Democratic and Popular Algerian Republic
Ministry of Higher Education and Scientific Research



Yahia Fares University of Medea
Faculty of Sciences
Department of Mathematics and Computer Science

Final Master Project

Submitted for Obtaining The Master's Degree in Computer Science

Speciality: Applied Artificial Intelligence

AI-Driven Market Trend Prediction, Business Strategy Advisor and Sentiment Analysis

---

Prepared and Presented by:	**Supervisor and Co-Supervisor:**
YAGOUB MERIEM	Dr. BENSAISSA AZZEDINE
DJELLEL SOUNDES FERIAL	Dr. MAZARI AHMED CHERIF

2024-2025

Meriem's Acknowledgements

First and foremost, I thank **Allah**, the Most Gracious, The most Merciful, for granting me the strength, patience, and guidance throughout this journey. Without His divine mercy and blessings, none of this would have been possible.

Mr Benaissa, you were not only a supervisor, but a mentor in every sense of the word. Your wisdom taught me more than academic knowledge, it taught me how to think with clarity, to work with integrity, and to persist with dignity.
Your door was always open, your words always thoughtful, and your trust in me was a powerful reminder that I was capable.
There are people we meet who leave quiet marks on our lives not through grand gestures, but through their sincerity, generosity, and belief in others. You are one of those people. From the bottom of my heart, thank you. This achievement will always carry your imprint.

I would like to express my deepest gratitude to **Mr. Mazari Ahmed Cherif**,
My co-supervisor, for his invaluable support, wisdom, and encouragement. His guidance has been a beacon throughout this academic endeavor.

To my dear **Father**, thank you for your wisdom, your steady presence, and for believing in me even when I couldn't believe in myself. Your support was the backbone of my perseverance.

To my beloved **Mother**, whose prayers lit my path and whose unconditional love gave me strength in moments of doubt thank you for being my haven. Your sacrifices are etched in my heart forever.

To my wonderful sisters, **Zineb** and **Selma**, and my brothers, **Farouk, Ibrahim, Abdelkader, and AbdelFetah**, I am truly blessed to be surrounded by such a loving family and to my future husband that's gonna be lucky to have me.

A special thank you to my soul sister and bestfriend **Soundous**. Sharing this journey with you made every step lighter and more meaningful. Your strength and kindness have inspired me greatly.

I also extend my heartfelt thanks to **Soundous's family** who's been like a family of mine, her parents, her sisters, **Razane and Chiraze**, her brothers **Mohamed Yacine, Rahim, Yanis**, and of course little **Wateen**, for their warmth, encouragement, and genuine support that felt like home.

To my dear **friends** and **colleagues**, thank you for the shared moments, the support, the laughter during stressful days. Your presence made this path less lonely and more meaningful. I carry with me not only the lessons of this journey, but the warmth of the people who walked beside me.

A special thank you to my dear friends **Djihad and Riyad**, whose constant support, sincere advice, and uplifting presence brought light to some of the hardest moments.

And lastly, to **Myself. Meriem** for holding on, for showing up every day, It was definitely hard, moments of doubt, fatigue, and frustration often weighed heavily, but it is precisely through those trials that we grow. The struggles, the late nights, the setbacks…
all of it shaped me, and reminded me of what truly matters.
Because in the end, it's not just the destination,
but the person you become along the way.
Thank you for not giving up.
To all of you: thank you for being part of this chapter of my life.
This achievement is as much yours as it is mine <3

Soundous's Acknowledgements

There are moments in life when words feel too small to hold the weight of our gratitude. This is one of those moments. As I reach the end of this journey, I look back and see the people who walked with me, lifted me, and never let me fall. This is for them with all my heart.

To my beloved **mother** your prayers were my anchor, your love my quiet strength. Through every hardship, your unwavering faith in me never wavered. You saw light in me, even when I couldn't see it myself. I owe every victory to your boundless patience, your sleepless nights, and your unconditional
love. You are my first teacher, my eternal guide, and the heart of my story.

To my father, **Yacine** you have been my pillar of strength. You showed me that true courage means facing fear and walking forward anyway. Fear is temporary, but regret is forever. You didn't just say those words you lived them. You inspired me to fight, to persist, to believe. Your faith in me gave me the power to believe in myself.

To my sisters, **Razane and Chiraze**, thank you for your love, your laughter, and your quiet cheerleading. You reminded me of joy, even on the hardest days.

To my brothers **Yacine and Nazime, Yanis and Wateen** you are the bright sparks that kept my spirit alive. Your innocence, your curiosity, and your belief in me became my daily dose of motivation. I dreamed higher because of you.

To my husband, **Rafik** how do I even begin? You have been my calm in the storm, my strength when I was weak, and my safe haven in the chaos. In my moments of silence, your love spoke loudest. In my moments of despair, your belief held me up. You never let go. You are not just a part of this achievement; you are the reason it was possible. I thank Allah for choosing you for me.

To my dearest friend and soul sister, **Meriem**, and the wonderful **Yagoub** family, your generosity, warmth, and support turned heavy days into bearable ones. You gave me more than friendship, you gave me family. I will forever cherish you.

Finally, I want to thank **Myself** for showing up when no one was watching, for working through the doubt and fatigue, for believing when it would've been easier to walk away. I thank myself for the strength I didn't know I had, for the faith I chose to hold onto, and for never, ever giving up. I am proud of the woman I've become.

To all who walked this road with me this victory belongs to you too.
With deepest love and eternal gratitude,
Thank you.

Abstract

In today's fast-paced business landscape, the ability to anticipate market trends and rapidly adapt strategies is paramount. This thesis introduces an innovative AI-driven framework that integrates market trend prediction, business strategy advising, and sentiment analysis to empower organizations with data-driven insights. By harnessing advanced machine learning techniques and natural language processing, the proposed system analyzes historical market data alongside real-time social media sentiment to forecast emerging trends. It emphasizes the importance of forecasting as a core function, combining traditional statistical approaches with AI-enhanced techniques to better predict and understand market behavior. The framework employs state-of-the-art time series forecasting and machine learning methods to achieve high prediction accuracy, while its sentiment analysis component provides valuable context regarding consumer attitudes and market dynamics. Extensive experiments demonstrate that this integrated approach offers a robust decision support tool for strategic business planning.

Our empirical evaluation revealed that the AutoGluon model outperformed traditional models such as ARIMA, Holt-Winters, and Prophet in forecasting accuracy, while VADER sentiment scores successfully captured fluctuations in market sentiment, contributing to more precise and contextualized predictions. This result underscores the effectiveness of combining quantitative trend modeling with qualitative sentiment analysis for enhanced business insight.

Ultimately, this research contributes to the evolution of AI in business analytics, paving the way for more adaptive, informed, and timely decision-making in volatile market environments.

Keywords: Market Trend Prediction, Business Strategy, Sentiment Analysis, Artificial Intelligence, Machine Learning, Data Analytics, Forecasting.

Résumé

Dans le contexte économique actuel en constante évolution, la capacité à anticiper les tendances du marché et à adapter rapidement les stratégies est essentielle. Ce mémoire propose un cadre innovant basé sur l'intelligence artificielle, intégrant la prédiction des tendances du marché, le conseil stratégique et l'analyse de sentiment pour fournir aux entreprises des insights fondés sur les données. En exploitant des techniques avancées d'apprentissage automatique et de traitement du langage naturel, le système analyse à la fois les données historiques du marché et les sentiments en temps réel sur les réseaux sociaux pour prévoir les tendances émergentes. Il souligne l'importance de la prévision comme fonction centrale, combinant approches statistiques traditionnelles et techniques améliorées par l'IA pour mieux comprendre et anticiper le comportement du marché. Le cadre utilise des méthodes de pointe en prévision de séries temporelles et en apprentissage automatique pour atteindre une grande précision, tandis que le module d'analyse de sentiment offre un contexte précieux sur les attitudes des consommateurs et la dynamique du marché.

Notre évaluation empirique a montré que le modèle AutoGluon dépasse les modèles traditionnels tels que ARIMA, Holt-Winters et Prophet en termes de précision de prévision. De plus, les scores de sentiment VADER ont permis de capter efficacement les fluctuations du ressenti du marché, apportant des prédictions plus contextualisées et pertinentes. Ces résultats démontrent l'efficacité de la combinaison entre modélisation quantitative et analyse qualitative pour améliorer la prise de décision stratégique.

Cette recherche contribue ainsi à l'évolution de l'IA dans l'analyse commerciale, ouvrant la voie à une prise de décision plus adaptative, éclairée et réactive dans des environnements de marché instables.

Mots-clés : Prédiction des tendances du marché, Stratégie d'entreprise, Analyse de sentiment, Intelligence artificielle, Apprentissage automatique, Analyse de données, Prévision.

الملخص

في ظل بيئة الأعمال السريعة التغير، أصبحت القدرة على التنبؤ باتجاهات السوق والتكيف السريع مع الاستراتيجيات أمرًا بالغ الأهمية. يقدم هذا البحث إطارًا مبتكرًا مدعومًا بالذكاء الاصطناعي، يدمج بين التنبؤ باتجاهات السوق، وتقديم النصائح الاستراتيجية، وتحليل المشاعر لتزويد المؤسسات برؤى مدعومة بالبيانات. من خلال استخدام تقنيات متقدمة في التعلم الآلي ومعالجة اللغة الطبيعية، يقوم النظام المقترح بتحليل البيانات السوقية التاريخية إلى جانب مشاعر المستخدمين على وسائل التواصل الاجتماعي في الوقت الفعلي، للتنبؤ بالاتجاهات الناشئة. يؤكد هذا الإطار على أهمية التنبؤ كوظيفة أساسية، حيث يجمع بين الأساليب الإحصائية التقليدية والتقنيات المحسّنة بالذكاء الاصطناعي لفهم سلوك السوق بشكل أدق.

أظهرت التقييمات التجريبية أن نموذج AutoGluon تفوّق على النماذج التقليدية مثل ARIMA وHolt-Winters وProphet من حيث دقة التنبؤ، في حين ساهم تحليل المشاعر باستخدام VADER في التقاط تقلبات الرأي العام، مما أضاف بُعدًا نوعيًا للتوقعات. وتُبرز هذه النتائج فعالية الدمج بين النمذجة الكمية والتحليل النوعي لتحسين القرارات الاستراتيجية.

تُسهم هذه الدراسة في تطوير تطبيقات الذكاء الاصطناعي في مجال تحليلات الأعمال، مما يمهّد الطريق لاتخاذ قرارات أكثر مرونة ووعيًا وتوقيتًا في بيئات سوقية متقلبة.

الكلمات المفتاحية: التنبؤ باتجاهات السوق، استراتيجية الأعمال، تحليل المشاعر، الذكاء الاصطناعي، التعلم الآلي، تحليل البيانات، التنبؤ.

Contents

Contents i

List of Tables iv

List of Figures v

Acronyms vi

General Introduction 1
Problem Statement 1
Research Objectives 2

1 Background and Literature Review 5
1.1 Introduction 5
1.2 Background and Context 5
1.3 Key Concepts 7
1.3.1 Market Trend Prediction 7
1.3.2 Sentiment Analysis 9
1.3.3 Business Strategy Advisor 10
1.4 Conceptual Interrelationship of Key Concepts 11
1.4.1 Impact of Sentiment Analysis on Market Trend Prediction 12
1.4.2 Impact of Market Trend Prediction on Business Strategy Advisor 12
1.4.3 Impact of Sentiment Analysis on Business Strategy Advisor 12
1.4.4 AI as the Core Connector 13
1.5 Evolution and Recent Developments 13
1.5.1 Evolution of Sentiment Analysis 13
1.5.2 Evolution of Market Trend Prediction 15
1.5.3 Evolution of Business Strategy Advisory 16
1.6 Generalities On Artificial Intelligence(AI) 16
1.7 Subfields of Artificial Intelligence 17
1.7.1 Machine Learning 17
1.7.2 Deep Learning 19
1.8 Artificial Neural Networks 20
1.9 Key Components of Neural Networks 21

1.10 Convolutional Neural Network 22
1.11 Recurrent Neural Network 23
1.12 Transformer Architecture 23
1.13 Large Language Models 25
1.14 Evaluation Metrics of Models 25
1.15 Related Work 28
1.16 Conclusion 29

2 Research Methodology 32
2.1 Introduction 32
2.2 Overview of the Research Methodology 32
2.3 Google Trends 33
2.3.1 Google Trends as a Market Indicator 33
2.3.2 Why Google Trends 34
2.4 Reddit 34
2.4.1 Reddit as a Sentiment Data Source 34
2.4.2 Why Reddit 34
2.5 Using Google Trends and Reddit Sentiment for Trend Prediction 35
2.5.1 Google Trends for Trend Prediction 35
2.5.2 Reddit Sentiment for Trend Prediction 35
2.5.3 Fusion for Better Forecasting 36
2.6 Data Collection 36
2.6.1 Reddit Data Collection 36
2.6.2 Google Trends Data Collection 37
2.7 Data Preprocessing 38
2.8 Feature Engineering 39
2.8.1 Google Trends Only 39
2.8.2 Google Trends + Reddit Sentiment 39
2.8.3 Google Trends + Reddit Sentiment + Lagged Features 40
2.9 Forecasting Models 41
2.9.1 ARIMA 41
2.9.2 Prophet 42
2.9.3 Holt-Winters 42
2.9.4 LSTM 43
2.9.5 Random Forest 44
2.9.6 AutoGluon 44
2.10 Large Language Models 47
2.10.1 LLaMA 48

2.10.2 Falcon-7B 48
2.10.3 Qwen3 48
2.11 Conclusion 49

3 Results & Discussion 51
3.1 Introduction 51
3.2 Development Environment 51
3.2.1 Hardware specifications 51
3.2.2 Software Environment 52
3.2.3 Libraries 52
3.3 Comparative Analysis of Forecasting Models 53
3.3.1 Overall Performance Trends 54
3.4 Topic-Specific Model Insights 57
3.4.1 Electric Cars 57
3.4.2 Cryptocurrency 59
3.4.3 Vegan Food 61
3.4.4 AI Startups 63
3.4.5 Home Fitness 65
3.5 Forecasting Model Performance 67
3.6 Evaluation of Large Language Models 72
3.6.1 Dataset Description and Preprocessing 73
3.6.2 Evaluated Models 73
3.6.3 Evaluation Results and Analysis 73
3.7 Conclusion 75

General Conclusion 76

Bibliography 77

Webography 81

List of Tables

1.1 Comparison of Machine Learning Methods 19
1.2 Deep Learning methods 20
1.3 Summary of Selected Studies on Market Trend Forecasting 29

3.1 AutoGluon MAE Across Topics 55
3.2 Random Forest MAE Across Topics 55
3.3 Prophet MAE Across Topics 56
3.4 ARIMA MAE Across Topics 56
3.5 Holt–Winters MAE Across Topics 56
3.6 Forecast Performance – Electric Cars 68
3.7 Forecast Performance – AI Startups 69
3.8 Forecast Performance – Vegan Food 69
3.9 Forecast Performance – Cryptocurrency 70
3.10 Forecast Performance – Home Fitness 71

List of Figures

1.1 The evolution of AI in marketing 6
1.2 The development of Sentiment Analysis in the research. . . 14
1.3 Evolution of of Market Trend Prediction with AI. 15
1.4 Supervised Learning 17
1.5 Reinforcement Learning 18
1.6 Unsupervised Learning 18
1.7 Architecture Of Artificial Neural Networks 21
1.8 Architecture of Convolutional Neural Networks 22
1.9 Architecture of Recurrent Neural Networks 23
1.10 Transformer Architecture 24
1.11 Confusion Matrix and Classification Metrics 27
2.1 Market Trend Prediction System 33
2.2 Data Collection from Reddit and Google Trends 36
2.3 AutoGluon Time Series Pipeline 45
3.1 Forecasting Models Mae Heatmap 53
3.2 Forecasting Models Mae Barchart 54
3.3 Forecast vs Actual Trend-Sentiment Analysis for ”Eletric Cars” 58
3.4 Forecast vs Actual Trend-Sentiment Analysis for ”Cryptocurrency” 60
3.5 Forecast vs Actual Trend-Sentiment Analysis for ”Vegan Food” 62
3.6 Forecast vs Actual Trend-Sentiment Analysis for ”AI Statups” 64
3.7 Forecast vs Actual Trend-Sentiment Analysis for Home Fitness 66
3.8 Google Trend Prediction Comparison for ”Electric Cars” . 68
3.9 Google Trend Prediction Comparison for ”AI Startups” . . 69
3.10 Google Trend Prediction Comparison for ”Vegan Food” . . 70
3.11 Google Trend Prediction Comparison for ”Cryptocurrency” 71
3.12 Google Trend Prediction Comparison for ”Home Fitness” . 72
3.13 Evaluation of LLMs on the FinGPT Dataset 74

Acronyms

API	Application Programming Interface
ARIMA	AutoRegressive Integrated Moving Average
AI	Artificial Intelligence
DL	Deep Learning
ETS	Exponential Smoothing State Space Model
FFN	Feed Forward Network
FNN	Feedforward Neural Network
GPT	Generative Pre-trained Transformer
LLMs	Large Language Models
LSTM	Long Short-Term Memory
LlaMA	Large Language Model Meta AI
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Squared Error
NLP	Natural Language Processing
PRAW	Python Reddit API Wrapper
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SGD	Stochastic Gradient Descent
SMAPE	Symmetric Mean Absolute Percentage Error
TFT	Temporal Fusion Transformer
VADER	Valence Aware Dictionary and sEntiment Reasoner

General Introduction

In the modern era, with the rapidly developed nature of business, the ability to properly estimate market trends and create informed conclusions has become increasingly important. Traditional forecasting methods, such as those relying solely on historical time series data, are often insufficient to capture the intricate and rapidly shifting dynamics of today's markets. These markets are shaped not only by economic indicators but also by the real-time opinions and sentiments expressed across online platforms such as Reddit.

