitter), the posts of important economists and politicians, discussions on sectoral forums, and even the audio recordings of company webinars, could provide it with an "early warning system" capability. This can be carried to the next level with the integration of multimodal artificial intelligence that can make inferences beyond text, from graphs in analyst presentations or from the tones of voice of CEOs in press conferences.

Finally, the scope of the methodology developed in this study can be adapted to different financial instruments such as futures contracts on VIOP or crypto assets, and the signals produced can be combined with various optimization techniques to be transformed into an integrated risk management and portfolio optimization module. These proposed research directions can extend the vision established by the current work: creating smarter, more transparent, and more resilient financial systems that combine human expertise with the scalable power of artificial intelligence.

REFERENCES

Araci, D. (2019). *FinBERT: Financial Sentiment Analysis with Pre-trained Language Models* (No. arXiv:1908.10063). arXiv. https://doi.org/10.48550/arXiv.1908.10063

Ball, R. (2009). The Global Financial Crisis and the Efficient Market Hypothesis: What Have We Learned? *Journal of Applied Corporate Finance*, *21*(4), 8–16. https://doi.org/10.1111/j.1745-6622.2009.00246.x

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166. https://doi.org/10.1109/72.279181

Berger, A. L., Pietra, V. J. D., & Pietra, S. A. D. (1996). A maximum entropy approach to natural language processing. *Comput. Linguist.*, *22*(1), 39–71.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, *3*(null), 993–1022.

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). *Language Models are Few-Shot Learners* (No. arXiv:2005.14165). arXiv. https://doi.org/10.48550/arXiv.2005.14165

Chen, J. C.-Y., Saha, S., & Bansal, M. (2024). *ReConcile: Round-Table Conference Improves Reasoning via Consensus among Diverse LLMs* (No. arXiv:2309.13007). arXiv. https://doi.org/10.48550/arXiv.2309.13007

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Cheng, D., Yang, F., Xiang, S., & Liu, J. (2022). Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, *121*, 108218. https://doi.org/10.1016/j.patcog.2021.108218

Cho, K., van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In D. Wu, M. Carpuat, X. Carreras, & E. M. Vecchi (Eds.), *Proceedings of SSST-8, Eighth Workshop on Syntax,*

Semantics and Structure in Statistical Translation (pp. 103–111). Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-4012

Chomsky, N. (1957). *Syntactic Structures*. Mouton & Co.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling* (No. arXiv:1412.3555). arXiv. https://doi.org/10.48550/arXiv.1412.3555

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Darwish, M., Hassanien, E. E., & Eissa, A. H. B. (2025). Stock Market Forecasting: From Traditional Predictive Models to Large Language Models. *Computational Economics*. https://doi.org/10.1007/s10614-025-11024-w

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (No. arXiv:1810.04805). arXiv. https://doi.org/10.48550/arXiv.1810.04805

Divate, Prof. M., Jadhav, P., Jha, A., Joshi, S., & Darak, K. (2024). Harnessing LLMs for Financial Forecasting: A Systematic Review of Advances in Stock Market Prediction and Portfolio Optimization. *International Journal for Research in Applied Science and Engineering Technology*, *12*(11), 1101–1105. https://doi.org/10.22214/ijraset.2024.65283

Fama, E. F. (1965). The Behavior of Stock-Market Prices. *The Journal of Business*, *38*(1), 34–105.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, *270*(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054

G. Cairns, A. J. (2014). Efficient Markets Hypothesis. In R. S. Kenett, N. T. Longford, W. W. Piegorsch, & F. Ruggeri (Eds.), *Wiley StatsRef: Statistics Reference Online* (1st ed.). Wiley. https://doi.org/10.1002/9781118445112.stat04698

Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review*, *34*, 100190. https://doi.org/10.1016/j.cosrev.2019.08.001

Gao, C., Lan, X., Li, N., Yuan, Y., Ding, J., Zhou, Z., Xu, F., & Li, Y. (2023). *Large Language Models Empowered Agent-based Modeling and Simulation: A Survey and Perspectives* (No. arXiv:2312.11970). arXiv. https://doi.org/10.48550/arXiv.2312.11970

Gao, S., Wen, Y., Zhu, M., Wei, J., Cheng, Y., Zhang, Q., & Shang, S. (2024). *Simulating Financial Market via Large Language Model based Agents* (No. arXiv:2406.19966). arXiv. https://doi.org/10.48550/arXiv.2406.19966

González, I. R., D'Souza, G. E., & Ismailova, Z. (2018). Agent-Based Modeling: An Application to Natural Resource Management. *Journal of Environmental Protection*, *9*(9), Article 9. https://doi.org/10.4236/jep.2018.99062

Gottwald, R. (1913). The dependence between stock price and intrinsic value of a stock. *AD ALTA Journal of Interdisciplinary Research*, *1871*, 1914–1944.