Sentiment analysis has emerged as a revolutionary technique that allows organizations to derive valuable insights from unstructured, text-based data including social media posts, news articles, and forum discussions. In this work, sentiment data is extracted primarily from Reddit, where VADER(Valence Aware Dictionary and Sentiment Reasoner), a rule-based sentiment analysis tool optimized for social media text, is applied to capture user sentiment in a quantifiable form.

Forecasting, once dominated by classical time-series models like ARIMA and Holt-Winters, now faces the challenge of incorporating high-volume, real-time data with varying structure. To address this, our framework integrates a hybrid set of forecasting models: **ARIMA** and **Holt-Winters** for statistical baseline predictions, **Prophet** for flexible and seasonality-aware forecasting, **Random Forest** for feature-rich prediction tasks, deep learning models like **LSTM**, and **AutoGluon**, an AutoML framework, to automatically tune and ensemble various machine learning models for optimal performance.

Additionally, Large Language Models (LLMs) such as **LLaMA** are fine-tuned on structured dialogue data to generate context-aware business strategy suggestions, adding a layer of interpretability to the system.

By merging qualitative sentiment signals with quantitative forecasting models, this AI-driven framework enables more accurate market predictions and strategic insights. The goal of this research is to bridge traditional and modern approaches, providing decision-makers with a robust tool to navigate volatility and seize emerging opportunities.

Problem Statement

Despite the abundance of market figures available today, many organizations struggle to effectively benefit from this information due to the

limitations of traditional forecast methods. Traditional models are usually unable to handle the instability and uncertainty of modern markets, and they often ignore important factors such as real-time consumer sentiment extracted from social media and news sources. This difference makes a sufficient challenge for businesses that need to decide on time, rapidly move market conditions. The main problem addressed in this thesis is the absence of an integrated structure that combines advanced market tendency prediction with sentiment analysis, supporting the creation of strategic business decisions.

Research Objectives

The primary objectives of this research are as follows:

- To develop an AI-powered structure that integrates the prediction of market trends with sentiment analysis, leading to improved forecast accuracy.
- To extract and analyze real-time consumer sentiment from an online platform and involve these insights in the forecast process.
- To design a strategic advisory module that generates actionable business recommendations based on integrated analysis.
- To validate the proposed structure through comprehensive experiments and a study of a real-world case, demonstrating its practical applicability in the unstable market environment.

By addressing these objectives, this thesis aims to contribute to the advancement of business analytics and strategic decision-making, helping organizations and individuals constantly respond to market changes and empowering their overall plan and performance.

This thesis is organized into three chapters:

- **Chapter 1: Background and Literature Review:** This chapter explores the key concepts of market trend prediction, sentiment analysis, and business strategy advisory, highlighting their interconnections and evolution. It also provides an overview of artificial intelligence and its subfields, including machine learning, deep learning, neural networks, and transformer-based architectures. The chapter concludes with a review of related work and current trends in the field.
- **Chapter 2: Research Methodology:** This chapter presents the methodology adopted for the study, focusing on data sources (Google Trends and Reddit), data collection strategies, preprocessing techniques, and feature engineering. It also outlines the forecasting models used—such as ARIMA, Prophet, Holt-Winters, LSTM, Random Forest, and AutoGluon—and introduces large language models like LLaMA, Falcon-7B, and Qwen3.
- **Chapter 3: Results and Discussion:** This chapter discusses the experimental setup, including the development environment and software stack. It then provides a comparative analysis of forecasting models across different topics, evaluates model performance, and analyzes the results of large language models on sentiment-driven forecasting tasks.

Chapter 1: Background and Literature Review

Chapter 1

Background and Literature Review

1.1 Introduction

There has been a deep change in the financial markets due to technological innovations and exponential growth of data. Traditional forecast models, mainly on historical numerical data, often struggled to capture the versatile and rapid shifting nature of market behavior[1]. These traditional approaches are limited when facing the instability and complexity inherent in modern financial systems.

By integrating artificial intelligence with sentiment analysis, it becomes possible to merge quantitative data with qualitative insight extracted from news, social media and other digital communications. This combined functioning provides a more holistic approach to the dynamics of the market, leading to better trend predictions and more effective strategic decision making.

1.2 Background and Context

The business intelligence landscape has changed significantly over time and the changes were largely caused by the rise of AI. Today, we are in a different space where there is simply too much data in too many formats for people to analyze data the old-fashioned way to make any useful interpretations. The combination of AI and business intelligence represents a new way to comprehend our modern, highly unpredictable world fo data, and ultimately will allow businesses to make data-driven decisions both timely and precise. The relationship between AI-driven business intelligence and market trend forecasting is complicated and multilayered. At its most basic level lies the inquiry into the approaches, methods, and algorithms, which have evolved predictive analytics - essential for organizations that are attempting to stay ahead of market dynamics. AI has already had some measurable successes in forecasting market trends accurately, enhancing decision-making agility, and identifying emerging opportunities. Furthermore, the analysis of consumer behavior - a key objective of business strategy - is also being influenced by AI more profoundly than it ever has. In the digital age, understanding consumer preferences and sentiments and providing personalized recommendations has become increasingly vital[2].

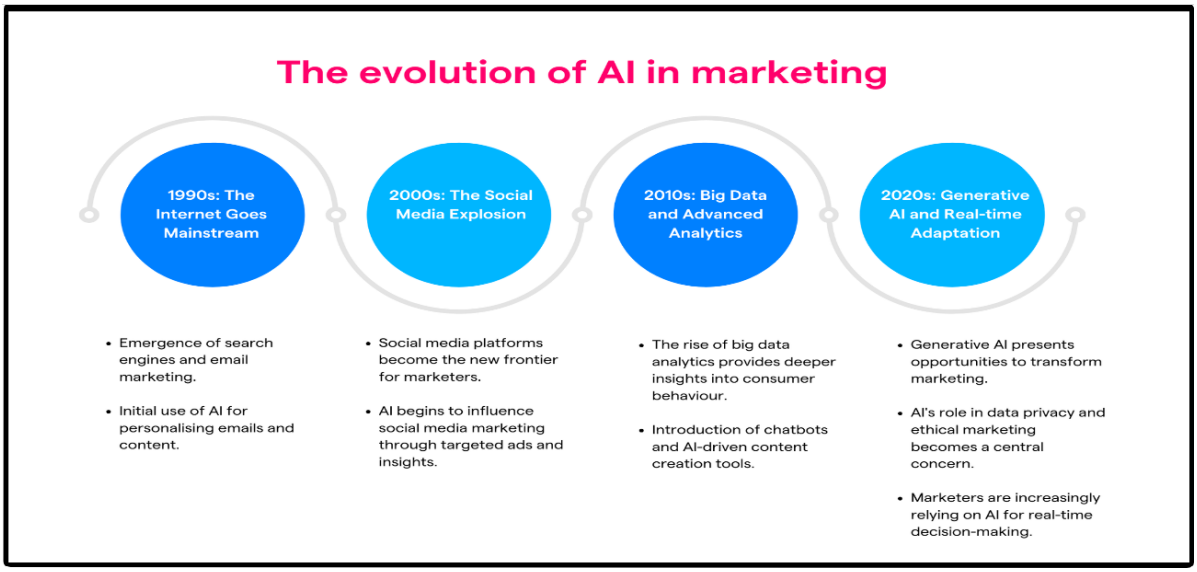


Figure 1.1: The evolution of AI in marketing

Artificial intelligence (AI) has evolved over the years, and generative AI is now really making people in the marketing industry wonder where should they be investing their time and money, and how cautiously they should embrace this technology[21].

This figure 1.1[21] illustrates the four major phases in the evolution of artificial intelligence within marketing practice, mapped across successive decades[21]:

- **1990s – The Internet Goes Mainstream:** Emergence of search engines and email marketing; initial applications of AI for content personalization and targeted email campaigns.
- **2000s – The Social Media Explosion:** Rise of social media platforms as marketing channels; early use of AI-driven audience segmentation, targeted advertising, and basic social-listening analytics.
- **2010s – Big Data and Advanced Analytics:** Increased abundance of large-scale consumer datasets allowing for deeper understanding of customer behavior; developments in AI-powered chatbots and content-creation tools to automate customer engagement.
- **2020s – Generative AI and Real-time Adaptation:**

  Advent of generative models transforming creative production and campaign optimization; growing focus on AI ethics and data privacy; widespread adoption of real-time, AI-driven decision-making in marketing operations.

1.3 Key Concepts

To establish a clear basis for this research, it is necessary to define the major concepts used during this work.

1.3.1 Market Trend Prediction

Building on this foundation, we now turn to the concept of Market Trend Prediction.

a) Definition of Market Trend Prediction

It refers to the process of predicting future market movements based on historical data, real-time economic indicators and external factors such as social media and news sentiment. Traditional statistical methods such as **ARIMA** (AutoRegressive Integrated Moving Average) and regression analysis have been widely used, but modern approaches rely on machine learning and deep learning techniques to improve rapid accuracy and adaptability.

b) Importance of Market Trend Prediction

The prophecy of market trend is essential for businesses, investors and policy makers as it enables data-operated decision making and strategic plan. By analyzing external factors such as historical data, economic indicators and social media spirit, organizations can estimate future market movements. Accurate trend allows forecast businesses to adapt to inventory management, pricing strategies and investment decisions, reduce financial risks and maximize profitability. Additionally, in unstable markets, the future models provide an initial warning of potential recession or emerging opportunities, which helps the firms to remain competitive.

c) Types of Market Trend Prediction Model

Market trend prediction models can be classified into several types based on their underlying functioning:

- **Statistical model:** Traditional forecast techniques such as ARIMA (Auto-Rigorous Integrated Moving Averages) and linear regression are widely employed in time-series analysis. These methods depend on historical data patterns and assume relative stability in trends and seasonality to generate future predictions[3].

- **Machine learning-based models:** support vector machines (SVMs), random forests, and gradient boosting leverage pattern recognition and historical data to improve predicted accuracy[4].
- **Deep learning models:** Neural networks, especially Long Short-Term Memory (LSTM) networks are applied to model complex, non-linear dependencies in financial time series and to integrate external market signals, offering superior handling of volatility and noise compared to traditional methods[5].
- **Sentiment-driven model:** Natural language processing (NLP) technology analyzes news articles, financial reports and spiritual media to assess market conditions and predict future trends[6].

d) Role of Market Trend Prediction

Forecasting market trends plays a crucial role across various domains. In stock market analysis, investors rely on predictive models to evaluate price movements, market volatility, and investment opportunities. In the retail and e-commerce sectors, businesses leverage demand forecasting to optimize supply chain management and pricing strategies. In cryptocurrency and forex trading, AI-powered models analyze real-time data to anticipate currency fluctuations and identify trading patterns. Additionally, governments and financial institutions utilize trend analysis for macroeconomic planning, policy development, and inflation control.

e) Advantages of Market Trend Prediction

There are plenty of benefits associated with market trend prediction. First of all, better decision making, in turn, helps businesses and investors make data-supported strategic choices. Additionally, risk mitigation becomes possible as the potential market can identify signs of recession and thereby reduce financial losses. Moreover, a competitive advantage is achieved since firms are enabled to estimate demand more accurately and stay ahead of competitors. Furthermore, automation and efficiency are enhanced through AI-operated models, which significantly reduce manual analysis while increasing forecast accuracy. Finally, real-time insights are made possible by modern forecasting systems that integrate live data to support dynamic and timely decision-making.

1.3.2 Sentiment Analysis

This section provides an overview of Sentiment Analysis.

a) Definition of Sentiment Analysis

Sentiment Analysis employs natural language processing (NLP) and machine learning methods to identify, extract, and measure subjective information, such as opinions, emotions, and attitudes from textual data. In the context of market trend prediction, it plays a vital role in interpreting customer feedback in order to determine public sentiment which can influence consumer behavior, investment decisions, and strategic business planning.

b) Types of Sentiment Analysis

Sentiment analysis can be classified into different types based on the level of analysis[7]:

- **Document-level sentiment analysis:** one determines the overall feeling of a complete document.
- **Sentence-level sentiment analysis:** analyzes the emotion at the sentence level to detect the positive, negative or neutral emotions.
- **Aspect-based sentiment analysis:** focuses on identifying the spirit related to any product, service or specific aspects of the subject.
- **Sentiment detection:** Goes beyond polarity(positive/negative) to identify feelings like happiness, anger or sadness.

c) Role of Sentiment Analysis

Sentiment analysis plays a vital role across multiple domains by extracting and quantifying public opinion from text. In marketing, it enables businesses to gauge customer responses, refine products and services, and monitor brand reputation in real time. In politics, sentiment analysis helps policymakers and campaigners understand public attitudes toward policies and elections. In finance, analyzing collective sentiment—often drawn from news, social media, or forums—can yield valuable insights into market movements and improve trend forecasting. Moreover, sentiment analysis enhances customer service by automating responses and elevating user experience, and it underpins social media monitoring efforts that track public opinion on a wide range of topics instantly.

d) Advantages of Sentiment analysis

The advantages of sentiment analysis are numerous. First, with access to real-time insights, organizations can respond to public sentiments very quickly therefore reducing the potential for a direct crisis. In addition, it promotes informed decision-making, enabling customers to make data-driven choices based on how the public is perceiving organizations or products. Furthermore, sentiment trends provide a competitive advantage by helping companies anticipate market movements and stay ahead of competitors. Finally, automation and efficiency are improved, as sentiment analysis reduces manual work by automating the process of opinion mining.

1.3.3 Business Strategy Advisor

This section provides an overview of the Business Strategy Advisor.

a) Definition of Business Strategy Advisor

A Business Strategy Advisor is a professional or system that provides expert guidance to organizations in formulating, evaluating, and refining strategic plans. Their role involves analyzing internal capabilities, market trends, and competitive landscapes to support informed decision-making and ensure long-term business growth and sustainability.

b) Importance of a Business Strategy Advisor

A business strategy advisor plays an important role in sustainable development and success organizations. They help businesses navigate market uncertainties, adapt operations and develop long-term strategic plans. By analyzing industry trends, competitive landscapes and financial data, they provide insights that enable companies to make informed decisions. Their expertise is required to scale businesses, enter new markets or improve overall efficiency.

c) Types of a Business Strategy Advisor

Business Strategy Advisor may be expert in various fields based on the needs of an organization:

- **Corporate Strategy Advisor:** Focus on overall business development, merger, acquisition and diversification strategies[8].

- **Financial Strategy Advisor:** provides guidance on budget, investment decisions and financial risk management[8].
- **Marketing Strategy Advisor:** Help businesses to develop branding, customer engagement and market entry strategies[8].
- **Operational Strategy Advisor:** Improving efficiency by adaptation of supply chains, resource allocation and process management[8].
- **Technology Strategy Advisor:** Assist companies in adopting new technologies, digital changes and IT infrastructure planning[8].

d) Role of a Business Strategy Advisor

A business strategy advisor plays a critical role in enhancing organizational performance by offering expert recommendations based on thorough analysis. Their responsibilities typically include conducting market research and competitive analysis, identifying opportunities for business expansion, developing risk management strategies, and ensuring that business goals are aligned with current industry trends. By integrating these insights, a business strategy advisor helps organizations navigate complex market environments and achieve sustainable growth.

e) Advantages of a Business Strategy Advisor

There are many benefits to being a business strategy advisor, first, it supports informed decision-making by providing companies with insights supported by data and industry trends. Also, a well-defined strategic plan provides a competitive edge enabling them to stay ahead of competitors. Additionally, it aids in risk mitigation by identifying and addressing potential risks before affecting business. Strategy advisors contribute to greater efficiency and development to streamline operations and adapt resources for better profitability. Moreover, they enhance a company's adaptability, ensuring that strategies can evolve in response to shifting market conditions.

1.4 Conceptual Interrelationship of Key Concepts

This section explores the conceptual interrelationship among the key elements of the study, highlighting how sentiment analysis, market trend prediction, and business strategy formulation are interconnected

1.4.1 Impact of Sentiment Analysis on Market Trend Prediction

Sentiment analysis has become a vital component in enhancing market trend forecasting. By extracting and quantifying public opinions from sources such as Reddit, sentiment analysis provides valuable indicators of future market movements. While negative attitudes may indicate possible downturns, positive sentiment patterns are frequently associated with growing market confidence. Incorporating sentiment data into predictive models allows for a more dynamic and responsive understanding of market behavior, organizations can predict changes earlier and more accurately by incorporating sentiment data into predictive models, which provide a more dynamic and responsive understanding of market behavior. For instance, research utilizing BERTopic, an advanced natural language processing technique, have demonstrated that deep learning models' ability to predict stock price trends can be greatly improved by examining the sentiment of topics extracted from stock market comments[10][9].

1.4.2 Impact of Market Trend Prediction on Business Strategy Advisor

An essential component of creating a successful business strategy advisory is forecasting market trends. Businesses can predict the state of the market in the future and take proactive measures by examining past data and finding trends, enabling them to make proactive decisions. This predictive capability allows organizations to allocate resources efficiently, anticipate customer needs, and stay ahead of competitors. By integrating trend forecasts into strategic planning, companies can make sure they are ready to take advantage of new possibilities and reduce risks rather than just responding to changes in the market. From a strategic perspective, market analysis provides businesses with a comprehensive understanding of their target audience. By analyzing demographic data, purchasing patterns, and consumer preferences, companies can tailor their products or services to meet the specific needs of their customers[10].

1.4.3 Impact of Sentiment Analysis on Business Strategy Advisor

Using sentiment analysis, a business strategy advisor can uncover in-depth information about customers' attitude towards brands, products, and overall market environment. By systematically extracting opinions from Reddit, companies will be able to better understand how brands, products,

and overall market are perceived by the public. Strategic experts with this data can then suggest more effective marketing campaigns, product innovations, and customer engagement strategies. Additionally, incorporating sentiment analysis to any business strategy ensures that companies can react more dynamically to market conditions and promote stronger connections with customers.

By analyzing customer reviews and interactions, businesses can get better insight into the satisfaction of customers, identify pain points in the marketplace, and boost brand advocacy. Positive sentiment reflects customers satisfaction and drives purchasing behavior, while negative sentiment highlights areas needing improvement, potentially influencing sales. Moreover, strong positive sentiment from loyal customers enhances brand advocacy, with word-of-mouth from brand advocates significantly impacting customer decisions[10].

1.4.4 AI as the Core Connector

Artificial intelligence – the thread that connects sentiment analysis with market trend prediction and business strategy advisory. By automating data extraction, processing, and interpretation of data from massive and dynamic data sources such as Reddit and Google Trends, AI ensures an ongoing analysis of public sentiment and signal of a market in real-time. Through predictive models that generate more accurate forecast of market trends in real-time, AI is also applying intelligent signal of the market to strategic business decisions. Using machine learning and natural language processing algorithms, AI ensures insights generated from vast and dynamic data sources are actionable, timely, and aligned with organizational objectives, further strengthening the coherence between market understanding and business strategy.

1.5 Evolution and Recent Developments

Understanding the progression of sentiment analysis, market trend prediction, and business strategy advisory highlights the growing importance of AI technologies in business decision-making.

1.5.1 Evolution of Sentiment Analysis

Sentiment analysis began with rule-based systems using keyword matching, which lacked contextual understanding[7]. The machine learning

era(mid-2000s) brought algorithms like Naive Bayes and SVMs, improving accuracy but requiring manual feature engineering. The deep learning phase(2010s) introduced RNNs and LSTMs, offering better context handling but still limited in nuance. In 2018, transformers like BERT revolutionized the field with bidirectional attention, enabling greater sensitivity to context and subtle emotions. Recent advances include domain-specific fine-tuning of models like RoBERTa and DistilBERT, and the rise of Large Language Models(e.g., GPT-3, LLaMA, Zephyr), which offer real-time sentiment detection and contextual explanations, making insights more actionable. The growing academic focus post-2022 highlights its increasing research significance.

Figure 1.2[11] illustrates the growth in academic publications and citations related to sentiment analysis over recent years, highlighting its increasing importance and attention in research.

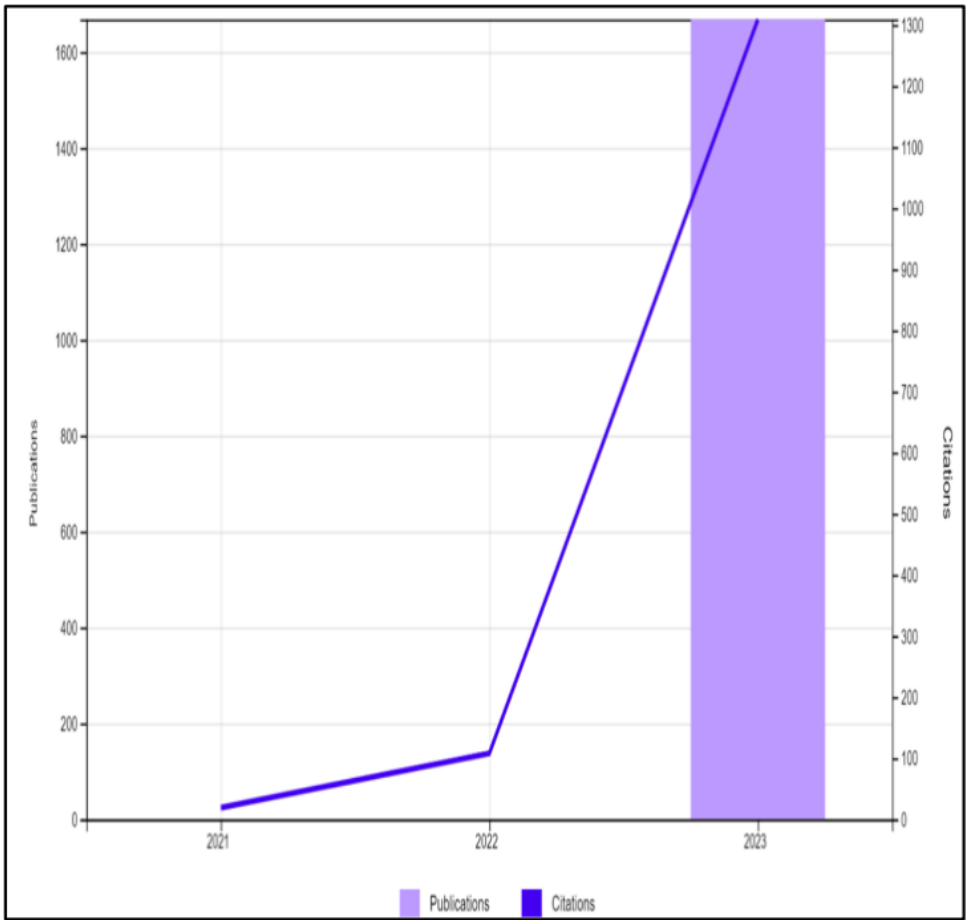


Figure 1.2: The development of Sentiment Analysis in the research.

1.5.2 Evolution of Market Trend Prediction

Early market predictions were manual and relied on expert intuition and historical data[11]. Traditional statistical methods(e.g., moving averages, regressions) lacked adaptability to market complexity. The 1980s–90s introduced technical analysis, which still couldn't incorporate unstructured data. In the 2000s, ML models like decision trees improved prediction over larger datasets but faced overfitting. The 2010s saw deep learning models(e.g., LSTM) that leveraged time series and integrated various data types (news, social media). Current hybrid systems use real-time data, reinforcement learning, and sentiment analysis for greater accuracy.

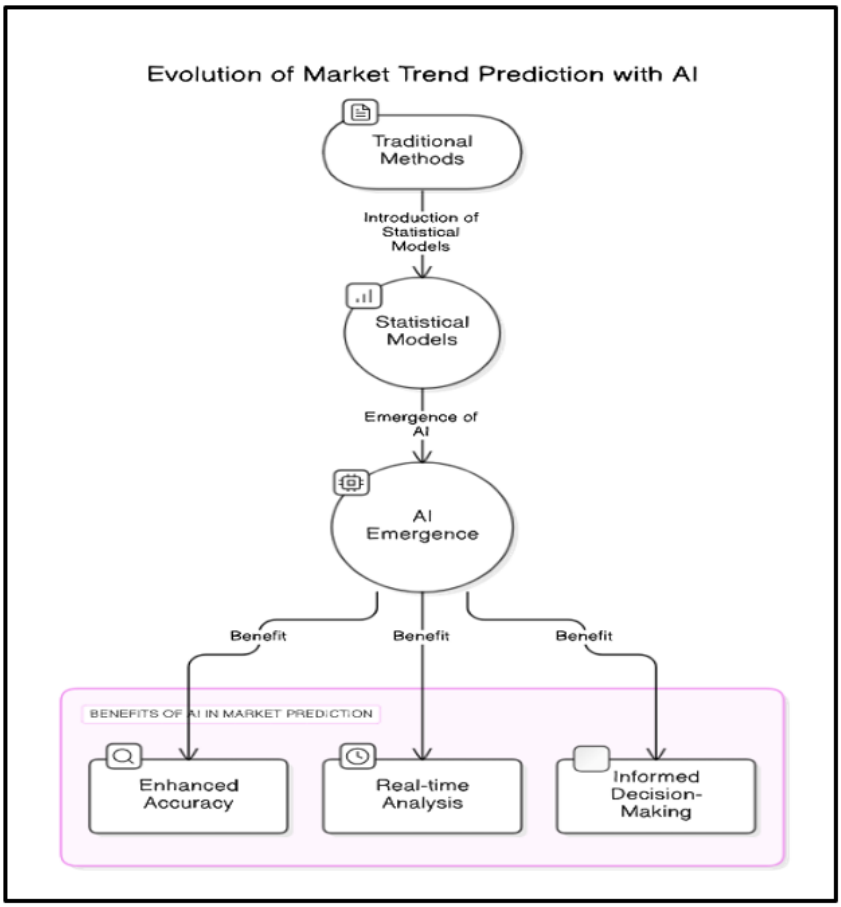


Figure 1.3: Evolution of of Market Trend Prediction with AI.

As shown in the figure1.3[12], the integration of AI models into market trend prediction systems allows for more precise forecasting, identifying patterns that were once impossible to detect, and enabling businesses to adapt swiftly to market changes.

1.5.3 Evolution of Business Strategy Advisory

Strategic planning shifted from intuition and historical analysis to data-driven approaches. Initially supported by basic tools(e.g., spreadsheets), businesses had limited actionable insights. With the advent of AI and ML, automated analytics enabled quicker strategic decisions. Today's systems integrate real-time data from social media and IoT, allowing dynamic strategy adaptation. Modern AI-powered advisors offer predictive analytics and sentiment-based insights, aiding in faster and smarter decision-making. Looking forward, AI is expected to enable autonomous strategy formulation, helping firms stay agile in evolving markets.

1.6 Generalities On Artificial Intelligence(AI)

Artificial Intelligence (AI) has become a cornerstone of modern technology, influencing a wide range of domains. it has developed rapidly in the last few decades, revolutionizing various industries and reshaping the way humans interact with technology. From self-driving cars to advanced medical diagnostics, AI-driven systems are becoming increasingly integral to daily life.

At its core, AI is a multidisciplinary field of computer science that aims to create systems capable of performing tasks that typically require human intelligence. These tasks include problem-solving, decision-making, learning, perception, and natural language understanding.

AI is designed to perform specific functions, such as image recognition, speech processing and recommendation systems. It can also undertake any intellectual work that a human can do. While general AI remains a theoretical concept, narrow AI applications have seen widespread adoption across numerous industries.

The evolution of AI has been fueled by advancements in computational power, improved algorithms, and the availability of large-scale data. It draws from multiple disciplines: mathematics (logic and probability) for formal reasoning, neuroscience for inspiration in neural network architectures, linguistics for natural language processing, cybernetics for control systems, experimental psychology for behavior modeling, and economics for decision and game theory. Together, these fields form the foundation upon which modern AI continues to grow and innovate.

1.7 Subfields of Artificial Intelligence

Artificial intelligence is composed of two primary components: machine learning and deep learning.

1.7.1 Machine Learning

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without explicit programming. Learning algorithms are used in many applications that we make use of daily. Every time a web-search engine like Google is used to search the internet, one of the reasons it works so well is because of a learning algorithm that has learned how to rank web pages. These algorithms are used for various purposes like data mining, image processing, predictive analytics, etc. to name a few. The main advantage of using machine learning is that, once an algorithm learns what to do with data, it can do its work automatically[12]. Machine Learning algorithms are mainly divided into four categories: Supervised learning, Unsupervised learning, Semi-supervised learning, and Reinforcement learning, these will be explained below.

Supervised Learning

Supervised learning uses labeled data to train models for classification and regression tasks. The model learns by comparing predictions with known outputs, as shown in Figure 1.4[12]. It involves training, evaluating, tuning, and deployment[13].