Griffin, D., & Tversky, A. (1992). The weighing of evidence and the determinants of confidence. *Cognitive Psychology*, *24*(3), 411–435. https://doi.org/10.1016/0010-0285(92)90013-R

Griffioen, G. A. W. (2003). *Technical analysis in financial markets*. Thela Thesis.

Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*(8), 10389–10397. https://doi.org/10.1016/j.eswa.2011.02.068

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hommes, C. H. (2005). Heterogeneous Agent Models in Economics and Finance. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.742384

Huang, J., Xiao, M., Li, D., Jiang, Z., Yang, Y., Zhang, Y., Qian, L., Wang, Y., Peng, X., Ren, Y., Xiang, R., Chen, Z., Zhang, X., He, Y., Han, W., Chen, S., Shen, L., Kim, D., Yu, Y., … Tsujii, J. (2025). *Open-FinLLMs: Open Multimodal Large Language Models for Financial Applications* (No. arXiv:2408.11878; Version 2). arXiv. https://doi.org/10.48550/arXiv.2408.11878

Jennings, N. R., Sycara, K., & Wooldridge, M. (1998). A Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, *1*(1), 7–38. https://doi.org/10.1023/A:1010090405266

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y., Chen, D., Dai, W., Chan, H. S., Madotto, A., & Fung, P. (2023). Survey of Hallucination in Natural Language Generation. *ACM Computing Surveys*, *55*(12), 1–38. https://doi.org/10.1145/3571730

Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263. https://doi.org/10.2307/1914185

Kearney, C., & Liu, S. (2014). Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, *33*, 171–185. https://doi.org/10.1016/j.irfa.2014.02.006

Kim, Y. (2014). *Convolutional Neural Networks for Sentence Classification* (No. arXiv:1408.5882). arXiv. https://doi.org/10.48550/arXiv.1408.5882

Kirkpatrick, C. D., & Dahlquist, J. R. (2010). Technical Analysis: The Complete Resource for Financial Market Technicians. FT Press.

Lample, G., & Conneau, A. (2019). *Cross-lingual Language Model Pretraining* (No. arXiv:1901.07291). arXiv. https://doi.org/10.48550/arXiv.1901.07291

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2020). *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations* (No. arXiv:1909.11942). arXiv. https://doi.org/10.48550/arXiv.1909.11942

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension* (No. arXiv:1910.13461). arXiv. https://doi.org/10.48550/arXiv.1910.13461

Li, H., Cao, Y., Yu, Y., Javaji, S. R., Deng, Z., He, Y., Jiang, Y., Zhu, Z., Subbalakshmi, K., Xiong, G., Huang, J., Qian, L., Peng, X., Xie, Q., & Suchow, J. W. (2024). *INVESTORBENCH: A Benchmark for Financial Decision-Making Tasks with LLM-based Agent* (No. arXiv:2412.18174). arXiv. https://doi.org/10.48550/arXiv.2412.18174

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach* (No. arXiv:1907.11692). arXiv. https://doi.org/10.48550/arXiv.1907.11692

Lo, A. W., Singh, M., & ChatGPT. (2023). From ELIZA to ChatGPT: The Evolution of Natural Language Processing and Financial Applications. *The Journal of Portfolio Management*, *49*(7), 201–235. https://doi.org/10.3905/jpm.2023.1.512

Loughran, T., & Mcdonald, B. (2011). When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x

Malkiel, B. (2011). *The Efficient-Market Hypothesis and the Financial Crisis*. https://www.semanticscholar.org/paper/The-Efficient-Market-Hypothesis-and-the-Financial-Malkiel/9448a1c80c9fa81dcb9043ac4f66e87cdbc541eb

Malkiel, B. G. (1991). Efficient Market Hypothesis. In J. Eatwell, M. Milgate, & P. Newman (Eds.), *The World of Economics* (pp. 211–218). Palgrave Macmillan UK. https://doi.org/10.1007/978-1-349-21315-3_28

Masood, F. (2024). Behavioral Finance: Investor Psychology in Volatile Markets. *International Journal of Emerging Multidisciplinaries: Social Science*, *3*(1), 8. https://doi.org/10.54938/ijemdss.2024.03.1.360

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space* (No. arXiv:1301.3781). arXiv. https://doi.org/10.48550/arXiv.1301.3781

Mishev, K., Gjorgjevikj, A., Vodenska, I., Chitkushev, L. T., & Trajanov, D. (2020). Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers. *IEEE Access*, *8*, 131662–131682. https://doi.org/10.1109/ACCESS.2020.3009626

Nelson, D. M. Q., Pereira, A. C. M., & De Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, 1419–1426. https://doi.org/10.1109/IJCNN.2017.7966019

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, *53*(4), 3007–3057. https://doi.org/10.1007/s10462-019-09754-z

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., … Zoph, B. (2024). *GPT-4 Technical Report* (No. arXiv:2303.08774). arXiv. https://doi.org/10.48550/arXiv.2303.08774

Patil, R., Boit, S., Gudivada, V., & Nandigam, J. (2023). A Survey of Text Representation and Embedding Techniques in NLP. *IEEE Access*, *11*, 36120–36146. https://doi.org/10.1109/ACCESS.2023.3266377

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global Vectors for Word Representation. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162

Potters, M., Cont, R., & Bouchaud, J.-P. (1997). *Financial markets as adaptative ecosystems* (No. arXiv:cond-mat/9609172). arXiv. https://doi.org/10.48550/arXiv.cond-mat/9609172

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language Models are Unsupervised Multitask Learners*.

Russell, S. J., & Norvig, P. (2003). *Artificial Intelligence: A Modern Approach* (2nd ed.). Pearson Education.

Ryzin, J. V., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1986). Classification and Regression Trees. *Journal of the American Statistical Association*, *81*(393), 253. https://doi.org/10.2307/2288003

Samuel, A. L. (1959). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, *3*(3), 210–229. https://doi.org/10.1147/rd.33.0210

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2020). *DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter* (No. arXiv:1910.01108). arXiv. https://doi.org/10.48550/arXiv.1910.01108

Schwager, J. D. (1995). *Technical Analysis*. John Wiley & Sons.

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2019). *Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005-2019* (No. arXiv:1911.13288). arXiv. https://doi.org/10.48550/arXiv.1911.13288

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need* (No. arXiv:1706.03762). arXiv. https://doi.org/10.48550/arXiv.1706.03762

Weiss, G. (Ed.). (1999). Multiagent systems: A modern approach to distributed artificial intelligence. MIT Press.

Weizenbaum, J. (1966). ELIZA—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, *9*(1), 36–45. https://doi.org/10.1145/365153.365168

Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, *10*(2), 115–152. https://doi.org/10.1017/S0269888900008122

Xiao, Y., Sun, E., Luo, D., & Wang, W. (2025). *TradingAgents: Multi-Agents LLM Financial Trading Framework* (No. arXiv:2412.20138). arXiv. https://doi.org/10.48550/arXiv.2412.20138

Yang, Y., UY, M. C. S., & Huang, A. (2020). *FinBERT: A Pretrained Language Model for Financial Communications* (No. arXiv:2006.08097). arXiv. https://doi.org/10.48550/arXiv.2006.08097

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2020). *XLNet: Generalized Autoregressive Pretraining for Language Understanding* (No. arXiv:1906.08237). arXiv. https://doi.org/10.48550/arXiv.1906.08237