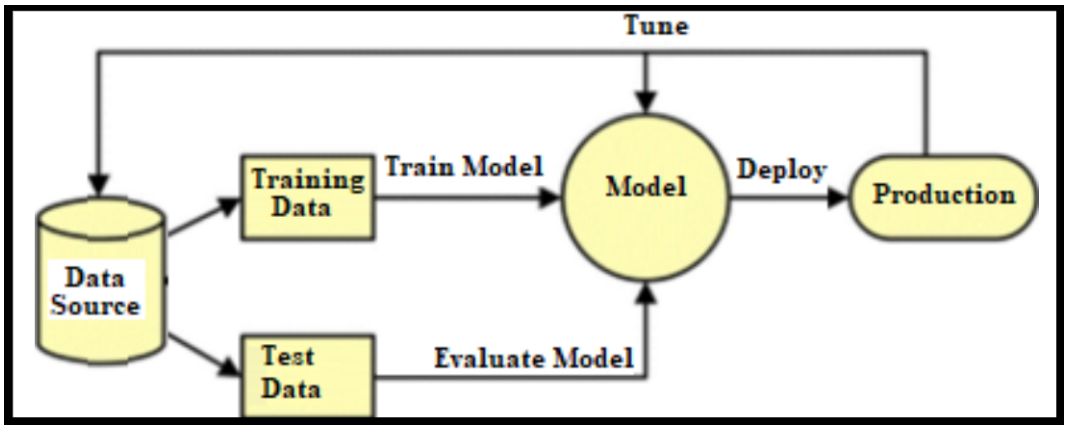


Figure 1.4: Supervised Learning

Semi-Supervised Learning

This approach combines a small amount of labeled data with a large set of unlabeled data, improving learning efficiency when labels are scarce.

Reinforcement Learning

In reinforcement learning, an agent interacts with an environment to maximize rewards, learning from feedback over time, as shown in Figure 1.5[12].

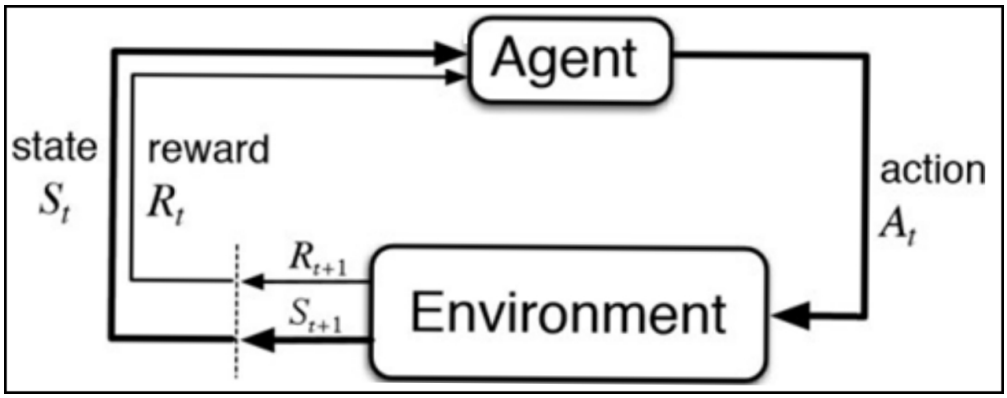


Figure 1.5: Reinforcement Learning

Unsupervised Learning

Unsupervised learning finds patterns in unlabeled data without human input. As in Figure 1.6[12], it is mainly used for clustering, association, and dimensionality reduction[13].

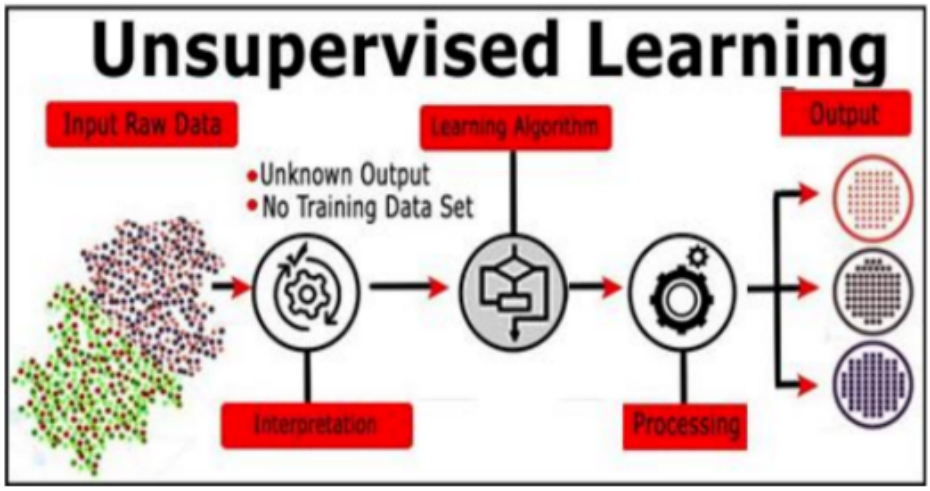


Figure 1.6: Unsupervised Learning

Methods	Description	Data Type	Common Algorithms	Applications
Supervised Learning	Learns from labeled data by mapping inputs to outputs.	Labeled data	Decision Trees, SVM, Neural Networks	Spam detection, Medical diagnosis, Sentiment analysis
Semi-Supervised Learning	Uses a small amount of labeled data and a large amount of unlabeled data.	Mix of labeled & unlabeled data	Self-training, Co-training, Graph-based methods	Speech recognition, Text classification, Medical imaging
Reinforcement Learning	Trains an agent to make sequential decisions based on rewards.	Interaction with environment	Q-Learning, Deep Q-Networks (DQN), Policy Gradient Methods	Game playing (e.g., AlphaGo), Robotics, Autonomous driving
Unsupervised Learning	Identifies patterns or structures in unlabeled data.	Unlabeled data	K-Means Clustering, PCA, Autoencoders	Customer segmentation, Anomaly detection, Topic modeling

Table 1.1: Comparison of Machine Learning Methods

1.7.2 Deep Learning

Deep Learning (DL) is a specialized branch of Machine Learning that focuses on training artificial neural networks to recognize patterns and make decisions. Inspired by the structure and function of the human brain, deep learning models consist of multiple layers of artificial neurons, allowing them to learn complex representations of data[14].

Deep Learning Methods

Some of the most effective techniques in deep learning are designed to accelerate training and improve model performance. These methods address common challenges such as overfitting, slow convergence, and instability during training. The following section introduces several widely adopted strategie, ranging from optimization algorithms to architectural enhancements, each with its own strengths and potential drawbacks. Table 1.2[15] outlines these techniques, providing a concise summary of their purpose, benefits, and limitations, and illustrating their relevance in different deep learning contexts.

Method	Description	Merits	Demerits
Backpropagation	Used in optimization problems	Calculates gradient efficiently	Sensitive to noisy data
Stochastic Gradient Descent	Finds optimal minimum in optimization problems	Avoids trapping in local minimum	Longer convergence time
Learning Rate Decay	Reduces learning rate gradually	Increases performance, reduces training time	Computationally expensive
Dropout	Drops units/connections during training	Prevents overfitting	Increases training time
Max Pooling	Applies a max filter	Reduces dimensions	Might lose important features
Batch Normalization	Normalizes inputs per batch	Stabilizes training, speeds up learning	Extra computation overhead
Skip-gram	Used in word embedding algorithms	Works on raw text, memory-efficient	Training is slow, softmax is expensive
Transfer Learning	Transfers knowledge from one model to another	Faster training, improves performance	Works only for similar tasks

Table 1.2: Deep Learning methods

1.8 Artificial Neural Networks

A neural network consists of a collection of connected nodes or neurons organized in layers: input layer, hidden layer, and output layer. Each connection between neurons is assigned a weight, and each neuron has an activation function. Amid training, the network adjusts these weights in order to reduce the error in predictions[16].

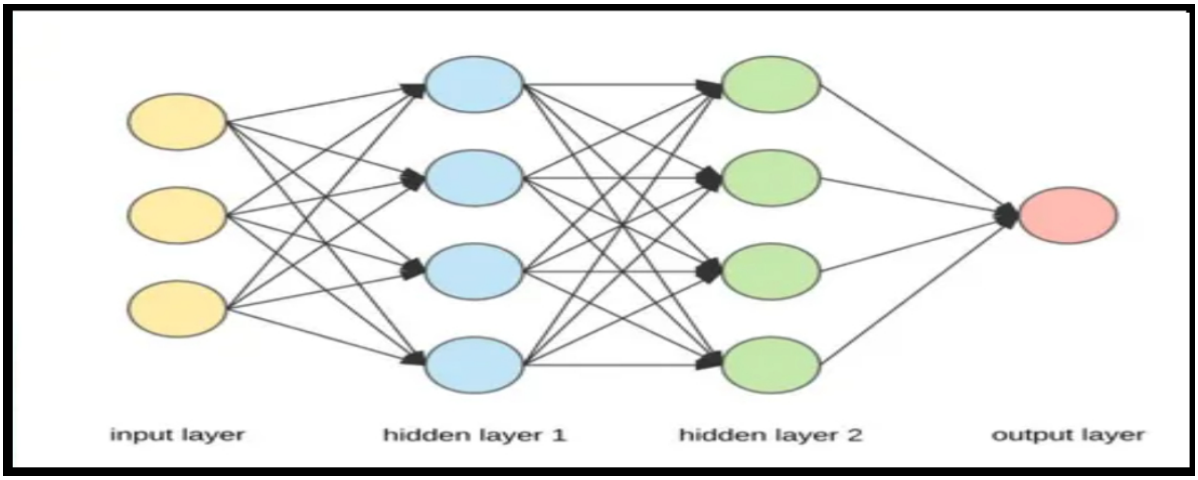


Figure 1.7: Architecture Of Artificial Neural Networks

This figure[2] illustrates a feedforward neural network (FNN) with an input layer, two hidden layers, and an output layer, showing how neurons are fully connected.

1.9 Key Components of Neural Networks

Neural networks have several key components that enable them to learn from data and make predictions. These include[8]:

- **Input Layer:** Receives raw data and passes it to the hidden layers; no computations are performed here.
- **Hidden Layers:** Perform weighted computations followed by activation functions to process data.
- **Output Layer:** Produces the final output. It varies based on the task (e.g., sigmoid for binary classification, softmax for multi-class, linear for regression).
- **Weights and Biases:** Parameters that determine the strength and flexibility of neuron connections. Adjusted during training using optimization algorithms.
- **Activation Functions:** It controls the output of a node, determining whether the node should be activated or not based on its input valueIt controls the output of a node, determining whether the node should be activated or not based on its input value. Common types are:

1. **Sigmoid:** Maps values to [0, 1], useful for binary classification.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{1.1}$$

2. **ReLU:** Outputs max(0, x), widely used due to efficiency.

$$f(x) = \max(0, x) \tag{1.2}$$

3. **Tanh:** Maps to [−1, 1], zero-centered, preferred over sigmoid in hidden layers.

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{1.3}$$

4. **Softmax:** Converts outputs to probabilities for multi-class classification.

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}} \tag{1.4}$$

1.10 Convolutional Neural Network

CNNs are specialized for processing grid-like data, especially images, as they are very effective at capturing spatial features (such as edges and textures) through a series of filters or kernels[17].. They use layers like convolution, pooling, and dense layers to extract and classify features[18].

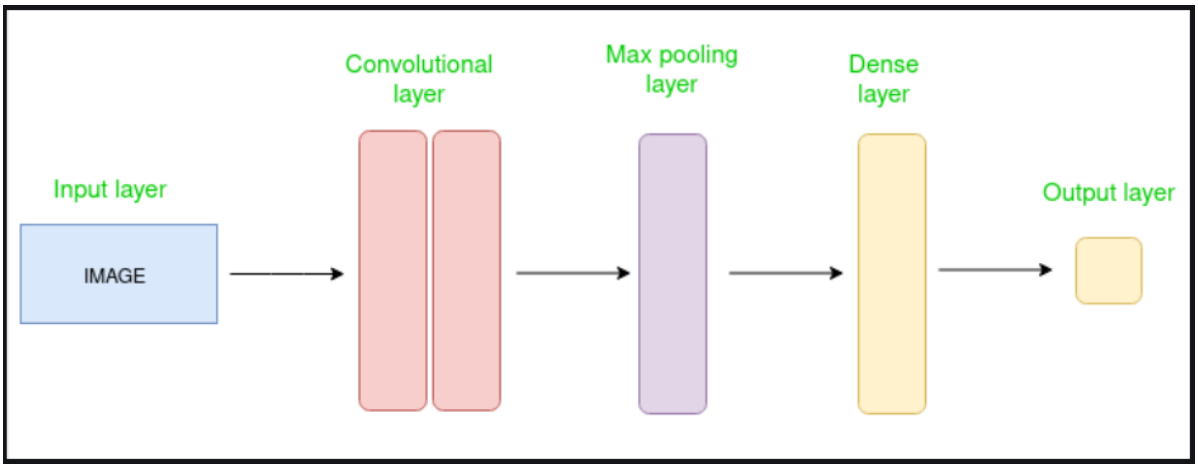


Figure 1.8: Architecture of Convolutional Neural Networks

This figure 1.8[7] illustrates the basic architecture of a Convolutional Neural Network, CNNs apply mathematical convolution by sliding a filter over an image and computing the dot product between the filter and pixel values, creating a feature map that highlights important patterns.

1.11 Recurrent Neural Network

Recurrent Neural Networks (RNNs) are a type of neural network designed to work with sequential data, such as time series or text. Unlike traditional neural networks, RNNs have connections that loop back, which allows them to maintain a memory of previous inputs. This makes them especially useful for tasks like speech recognition, language modeling, or stock price prediction.

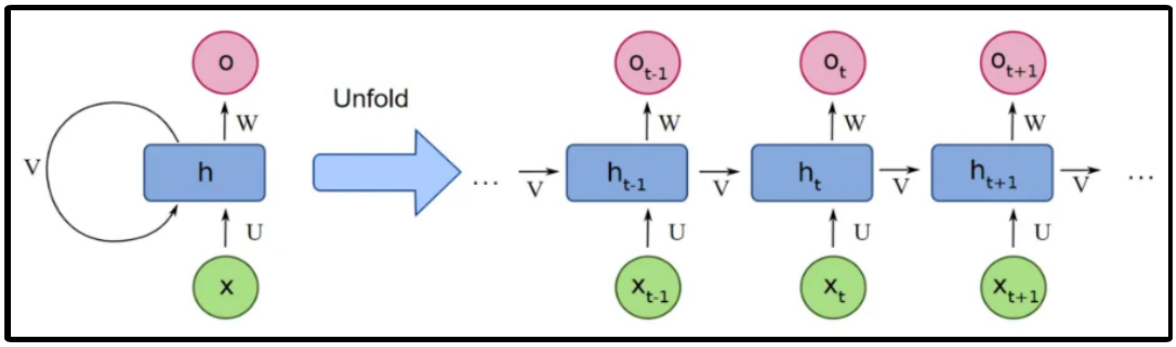


Figure 1.9: Architecture of Recurrent Neural Networks

This figure 1.9[6] illustrates the unfolding of a Recurrent Neural Network, showing how hidden states (h) are passed through time to process sequential data.

1.12 Transformer Architecture

The Transformer architecture developed by Vaswani et al. (2017) represents a new approach in sequence modeling by entirely eliminating recurrence and proposing a mechanism called **self-attention**, which enables the model to weigh the influence of different tokens in an input sequence relative to one another, regardless of their position. This allows the model to capture long-range dependencies more effectively and process longer inputs in parallel, significantly improving computational efficiency compared to traditional recurrent architectures. The architecture consists of an encoder-decoder structure as mentioned in the figure below[1].

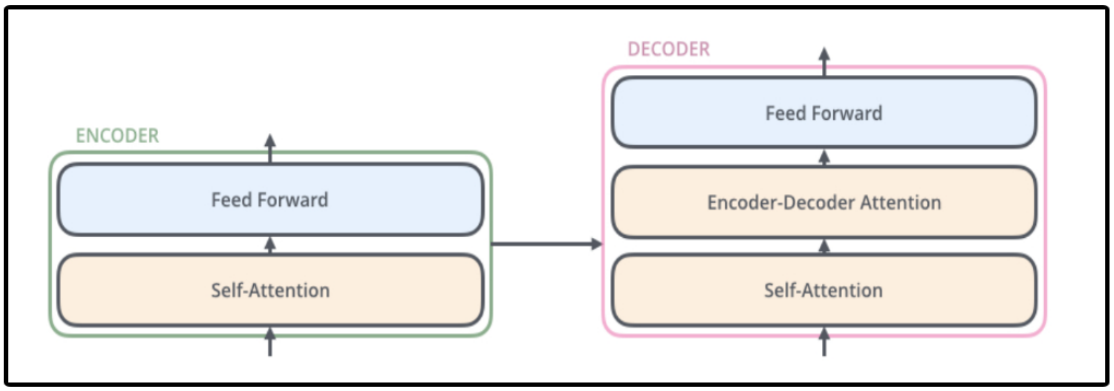


Figure 1.10: Transformer Architecture

The Transformer is built on an **encoder-decoder architecture**.

Encoder Architecture

Each encoder block is composed of[1]:

- **Multi-head Self-Attention Layer:** Allows each token to attend to all others in the input sequence, capturing contextual relationships irrespective of their distance.
- **Feed-Forward Network (FFN):** A position-wise network applied to each token independently, enhancing the representation.
- **Layer Normalization and Residual Connections:** Used after each sublayer to stabilize training and enable deeper architectures.

Decoder Architecture

The decoder generates the output sequence using[1]:

- **Masked Multi-head Self-Attention:** Prevents tokens from attending to future positions during training.
- **Encoder-Decoder Attention:** Lets the decoder attend to encoder outputs to align input context with output generation.
- **Feed-Forward Network and Normalization:** As in the encoder.

Attention Mechanism

The attention mechanism helps the Transformer focus on important words in a sequence by measuring how each word relates to others. Multi-head attention runs several attention processes in parallel to capture different relationships at once. Since the Transformer doesn't process words in order, positional encodings are added to give the model information about word positions[28].

The Transformer's self-attention-based design enables highly parallel computation and better modeling of long-range dependencies. Its architecture has since become foundational for many modern NLP models[28].

1.13 Large Language Models

A large language model (LLM) is an example of a deep learning algorithm able to perform many natural language processing (NLP) tasks. Large language models are based on transformer models and are trained using a considerable amount of data, hence large. This allows these models to recognize, translate, predict, or generate text or other content. Besides training artificial intelligence (AI) applications to learn human languages, large language models also can be trained to perform a variety of tasks like understanding protein structures, writing software code, and more. Like the human brain, large language models are first pre-trained, and then fine-tuned to be able to solve learning problems such as text classification, question answering, document summarization, or text generation. Their ability to solve tasks can be used in fields like healthcare, finance, entertainment, etc., where large language models serve a more significant role in NLP applications, such as translation, chatbots, AI assistants, and so on. Large language models also have large numbers of parameters, which function like memories accumulated during training[9].

1.14 Evaluation Metrics of Models

Evaluation metrics are quantitative measures used to assess the performance and effectiveness of a statistical or machine learning model. These metrics provide insights into how well the model is performing and help in comparing different models or algorithms[3]. When new machine learning models are created, it is necessary to compare their performance to the already existing ones. Evaluation serves two purposes: methods that do

not perform well can be discarded, and the ones that seem promising can be further optimized[19].

For Classification:

- **Accuracy:** The accuracy of a classification model is defined as the proportion of instances that were correctly classified compared to all instances in the dataset. Accuracy can be expressed as a ratio or percentage[5], and is calculated as follows:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

- **Precision (Positive Predictive Value):** Precision measures the proportion of true positive predictions among all positive predictions[5].

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall (Sensitivity or True Positive Rate):** Recall measures the proportion of true positives among all actual positive instances[5]. Calculated using the following formula:

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score:** F1-Score is the harmonic mean of precision and recall, balancing both.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Confusion Matrix:** A confusion matrix is a table showing the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

- **ROC Curve (Receiver Operating Characteristic Curve):** The ROC curve is a graphical representation of the trade-off between true positive rate and false positive rate.

- **AUC (Area Under Curve):** AUC measures the area under the ROC curve, indicating the model's ability to distinguish between classes.

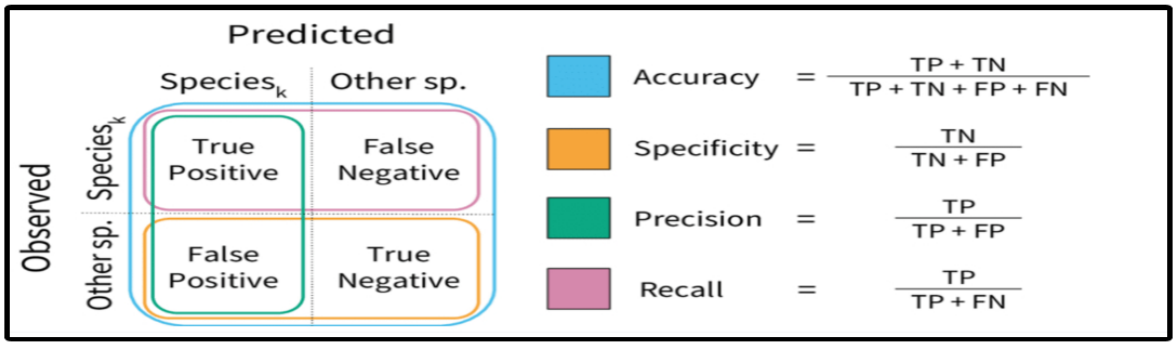


Figure 1.11: Confusion Matrix and Classification Metrics

This figure 1.11[5] shows the confusion matrix along with key classification metrics explained previously: accuracy, specificity, precision, and recall.

For Regression:

- **Mean Absolute Error (MAE):** MAE is the average of the absolute differences between predicted and actual values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- **Mean Squared Error (MSE):** MSE is the average of the squared differences between predicted and actual values.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- **Root Mean Squared Error (RMSE):** RMSE is the square root of MSE, which brings the error metric back to the original scale of the data.

$$\text{RMSE} = \sqrt{\text{MSE}}$$

- **R-Squared (R^2):** R^2 measures the proportion of variance in the dependent variable that is predictable from the independent variable(s).

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

- **Adjusted R-Squared:** Adjusted R^2 is a modified version of R^2 that accounts for the number of predictors in the model.

1.15 Related Work

The prediction of stock market trends has been a subject of extensive research, with numerous studies exploring machine learning techniques for forecasting stock price movements.

Forecasting market trends is a critical component of strategic decision-making in fields such as finance, retail, and supply chain. Companies who are able to accurately forecast market trends can more effectively optimize inventory management, manage risk, and improve profitability. However, the complexity of global markets and the sheer volume of data available means that more sophisticated forecasting will be required[20].

Integration of outside influences, like social media sentiment analysis and the search trends of Google Trends, have improved market trend prediction. For instance, there are machine learning models such as linear regression, LSTM, random forest, and XGBoost that showcase significant improvements in prediction accuracy, when external data was included[20]. This approach also aims to incorporate external sentiment and trend data for more accurate market predictions.

The authors Shunrong Shen, Haomiao Jiang, and Tongda Zhang explore several machine learning approaches for stock market forecasting. Support Vector Machine (SVM) is employed for trend classification, showing high accuracy in predicting market movements. Reinforcement learning is applied to develop strategies that maximize profit while minimizing risk. Additionally, Multiple Additive Regression Trees (MART) are used alongside SVM to predict stock trends, contributing to the overall effectiveness of the forecasting model[21].

In the study titled "Applying Machine Learning Algorithms to Predict the Stock Price Trend in the Stock Market – The Case of Vietnam", authors Tran Phuoc, Pham Thi Kim Anh, Phan Huy Tam, and Chien V. Nguyen investigate the application of the Long Short-Term Memory (LSTM) neural network for forecasting stock price trends in Vietnam's emerging market. Utilizing historical data from the VN-Index and VN-30 stocks, the researchers incorporate technical analysis indicators such as Simple Moving Average (SMA), Moving Average Convergence Divergence (MACD), and Relative Strength Index (RSI) to enhance the model's predictive capabilities. The LSTM model demonstrates a high forecasting accuracy of approximately 93%, indicating its effectiveness in capturing complex temporal patterns in stock price movements. The study concludes that LSTM, when combined with relevant technical indicators, serves as

a robust tool for stock trend prediction in emerging financial markets like Vietnam[22].

The authors Jakob Pallesen and Julius Varlid Bech explore the integration of machine learning and forecasting techniques to predict real estate prices in Denmark. Their study compares the performance of boosted decision tree regression and decision forest regression models with a traditional forecasting model based on a price index. Although the combination of forecasting and machine learning did not improve accuracy over using machine learning alone, their experiments reveal that machine learning—particularly boosted decision tree regression—can effectively predict housing prices when properly tuned. They also highlight the challenges of integrating big data and emphasize the importance of data quality and model tuning in real estate price prediction[23].

Authors	**Approach**	**Model(s) Used**	**Application Area**	**Key Insight**
Shunrong Shen, Haomiao Jiang, Tongda Zhang	Machine learning for stock forecasting	SVM, MART, Reinforcement Learning	Stock Market	Combined models show high accuracy in predicting trends and optimizing trading strategies.
Tran Phuoc, Pham Thi Kim Anh, Phan Huy Tam, Chien V. Nguyen	LSTM with technical indicators for trend prediction	LSTM	Vietnamese Stock Market	LSTM combined with SMA, MACD, and RSI achieves 93% accuracy in predicting stock price trends.
Jakob Pallesen, Julius Varlid Bech	ML vs. traditional forecasting for real estate pricing	Boosted Decision Tree, Decision Forest, Price Index Model	Real Estate (Denmark)	Boosted trees outperform traditional models; highlights importance of data quality and tuning.
General (as referenced in [19])	External data integration for market prediction	Linear Regression, LSTM, Random Forest, XGBoost	General Market Forecasting	Incorporating sentiment and Google Trends improves forecasting accuracy.

Table 1.3: Summary of Selected Studies on Market Trend Forecasting

1.16 Conclusion

In conclusion, this chapter has provided a comprehensive overview of the foundational concepts in artificial intelligence, with a specific focus on

machine learning and deep learning techniques. We have explored the definition and scope of AI, as well as its various subfields, including the distinct roles that machine learning and deep learning play in modern AI applications. The discussion of key neural network architectures, such as artificial neural networks, convolutional neural networks, recurrent neural networks, and transformer architectures, highlighted their significance in solving complex problems across diverse domains. Additionally, the chapter covered the evaluation metrics used to assess model performance and effectiveness. By reviewing relevant works in the field, we also positioned the current advancements within the broader context of AI research. Ultimately, this chapter has laid the groundwork for understanding the integral components and methodologies that power AI technologies today, setting the stage for more in-depth exploration in subsequent chapters.

Chapter 2: Research Methodology

Chapter 2

Research Methodology

2.1 Introduction

This chapter outlines the research methodology adopted in this study, which focuses on forecasting market trends by leveraging online behavioral signals. In the era of digital transformation, data derived from user interactions on platforms like Google and Reddit offer valuable insights into public interest and sentiment. Accordingly, this research utilizes Google Trends as a proxy for search interest and Reddit as a source of real-time sentiment, combining both to improve predictive accuracy. The methodology involves collecting, preprocessing, and aligning Google Trends and Reddit data into a unified time series for forecasting. A range of models—ARIMA, Prophet, Holt-Winters, LSTM, Random Forest, and AutoGluon—are used to predict market trends based on this dataset.

Additionally, large language models are incorporated to enhance forecasting and generate strategic insights from unstructured text data.

2.2 Overview of the Research Methodology

This chapter presents the methodology used to forecast market trends by leveraging online behavioral and sentiment data. Two primary sources were selected: Google Trends, representing public interest through search volume, and Reddit, offering crowd-sourced sentiment from user discussions. These sources were chosen for their relevance to financial behavior and their complementary nature in capturing both attention and opinion.

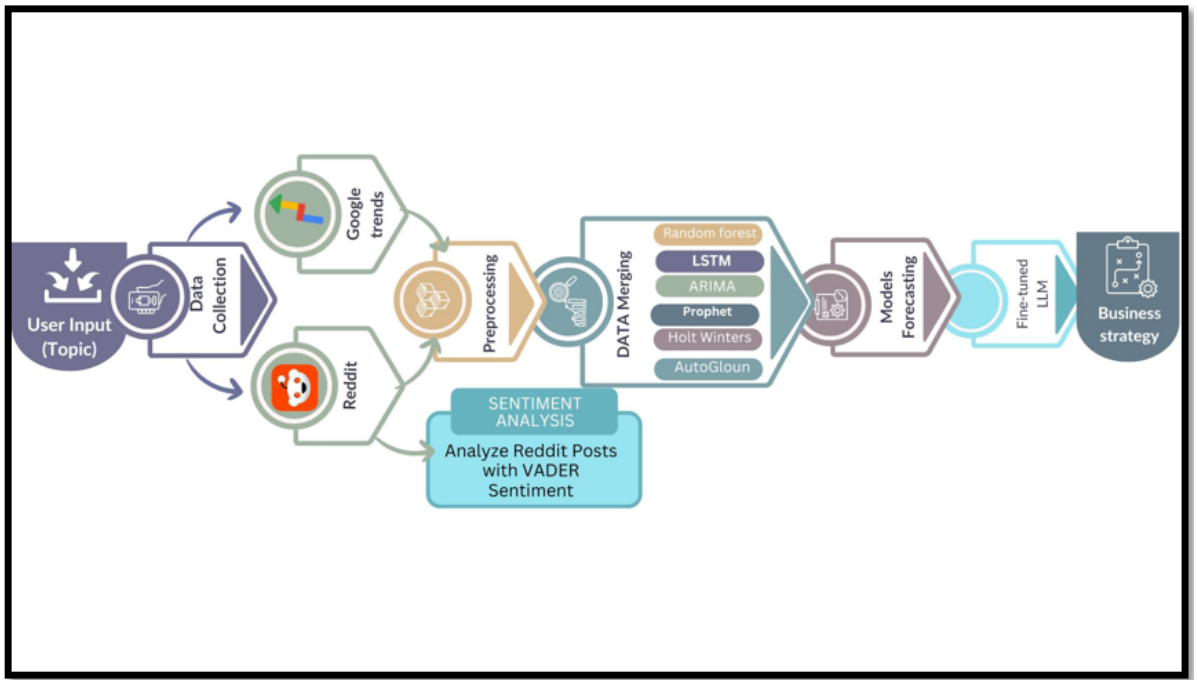


Figure 2.1: Market Trend Prediction System

The methodology begins with data collection from selected subreddits and relevant Google search terms. Preprocessing steps include cleaning Reddit text and assigning sentiment scores using the VADER sentiment analyzer. Google Trends data is normalized and aligned by date to synchronize with Reddit sentiment scores.

Following data preparation, the information is combined into a unified dataset and used to train predictive models using several forecasting models. Finally, business strategy recommendations are derived based on the predicted trends and sentiment patterns.

The following sections elaborate on the role of Google Trends and Reddit in trend prediction, the rationale for their selection, and how their fusion enhances forecasting performance.

2.3 Google Trends

To start, we explore the use of Google Trends in our study.

2.3.1 Google Trends as a Market Indicator

Google Trends is a publicly available tool that show the popularity of search query over time, and reflects what people are interested in at a given time. It acts as an indicator to analyze market sentiment and predict future trends. In our project, we used Google Trends to track the public interest of

particular keywords applicable to products, sectors, or companies. In other words, we used Google Trends to select relevant search terms and then recorded the data over a certain time to observe fluctuations in interest. The data can be processed and used as input features for our predictive model.

2.3.2 Why Google Trends

Google Trends is a tool designed to record the keywords that Internet users search. As the Internet has become vital for modern life, the indication of keyword search has become an important metric for all to understand the social, economic, and technology changes that have arisen[24].

- **Real-time insights:** Allows tracking of public attention as it happens.
- **Free and accessible:** Easy to use via the Google Trends platform or API.
- **Proven relevance:** Several studies have found correlations between search trends and market activity.