APPENDIX

Appendix 1: Fusion Agent LLM Prompt

```
# TASK: Multi-Analysis Fusion and Strategic Scoring

# ROLE / IDENTITY
You are a "Chief Strategist" or "Head of Research" who evaluates reports from different expert
analyst teams (Company, Sector, Macro) and prepares a final strategic briefing. Your job is
not just to combine scores, but to interpret analyst consensus and conflicts, identify the
core narrative behind the news, and produce multi-dimensional, actionable scores for an
investor.

# CORE INFO / FOCUS UNIVERSE
When making decisions, keep in mind the importance of BIST 30, the main index of Borsa
Istanbul:
BIST30_Stocks = [
  "AEFES", "AKBNK", "ASELS", "ASTOR", "BIMAS", "CIMSA", "EKGYO", "ENKAI",
  "EREGL", "FROTO", "GARAN", "HEKTS", "ISCTR", "KCHOL", "KOZAL", "KRDMD",
  "MGROS", "PETKM", "PGSUS", "SAHOL", "SASA", "SISE", "TAVHL", "TCELL",
  "THYAO", "TOASO", "TTKOM", "TUPRS", "ULKER", "YKBANK"
]

# INSTRUCTIONS
Carefully read one or more expert analyses provided to you. Understand each one's score and
rationale. Then, follow the chain-of-thought below to synthesize these analyses and produce
your output in the specified JSON format.

# CHAIN OF THOUGHT (Step-by-Step Strategic Synthesis)
1.  **Ingest the Inputs:** Identify which analyst reports you received and each one's main
score/argument. (e.g., "I have a +0.8 Company Analysis and a -0.3 Macro Analysis.").
2.  **Determine the Main Theme (Dominant Narrative):** Ask yourself: "What's the real story of
this news?"
    *   If the news centers on a single company, the main theme is "Company-Specific
Opportunity/Risk" (Micro).
    *   If the news centers on the overall market (rates, inflation), the main theme is
"Market Sentiment" (Macro).
3.  **Compute the "Alpha Score":** This score represents the **company-specific, market-
divergent opportunity** created by the news.
    *   If there is a `Company Analyst` report, its score is the primary determinant of
`alpha_score`. This is the "pure" company effect of the news.
    *   If there is no `Company Analyst` report, this score is 0.0 because there is no
company-specific alpha.
4.  **Compute the "Market Context Score":** This score represents the **direction of the
general market environment** in which the trade would occur.
    *   Combine the scores of the `Macro Agent` and the `Sector Analyst` to form this value.
    *   If only one exists, use its score directly. If both exist, you may take a weighted
average depending on the main theme. (e.g., If it's a sector regulation, give more weight to
the Sector score.)
    *   If neither report exists, this score is 0.0.
5.  **Interpret Conflicts and Form the Final Score:** This is the most critical step.
    *   **Alignment Case:** If `alpha_score` and `market_context_score` have the same sign
(both positive or both negative), this is a high-conviction signal. The final score should
reflect and amplify this.
    *   **Conflict Case (Most Important Scenario):** If `alpha_score` is positive but
`market_context_score` is negative (good company news, bad market), this means "swimming
against the current." The final score should temper alpha's strength by the market risk.
    *   Calculate the final `final_synthesized_score` according to the weight of the main
theme. For example, if the main theme is "Micro," a formula like `(0.7 * alpha_score) + (0.3 *
market_context_score)` may be used.
6.  **Write the Strategic Briefing:** In `synthesis_rationale`, write an executive summary
that explains all of the above steps in investor-friendly language.

# INPUT DATA
// Note: The input may contain 1, 2, or 3 analyses. The system should handle this dynamically.
```

```
{
  "COMPANY_ANALYSIS": <JSON from Company Analyst or null>,
  "SECTOR_ANALYSIS": <JSON from Sector Analyst or null>,
  "MACRO_ANALYSIS": <JSON from Macro Agent or null>
}

# OUTPUT FORMAT (JSON)
{
  "stocks" = [
    "affected_stock": "The BIST 30 ticker directly affected by the news.",
    "short_term_impact": "Expected impact and importance in the short term (1-7 days). A
number between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "medium_term_impact": "Expected impact and importance in the medium term (1-6 months). A
number between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "long_term_impact": "Expected impact and importance in the long term (6+ months). A number
between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "confidence_score": "Confidence in the accuracy of the agent's analysis (0.0 to 1.0)." //
STRICTLY a number between 0.0 and 1.0.
  ],
  "sectors" = [
    "affected_sector": "The sector directly affected by the news.",
    "short_term_impact": "Expected impact and importance in the short term (1-7 days). A
number between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "medium_term_impact": "Expected impact and importance in the medium term (1-6 months). A
number between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "long_term_impact": "Expected impact and importance in the long term (6+ months). A number
between -1.0 and 1.0.", // STRICTLY a number between -1.0 and 1.0.
    "confidence_score": "Confidence in the accuracy of the agent's analysis (0.0 to 1.0)." //
STRICTLY a number between 0.0 and 1.0.
  ],
  rationale = "A detailed, step-by-step explanation of how this synthesized score was
reached."
}

# RULE
Your output MUST be ONLY the valid JSON object strictly matching the format above. Do NOT add
any explanation, comments, or text outside the JSON object.
```