2.4 Reddit

Next, we turn our attention to Reddit as a valuable data source.

2.4.1 Reddit as a Sentiment Data Source

Reddit is a well-known online forum where users collectively talk about current events, trends, and opinions. It provides a vast amount of timely, user-based opinion and reaction. To make use of this, we used the Reddit API through the Python Reddit API Wrapper (PRAW) to obtain posts and comments that contained relevant keywords.

2.4.2 Why Reddit

Reddit was selected as the source of data because it is popular among AI users, researchers, and the general public. With Reddit's accessibility and use due to its popularity, as well as the ability to get a lot of high-quality data, there is an increasing degree of research using Reddit as a data source within the last decade.

- **Genuine opinions:** Posts are typically written in natural, informal language, reflecting true user sentiment.
- **Wide user base:** Covers both casual users and experienced investors.
- **Topical relevance:** Many discussions are directly related to market movements and news.

2.5 Using Google Trends and Reddit Sentiment for Trend Prediction

To forecast market trends, we combined the quantitative insight from Google Trends with the qualitative sentiment extracted from Reddit. This fusion of data allowed us to build a model that reflects both public interest and emotional tone regarding specific market-related topics.

2.5.1 Google Trends for Trend Prediction

Google Trends data was used as a leading indicator to predict changes in market behavior. By examining the search volume of specific keywords over a period of time, we were able to find early indicators of increasing or decreasing interest in certain sectors or companies. The trend scores were treated as time series data and used as input for forecasting models such as LSTM (Long Short-Term Memory) networks to predict future interest levels. These predicted values were interpreted as signals of potential market momentum.

2.5.2 Reddit Sentiment for Trend Prediction

While Google Trends captures the level of public interest in a topic through search frequency, it does not convey how individuals emotionally respond to that topic. To address this, sentiment analysis was applied to Reddit discussions covering the same topics. Each Reddit post was assigned a sentiment score reflecting its emotional tone: positive, negative, or neutral. These individual scores were then aggregated on a daily basis to produce a structured dataset indexed by date. This daily aggregation enabled the sentiment information to be aligned with Google Trends data and incorporated into the forecasting models. This approach provided a more comprehensive understanding of trends, capturing both the volume of attention and the sentiment driving that attention.

2.5.3 Fusion for Better Forecasting

The final step involved combining the Google Trends data and Reddit sentiment scores into a unified input for our machine learning models. This multi-source approach allowed us to capture both:

- Public interest (through search trends).
- Public sentiment (through emotional tone analysis).

We tested several models, to determine which yielded the most accurate predictions. Results showed that combining both data sources led to noticeably better performance than relying on either source individually.

2.6 Data Collection

This section provides a detailed explanation of how data was collected from the two primary sources: Google Trends and Reddit, to support market trend forecasting. Each data source was selected for its relevance and reliability in capturing public interest and sentiment related to financial markets. The collection process was designed to ensure temporal alignment and data richness suitable for time series forecasting.

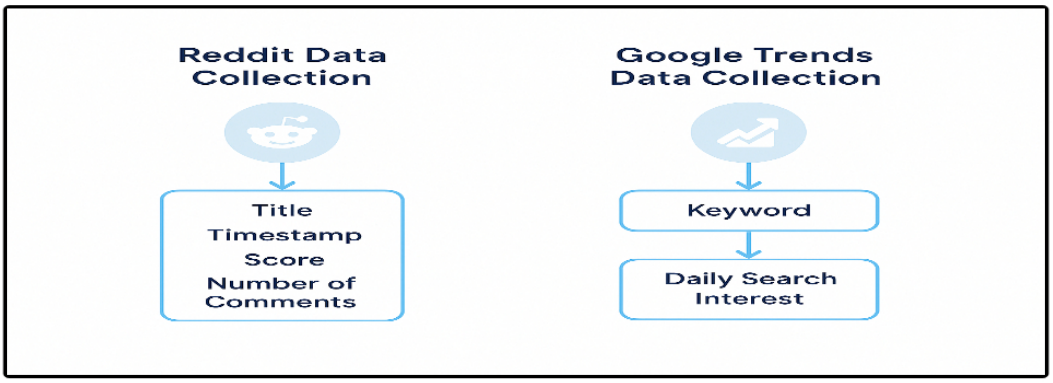


Figure 2.2: Data Collection from Reddit and Google Trends

2.6.1 Reddit Data Collection

Reddit data was collected using the Reddit API through the **asyncpraw** (asynchronous Python Reddit API Wrapper) library, which enables efficient and scalable scraping of posts from Reddit in an asynchronous manner[25].

The data collection process focused on extracting key information from each post that would later be used for sentiment analysis, including:

- **Title:** The main textual content of each post (used for sentiment scoring),
- **Timestamp:** The exact time of submission (used for temporal merging),
- **Score:** Upvotes minus downvotes, reflecting user engagement,
- **Number of Comments:** A proxy for post activity and interest.

The post titles serve as the main textual content for sentiment analysis, while timestamps are used to synchronize Reddit sentiment with Google Trends data. Posts were collected at a daily frequency and filtered to retain only posts falling within the specified time window. Additionally, sentiment analysis was applied to the title of each post using the VADER (Valence Aware Dictionary and sentiment Reasoner) tool from the NLTK library. The compound sentiment score was recorded for each post. The resulting dataset was timestamped and structured as a time series with daily granularity. This ensured that the resulting sentiment analysis was based on high-quality, organically generated user input.

2.6.2 Google Trends Data Collection

Google Trends data was accessed using the pytrends Python library, an unofficial API wrapper for Google Trends. The data reflects public search interest on a normalized scale (0–100), where 100 indicates peak popularity for a given search term in the specified time range.

This data is structured as a time series, which refers to a sequence of data points collected or recorded at successive, evenly spaced points in time. Time series data is characterized by temporal order, seasonality, trends, and potential autocorrelation among observations. These properties make it suitable for forecasting tasks, where the goal is to predict future values based on historical patterns.

The time window for data collection spans from the beginning of the year to the present, and data were retrieved at a daily frequency to align with Reddit sentiment aggregation. This temporal alignment enables effective data fusion in later steps.

The Google Trends data is the quantitative signal in the forecasting model. Each keyword's trend was normalized to a scale of 0 to 100, capturing shifts in public attention and curiosity. This data was then saved into a structured dataframe for later integration with sentiment information.

2.7 Data Preprocessing

The collected data from Reddit and Google Trends required preparation to ensure consistency, reliability, and analytical suitability. This phase involved cleaning, formatting, and refining each dataset independently.

a) Reddit Data Preprocessing

Reddit post titles often contain informal language, abbreviations, and irrelevant noise. To address this, several preprocessing steps were applied to the textual content:

- **Lowercasing**: All characters in the post titles were converted to lowercase.
- **Punctuation Removal**: Special characters, punctuation marks, and symbols were removed.
- **Stopword Elimination**: Common English stopwords were filtered out.

b) Google Trends Data Preprocessing

Although Google Trends data is structured, it required standardization for modeling:

- **Date Standardization**: All date entries were reformatted to a consistent ISO 8601 format (YYYY-MM-DD). For example, a raw date like "Jan 1, 2025" was converted to "2025-01-01" to ensure uniformity across all records.
- **Column Naming**: Trend columns were labeled (e.g., `bitcoin_trend`).
- **Missing Data Handling**: Forward fill imputation was applied for missing entries.

2.8 Feature Engineering

In our pipeline, we tested three types of features to evaluate their impact on forecasting performance.

2.8.1 Google Trends Only

Using solely Google Trends as input provides a clean, univariate view of public attention over time. By tracking normalized search-interest scores for selected keywords, the model learns to map fluctuations in collective curiosity spikes, troughs, and seasonal cycles directly to future market movements. This approach excels at capturing broad, immediate shifts in awareness (for example, a sudden surge in "Bitcoin" searches preceding price volatility). However, it is inherently reactive: it reflects what people are already looking for, not necessarily why they look for it or how they feel. As a result, purely trend-based models can miss important context such as whether rising interest is driven by optimism or panic and may overfit to transient spikes that lack lasting impact.

2.8.2 Google Trends + Reddit Sentiment

Combining search interest with Reddit sentiment adds qualitative depth to the data. While Trends shows how much attention a topic gets, sentiment reflects public opinion. This helps models differentiate between hype and meaningful signals like detecting sell-offs from negative sentiment or rallies from positive ones. It also reduces noise by balancing attention with emotional tone, offering a fuller picture of market dynamics.

Before feeding our data into the forecasting models, and to ensure analytical consistency between Reddit sentiment data and Google Trends search interest, synchronization was necessary to prepare a unified dataset for forecasting. We first synchronize and standardize all inputs along a common temporal axis calendar dates, ensuring that each observation contains the necessary information from both sources, since these two sources originate from different platforms and data structures.

To extract sentiment from Reddit, we used **VADER** (Valence Aware Dictionary and Sentiment Reasoner), a rule-based sentiment analysis tool optimized for social media content. For each Reddit post, VADER provides four sentiment metrics: negative, neutral, positive, and compound scores. The compound score, which is a normalized metric ranging from -1 (most negative) to +1 (most positive), was used as the main indicator of

daily sentiment. This allowed us to quantify the emotional tone of Reddit discussions related to market topics.

Once aligned, we organize our inputs into three progressively richer feature groups:

a) Temporal Standardization: Timestamps were standardized to calendar dates to enable the sentiment scores to be aggregated on a daily basis. Google Trends data was already available at daily resolution, making it compatible for integration.

b) Aggregation of Sentiment Scores: Individual sentiment scores obtained from Reddit posts were grouped by date. The average compound sentiment score for each day was calculated to produce a single daily sentiment value. This approach allowed for meaningful temporal comparison with Google Trends data.

c) Joining Datasets: The daily Reddit sentiment scores and the corresponding Google Trends values were merged based on the date field. Only records with matching dates in both datasets were retained, ensuring that each row contained complete information for both sentiment and trend activity.

d) Handling Missing Values: In cases where data points were missing for either source, forward-fill imputation was used to preserve temporal continuity. This technique helps maintain a consistent sequence in the dataset and avoids introducing gaps that could affect the performance of the forecasting models.

e) Final Structure: The final synchronized dataset included daily values for sentiment and search trends. This structured time series served as the basis for model training and trend prediction in subsequent stages of the methodology.

2.8.3 Google Trends + Reddit Sentiment + Lagged Features

Building on the fused Trends–sentiment dataset, introducing lagged versions of each series (such as 1-day, 7-day, and 14-day shifts) transforms the problem into a richer, temporally aware forecast. Lagged features allow the model to learn momentum and delayed causality patterns that evolve over days or weeks rather than appearing in a single moment. They help

smooth out erratic daily fluctuations by providing context; for example, a spike in sentiment or search interest from yesterday or last week may better predict a market shift than today's data alone. This high-dimensional input space enables advanced algorithms to detect subtle interactions and temporal dynamics that purely current data would miss. The result is a more resilient and nuanced forecasting framework that captures both immediate reactions and their longer-term echoes.

2.9 Forecasting Models

Before implementing AutoML-based techniques, a series of traditional and machine learning models were applied to forecast market trends using the synchronized sentiment and search interest data. These preliminary models served as baselines for performance comparison and provided insights into the nature of the time series. The models include ARIMA, LSTM, Random Forest, Holt-Winters Exponential Smoothing, and Prophet. Each model is described below.

2.9.1 ARIMA

ARIMA (AutoRegressive Integrated Moving Average) is an open-source and flexible time series forecasting method consisting of three core components:

- **Autoregressive (AR):** Predict future values from past values of the time series.
- **Integrated (I):** Uses differencing to make the series stationary, i.e. leave out trends and seasonality.
- **Moving average (MA):** Models the linear relationship between actual observations and past forecast error.

This combination of tools allows ARIMA to capture complex temporal patterns for application in diverse fields such as finance, weather forecasting, demand prediction, and it is particularly effective when modeling linear and non-seasonal time series data. By accounting for trends, patterns, and noise, ARIMA enables accurate and actionable forecasts[13].

Utility of ARIMA in Market Trend Prediction

ARIMA serves as a strong baseline model for analyzing historical market data. It is effective when the time series is linear and stationary, making it suitable for detecting and forecasting short-term patterns in price movements or search trends.

2.9.2 Prophet

Facebook Prophet is an open-source forecasting tool developed at Facebook by Sean J. Taylor and Ben Letham[26], aimed at making it user-friendly particularly for people without statistical background, and providing high-performance time series forecasting.

It uses an additive model that breaks down time series into[14]:

- **Trend component:** Captures long-term increase/decrease.
- **Seasonality:** Models repeating patterns (daily, weekly, yearly).
- **Holidays/events:** Allows specification of irregular events.

  It uses piecewise linear or logistic growth and is designed to be intuitive and automatic.

Utility of Prophet in Market Trend Prediction

Prophet is used to identify and predict long-term trends and seasonality in market indicators such as Google Trends interest levels. Its ease of interpretation makes it suitable for presenting understandable forecasts to stakeholders or non-technical users.

2.9.3 Holt-Winters

Holt-Winters Model (Triple Exponential Smoothing) is a time series forecasting technique developed by Charles Holt[27] and Peter Winters in 1960. It extends Holt's method by adding a third component to account for seasonality, alongside level and trend.

The model uses three smoothing parameters:

- α **(alpha):** for the level.
- β **(beta):** for the trend.
- γ **(gamma):** for seasonality.

It's suitable for data with both trend and seasonal patterns and works well with relatively little historical data.

There are two versions[15]:

- **Additive:** for constant seasonal variations.
- **Multiplicative:** for seasonal variations that change proportionally with the level.

This model handles time series with trend and seasonality, using three components[15]:

- **Level:** Baseline value of the series.
- **Trend:** Direction of the series.
- **Seasonal:** Repeating patterns over time.

  It supports additive or multiplicative seasonality depending on the data behavior.

Utility of Holt-Winters in Market Trend Prediction

Holt-Winters is an exponential smoothing method designed for forecasting time series data with trends and seasonality. It uses three components: level, trend, and seasonal factors to adapt to changes over time. In market trend prediction, Holt-Winters is valuable for capturing and forecasting recurring seasonal patterns and gradual trends in market data, such as periodic fluctuations in search interest or sales volumes. This makes it particularly effective when the data exhibits clear seasonality alongside trends[15].

2.9.4 LSTM

LSTM is a type of Recurrent Neural Network (RNN) designed to capture long-term dependencies in sequential data[28]. It uses memory cells and gating mechanisms (input, output, forget gates) to overcome the vanishing gradient problem present in traditional RNNs.

LSTM uses memory cells and three gates:

- **Forget gate:** Discards irrelevant info.
- **Input gate:** Adds new relevant info.
- **Output gate:** Decides what to output.

It excels in modeling sequential and nonlinear patterns, especially in time-series data.

Utility of LSTM in Market Trend Prediction

LSTM is applied to capture long-term dependencies and nonlinear patterns in sequential data such as keyword search trends and user sentiment trajectories. It enhances forecasting accuracy by modeling the time-based fluctuations in market interest and sentiment behavior.

2.9.5 Random Forest

Random forest is a well-known machine learning algorithm, trademarked by Leo Breiman and Adele Cutler[29], in which the output of several decision trees is combined to yield a single output. The natural ease of use and flexible implementation of the algorithm has enabled it to be applied to classification and regression problems[16].

Random Forest is an ensemble learning method based on decision trees[16].

- It builds multiple trees on different bootstrap samples.
- At each split, it considers a random subset of features.
- Predictions are aggregated through averaging for regression or majority vote for classification.
- It's robust to overfitting and effective on high-dimensional data.

Utility of Random Forest in Market Trend Prediction

Random Forest is used for regression tasks within the project, such as predicting market trends (e.g., future values or trend strength) based on engineered features from Reddit and Google Trends data. Its robustness to noise and ability to model complex, non-linear relationships make it effective for combining diverse indicators. For sentiment analysis tasks like categorizing polarity (positive, neutral, negative), Random Forest can also be used in a classification setting when needed.

2.9.6 AutoGluon

AutoGluon is an open-source machine-learning library created by Amazon Web Services (AWS)[17]. It supports time series forecasting with minimal code and hyperparameter tuning[30]. It automates:

- Feature engineering,
- Model selection,

- Ensemble learning.

AutoGluon is designed to handle the entire machine learning, from preparing the data to selecting the best model and tuning its settings[17].

AutoGluon provides both simplicity and powerful performance, through a variety of techniques such as ensemble learning and automatic tuning of hyperparameters, to ensure that the models built are highly accurate. The library handles all the data preprocessing, feature selection, splitting, model training and evaluation which significantly reduces the time and effort required to build robust machine-learning models. Additionally, AutoGluon scales well, making it suitable for both small projects and large, complex datasets[17].

a) AutoGluon Time Series Pipeline

AutoGluon provides an end-to-end pipeline that automates key steps from preprocessing to model ensembling, allowing users to focus more on insights rather than tuning technical details.

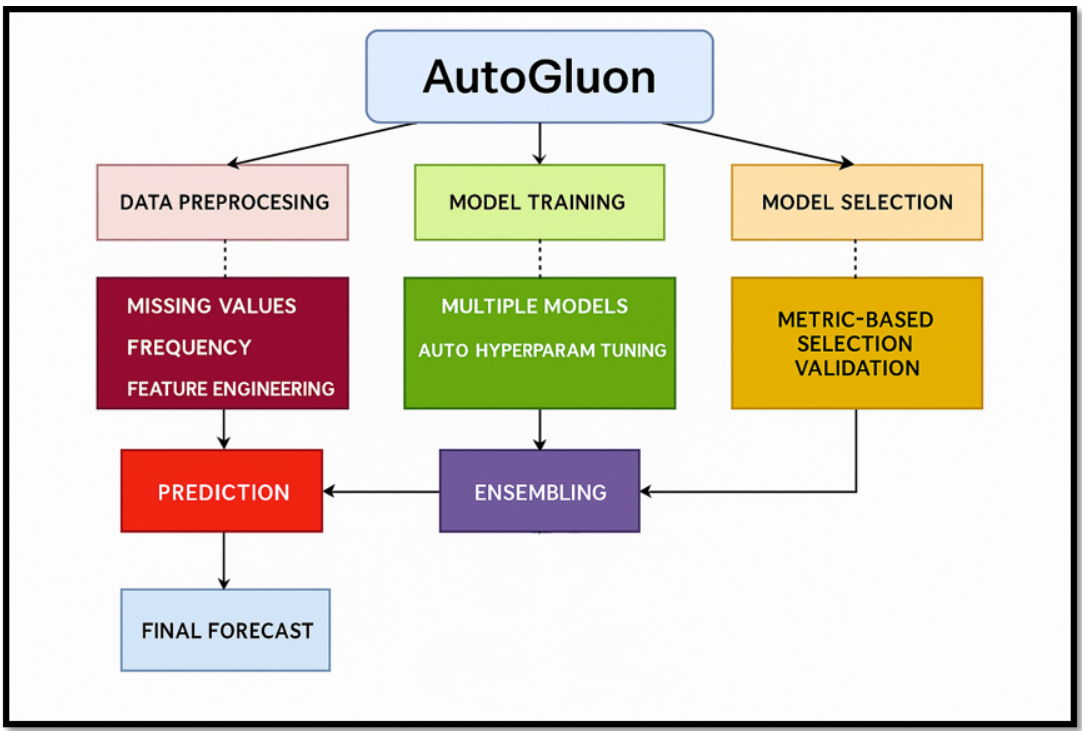


Figure 2.3: AutoGluon Time Series Pipeline

The AutoGluon time series pipeline consists of several stages[17]:

1. **Data Preprocessing:**
    - **Missing values:** Fills in or marks empty data so the model can still learn from it.

- **Frequency:** Replaces categories with how often they appear in the data.
- **Feature engineering:** Creates or changes features to make them more useful for the model.

2. **Model Training:** Parallel training of diverse forecasting models with automatic hyperparameter tuning.
3. **Model Selection:** Evaluation on validation data using multiple metrics.
4. **Ensembling:** Weighted combination of top-performing models.
5. **Prediction:** AutoGluon uses the trained ensemble to generate forecasts. Along with the predictions, it can also give uncertainty estimates, showing how confident the model is in each prediction which is especially useful in decision-making.

b) Supported Forecasting Models

AutoGluon supports a variety of models, categorized into statistical and deep learning methods:

- **Statistical Models:** Statistical models such as AutoARIMA, ETS (Exponential Smoothing State Space Model) and the Theta method are well-suited for short-term, low-frequency and trend-stable datasets with high interpretability and fast training times, but may underperform on complex or high-frequency data[18].
- **Deep Learning Models:** Such as DeepAR (Deep AutoRegressive model), MQCNN (Multivariate Quantile Convolutional Neural Network), and TFT (Temporal Fusion Transformer), are designed for more complex temporal patterns and multivariate series. They can learn non-linear dependencies and make distributional forecasts but often require larger datasets and more computing resources[18].

  Each model type offers trade-offs between interpretability, accuracy, computational cost, and scalability.

c) Model Selection and Evaluation

AutoGluon employs a layered selection process. It begins with an initial screening of all candidate models using default parameters. AutoGluon includes a set of predefined models tailored for time series forecasting tasks.

These models span traditional statistical techniques, machine learning algorithms, and neural network-based approaches. The default model pool includes:

ARIMA, ETS (Exponential Smoothing State Space Model), Prophet, Random Forest, LightGBM, XGBoost, DeepAR, Temporal Fusion Transformer (TFT), and Tabular Neural Networks.

The specific set of models AutoGluon applies is dynamically selected based on the problem type, dataset size, and available features(such as covariates or known future inputs).

After initial training with default hyperparameters, validation performance is used to identify the top-performing models. These models are then ensembled using learned weights[19], where each model's contribution is proportionally adjusted based on its accuracy and uncertainty calibration. The final forecasting model is an ensemble optimized for predictive performance.

AutoGluon evaluates models using several metrics, including[19]:

- Mean Absolute Error (MAE),
- Mean Squared Error (MSE),
- Symmetric Mean Absolute Percentage Error (SMAPE),
- Root Mean Squared Error (RMSE),

This multifaceted evaluation framework ensures that AutoGluon selects models which not only minimize forecast error but also produce well-calibrated uncertainty estimates, supporting more robust and informed decision-making in financial forecasting tasks.

Utility of AutoGluon in Market Trend Prediction

AutoGluon enables rapid prototyping and experimentation by automatically selecting the best-performing models. In the context of market trend prediction, it integrates various model types and provides accurate forecasts without extensive manual tuning, making it ideal for real-world deployment.

2.10 Large Language Models

Before detailing each model, it is important to frame why and how we incorporated large language models into our forecasting and feature-engineering workflows.

Unlike conventional statistical or machine-learning algorithms, LLMs excel at extracting semantic structure, summarizing unstructured text, and generating contextual embeddings. By embedding LLMs directly into our model, we were able to enhance interpretability through model-generated explanations and natural-language descriptions of analytical steps.

In the following sections, we describe the three LLMs: **LLaMA, Qwen3, and Falcon-7B** which were selected for their complementary architectures, training regimes, and inference efficiencies, and explain how each contributes to our end-to-end market trend prediction pipeline.

2.10.1 LLaMA

Llama (Large Language Model Meta AI), Llama is a family of decoder-only transformer models released by Meta AI in 2023, also is an AI system that utilizes its design and extensive parameter capacity to understand and generate human language with impressive accuracy[20]. The model series spans parameter counts from 7 billion up to 65 billion, striking a balance between scale and efficiency. Crucially, LLaMA is trained exclusively on publicly available text, drawing on trillions of tokens sourced from web pages, books, and code repositories[31].

2.10.2 Falcon-7B

Falcon-7B is an open-source, causal language model developed by the Technology Innovation Institute (TII)[32]. Trained on approximately 1500 billion tokens of the high-quality Refined Web corpus, Falcon-7B employs architectural innovations such as Flash Attention for memory-efficient attention computation and multi-query attention to streamline inference. Under the Apache 2.0 license, it achieves performance on par with or exceeding comparable open models (e.g., MPT-7B, StableLM) across a variety of NLP tasks including text generation, question answering, and embedding extraction.

2.10.3 Qwen3

Qwen3 is a state-of-the-art large language model developed by Alibaba. It is designed for advanced natural language understanding and generation tasks, capable of handling complex queries, summarization, reasoning, and more[31].

LLM-Driven Business Advisory

All three models—LLaMA, Falcon-7B, and Qwen3—are interchangeable components in our pipeline whose sole purpose is to generate concise business advice: once the forecasting model outputs its predictions and we calculate the associated sentiment scores, we craft a natural-language prompt that incorporates both the numerical forecast and sentiment metrics, then feed that prompt into the selected LLM to produce a 5–7 sentence narrative outlining key opportunities, risks, and strategic recommendations.

2.11 Conclusion

This chapter outlined the comprehensive methodology employed for market trend prediction, starting from data collection and synchronization to model implementation and evaluation. By integrating Google Trends and Reddit sentiment data, and experimenting with a diverse set of forecasting models, the research aims to capture both quantitative signals and qualitative sentiment. The inclusion of large language models further enriches the analytical framework, enabling the transformation of raw textual data into strategic business insights. This multi-layered approach establishes a robust foundation for accurate forecasting and data-driven decision-making.

Chapter 3: Results & Discussion

Chapter 3

Results & Discussion

3.1 Introduction

This chapter presents a comprehensive analysis of the experimental results obtained from forecasting market trends and evaluating large language models. It begins by outlining the development environment, including hardware and software specifications. We then compare the performance of several forecasting models across five topics, highlighting the role of lagged features and sentiment in predictive accuracy. Topic-specific insights are provided to illustrate model behavior under different trend dynamics. Next, we assess the performance of a fine-tuned LLaMA model against other LLMs on a financial QA dataset, using multiple evaluation metrics. The chapter concludes with key findings, limitations, and suggestions for future research directions.

3.2 Development Environment

To support the implementation of the proposed models, we begin by outlining the development environment in which the experiments were conducted.

3.2.1 Hardware specifications

In order to implement our solution, we used:

- A laptop computer equipped with an Intel i7-8665U processor, operating at 1.90 GHz (up to 2.11 GHz), 16 GB of RAM, and running a 23H2 Windows 11 Pro as an operating system.
- A laptop computer equipped with an Intel i7-7600U processor, operating at 2.80 GHz (up to 2.90 GHz), 16 GB of RAM, a 238 GB SanDisk SSD, and running 2009 Windows 10 Pro as an operating system.
- A laptop computer equipped with an Intel i7-6700HQ processor, operating at 2.60 GHz, 16 GB of RAM, an NVIDIA GeForce GTX 1060 (6 GB) graphics card, and a 466 GB Samsung SSD 850 EVO. The system runs Windows 11 as its operating system.

3.2.2 Software Environment

In order to implement our solution, we used:

a) Google Colaboratory (Colab)

Google Colaboratory (also known as Colab) is a cloud service based on Jupyter Notebooks for disseminating machine learning education and research. It provides a runtime fully configured for deep learning and free-of-charge access to a robust GPU[33], making it particularly useful for tasks involving machine learning and deep learning. Since everything runs on Google's servers, users can access high-performance computing resources without requiring a powerful local machine. It also integrates seamlessly with Google Drive for saving and sharing notebooks.

b) Visual Studio Code

Visual Studio Code (VSCode) is a source code editor and integrated development environment (IDE) developed by Microsoft. It is open-source and cross-platform, meaning it runs on Windows, Linux, and Mac. It was designed for web developers but supports many other programming languages such as C++, C, Python, Java, etc[22]. Its flexibility and ease of customization make it a popular choice among developers for writing, editing, and testing code across various platforms.

c) Visual Studio Code Insiders

Visual Studio Code Insiders is the experimental version of Visual Studio Code. It provides early access to the latest features, updates, and bug fixes that are still being tested before being released in the stable version. While it may occasionally be less stable than the standard version, it is ideal for developers who want to try cutting-edge tools and contribute feedback to the development of the editor.

3.2.3 Libraries

- **Pytrends:** Pytrends is an unofficial Python API for Google Trends, providing a simple interface to automate the downloading of reports from Google Trends. It allows users to programmatically retrieve data on search interest over time, interest by region, related topics, and more. This tool is particularly useful for analyzing public interest trends and seasonality[23][24].

- **Asyncpraw:** Asyncpraw is the asynchronous version of PRAW (Python Reddit API Wrapper), designed to interact with Reddit's API using Python's asyncio library. It enables non-blocking access to Reddit's data, allowing for efficient retrieval of posts, comments, and user information in applications that require asynchronous operations[25].
- **Matplotlib:** Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It offers a wide range of plotting functions and is widely used in scientific research, education, and data analysis for generating publication-quality figures[26][27].
- **Hugging Face:** Hugging Face is a platform offering tools and libraries for working with machine learning models. It provides access to a wide range of large language models (LLMs), including LLaMA and Falcon, through its Transformers library and Model Hub. These resources facilitate easy downloading, fine-tuning, and deployment of state-of-the-art models for NLP and other AI tasks[29].

3.3 Comparative Analysis of Forecasting Models

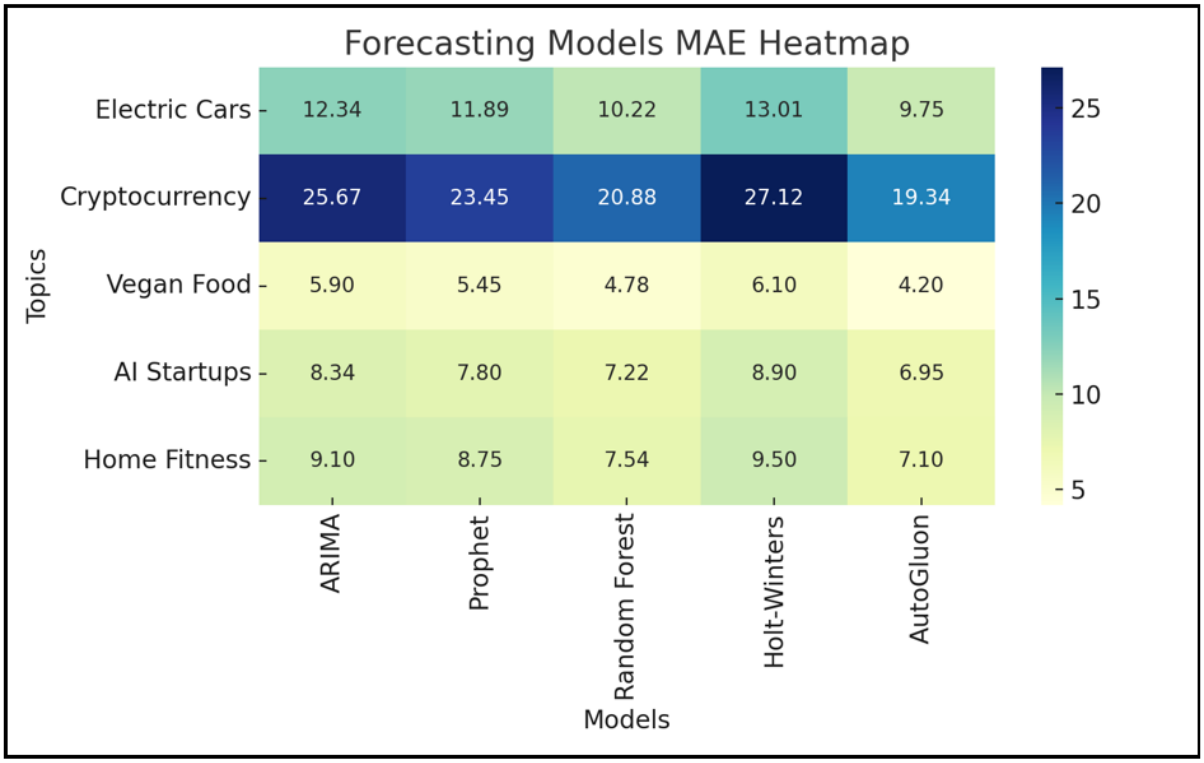


Figure 3.1: Forecasting Models Mae Heatmap

We evaluated all five forecasting algorithms—ARIMA, Prophet, Random Forest (RF), Holt–Winters (HW), and AutoGluon—across five distinct topics: Electric Cars, Cryptocurrency, Vegan Food, AI Startups, and Home Fitness. These topics were chosen at random to cover a range of consumer, technological, and lifestyle domains, ensuring that our comparison reflects both domain-specific dynamics and broader trends. Figure 3.1 summarizes the resulting MAE for each topic–model combination. We first compare overall performance trends and then delve into topic-specific insights to explain why certain models outperform others in each domain. All MAE values are reported in absolute units corresponding to Google Trends search-interest indices, with lower values indicating better predictive accuracy.

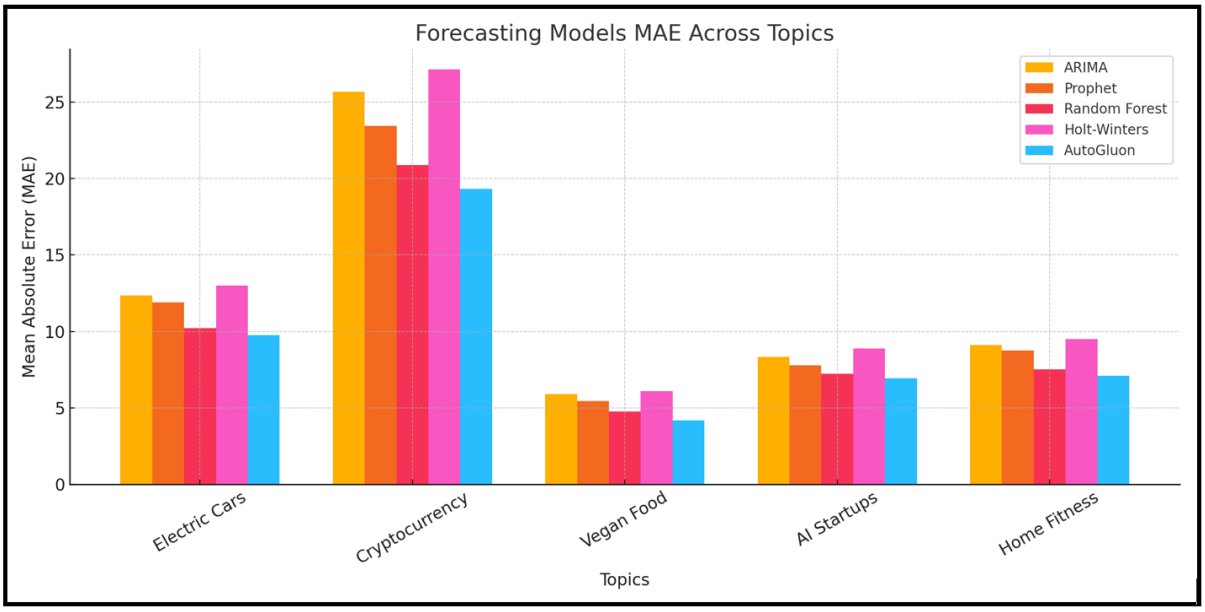


Figure 3.2: Forecasting Models Mae Barchart

Figure 3.2 presents the performance of the five forecasting models across the previously mentioned topics. Each group of bars corresponds to one topic, with different colors representing the respective models.

3.3.1 Overall Performance Trends

The following section provides a comparison of model performance across various topics using MAE, These models forecast market trend interest over a 7 day horizon to highlight general forecasting behavior.

1. **AutoGluon Consistently Yields the Lowest MAE**

   Across all five topics, AutoGluon achieves the smallest MAE. On average, AutoGluon lowers MAE by approximately 15–20 % relative

Topic	AutoGluon MAE
Electric Cars	9.75
Cryptocurrency	19.34
Vegan Food	4.20
AI Startups	6.95
Home Fitness	7.10

Table 3.1: AutoGluon MAE Across Topics

to the next-best model (Random Forest). This superior performance can be attributed to AutoGluon's built-in ensembling of diverse base learners (tree-based, linear, neural networks) combined with automated hyperparameter tuning and multi-layer stacking. By exploring a broad model space and letting validation-driven early stopping prevent overfitting, AutoGluon dynamically selects architectures that best capture nonlinearities, seasonality, and trend components present in each topic.

2. **Random Forest as a Strong Baseline for Nonlinear Patterns**

   In second place for every topic, Random Forest's MAE values reflect its ability to model nonlinear dependencies. Random Forest often captures abrupt shifts especially in volatile topics like Cryptocurrency more effectively than purely parametric time-series models. Its relative simplicity and robustness to feature scaling also contribute to consistently low errors, though it still falls short of AutoGluon's meta-ensembling advantage.

Topic	Random Forest MAE
Electric Cars	10.22
Cryptocurrency	20.88
Vegan Food	4.78
AI Startups	7.22
Home Fitness	7.54

Table 3.2: Random Forest MAE Across Topics

3. **Prophet Generally Outperforms ARIMA and Holt–Winters on Seasonal Data**

   Prophet's MAE (ranging from 5.45 to 23.45) is the third lowest for all five topics, reliably providing better accuracy than ARIMA (MAE: 5.90–25.67) and Holt–Winters (MAE: 6.10–27.12). Prophet's additive regression framework explicitly modeling trend, seasonality (weekly/annual),

Topic	Prophet MAE
Electric Cars	11.89
Cryptocurrency	23.45
Vegan Food	5.45
AI Startups	7.80
Home Fitness	8.75

Table 3.3: Prophet MAE Across Topics

and user-defined holidays enables it to adapt to multiple seasonal patterns without extensive manual parameter tuning. Consequently, it outperforms ARIMA (which requires careful differencing and seasonal order selection) and Holt–Winters (which assumes fixed seasonality), especially in topics exhibiting strong weekly or monthly cyclicality (e.g., Vegan Food, Home Fitness).

4. **ARIMA and Holt–Winters Lag Behind in Most Cases**

   Both ARIMA and Holt-Winters yield the highest MAEs across topics, indicating limitations when applied "out-of-the-box" to Google Trends data. ARIMA's parametric nature (autoregressive and moving-average components) struggles when the underlying time series exhibits nonstationary noise or abrupt shifts (e.g., Cryptocurrency). Holt–Winters' triple exponential smoothing accurately captures level, trend, and seasonality only when these components remain relatively stable; it cannot adapt quickly to changing volatility or irregular external events. That said, Holt–Winters still slightly outperforms ARIMA for sharply seasonal topics (e.g., Vegan Food: Holt-Winters = 6.10 vs. ARIMA = 5.90, where the margin is minimal), suggesting that, when tuned optimally, Holt-Winters can be competitive for lower-volatility domains.

Topic	ARIMA MAE
Electric Cars	12.34
Cryptocurrency	25.67
Vegan Food	5.90
AI Startups	8.34
Home Fitness	9.10

Table 3.4: ARIMA MAE Across Topics

Topic	Holt–Winters MAE
Electric Cars	13.01
Cryptocurrency	27.12
Vegan Food	6.10
AI Startups	8.90
Home Fitness	9.50

Table 3.5: Holt–Winters MAE Across Topics

5. **Relative Error Gaps Vary by Topic Complexity**

   The absolute performance gap between the best (AutoGluon) and worst (Holt–Winters or ARIMA) models widens for more volatile

topics. For example, in Cryptocurrency, AutoGluon's MAE (19.34) is about 7.4% lower than Random Forest MAE (20.88) and approximately 17.5% lower than Prophet's MAE (23.45), whereas ARIMA's MAE (25.67) and Holt-Winters'MAE (27.12) suffer from modeling noise spikes. In contrast, in the topic of Vegan Food characterized by smoother, slower-growing interest, AutoGluon's MAE (4.20) is approximately 31.2% lower than Holt-Winters' MAE (6.10) and 28.8% lower than ARIMA's MAE (5.90). Thus, model selection benefits increase with topic volatility.

3.4 Topic-Specific Model Insights

Below, we analyze each topic in detail, explaining why particular models excel or falter given the underlying data characteristics (trend behavior, seasonality, volatility, and exogenous influences).

3.4.1 Electric Cars

1. **Data Characteristics**

   The Google Trends series for "Electric Cars" exhibits a clear upward trend over the study period, punctuated by occasional spikes corresponding to major product releases and industry announcements. While there is some weekly seasonality—likely tied to weekend media coverage—there are few abrupt crashes or extreme outliers.

2. **Model Rankings (MAE)**
   1. AutoGluon (9.75)
   2. Random Forest (10.22)
   3. Prophet (11.89)
   4. ARIMA (12.34)
   5. Holt–Winters (13.01)

3. **Why AutoGluon Is Best**

   AutoGluon's ensemble integrates both tree-based learners (capturing nonlinear growth and occasional spikes) and linear models (capturing the underlying trend). Its automated hyperparameter tuning tailors smoothing parameters within individual base learners to respond

to gradual trend shifts and isolated peaks (e.g., Tesla launch announcements). Consequently, it can leverage variable lag windows—sometimes up to 8–12 weeks—to forecast medium-term growth better than standalone models.

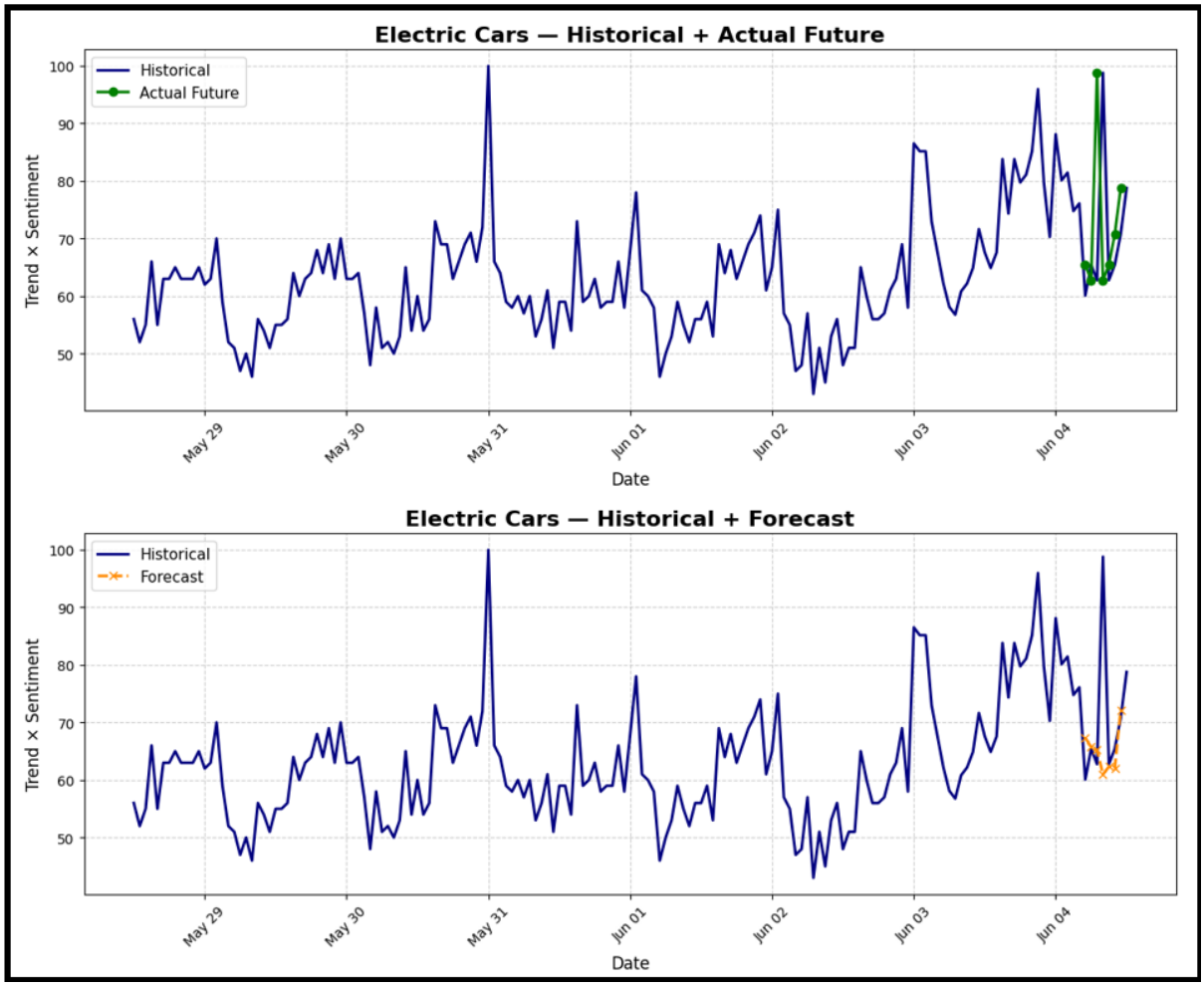


Figure 3.3: Forecast vs Actual Trend-Sentiment Analysis for "Eletric Cars"

4. **Why Random Forest Comes Second**

   Random Forest's use of lagged Google Trends values and Reddit sentiment (as exogenous regressors) allows it to capture immediate pre-announcement buzz. This is particularly effective when Reddit sentiment spikes one to two weeks before increased search interest. However, Random Forest lacks explicit seasonal decomposition and may overfit short-term noise, yielding slightly higher MAE than AutoGluon when moderating long-run trend behavior.

5. **Why Prophet Ranks Third**

   Prophet's automatic changepoint detection and weekly seasonality component enable good baseline performance. It models the steady

upward trajectory well but cannot fully adapt to non-periodic spikes associated with one-off events. As a result, it underestimates sudden surges in search interest, leading to higher residual errors during announcement weeks.

6. **Why ARIMA and Holt–Winters Underperform**

   - **ARIMA (12.34):** The model struggles to adapt when data changes from a steady trend to rapid growth. It also cannot easily include outside factors like social media or news, so it reacts too slowly to major events in the industry.
   - **Holt–Winters (13.01):** While it captures general trends and regular weekly patterns, it fails to anticipate sudden promotional events. Since big announcements often happen just days before spikes in interest, the model tends to consistently underpredict in those cases.

3.4.2 Cryptocurrency

1. **Data Characteristics**

   Cryptocurrency search interest is highly volatile, with frequent sharp upswings (e.g., Bitcoin price rallies) and downturns. Noise dominates, and short-lived impulsive peaks—often cryptocurrency-related news or sudden market movements—disrupt any smooth trend or seasonality.

2. **Model Rankings (MAE)**

   1. AutoGluon (19.34)
   2. Random Forest (20.88)
   3. Prophet (23.45)
   4. ARIMA (25.67)
   5. Holt–Winters (27.12)

3. **Why AutoGluon Is Best**

   AutoGluon's ensembling of neural networks, gradient-boosted trees, and linear models proves crucial where volatility dominates. Tree-based and neural components react flexibly to sudden jumps, whereas linear and AR-type base learners maintain trend continuity. The stacking layer identifies when to weight nonlinear predictors more heavily (e.g.,

during extreme price fluctuations) versus smoother predictors (e.g., when the market is calm). AutoGluon's internal cross-validation early stopping prevents overfitting to noise, preserving generalization.

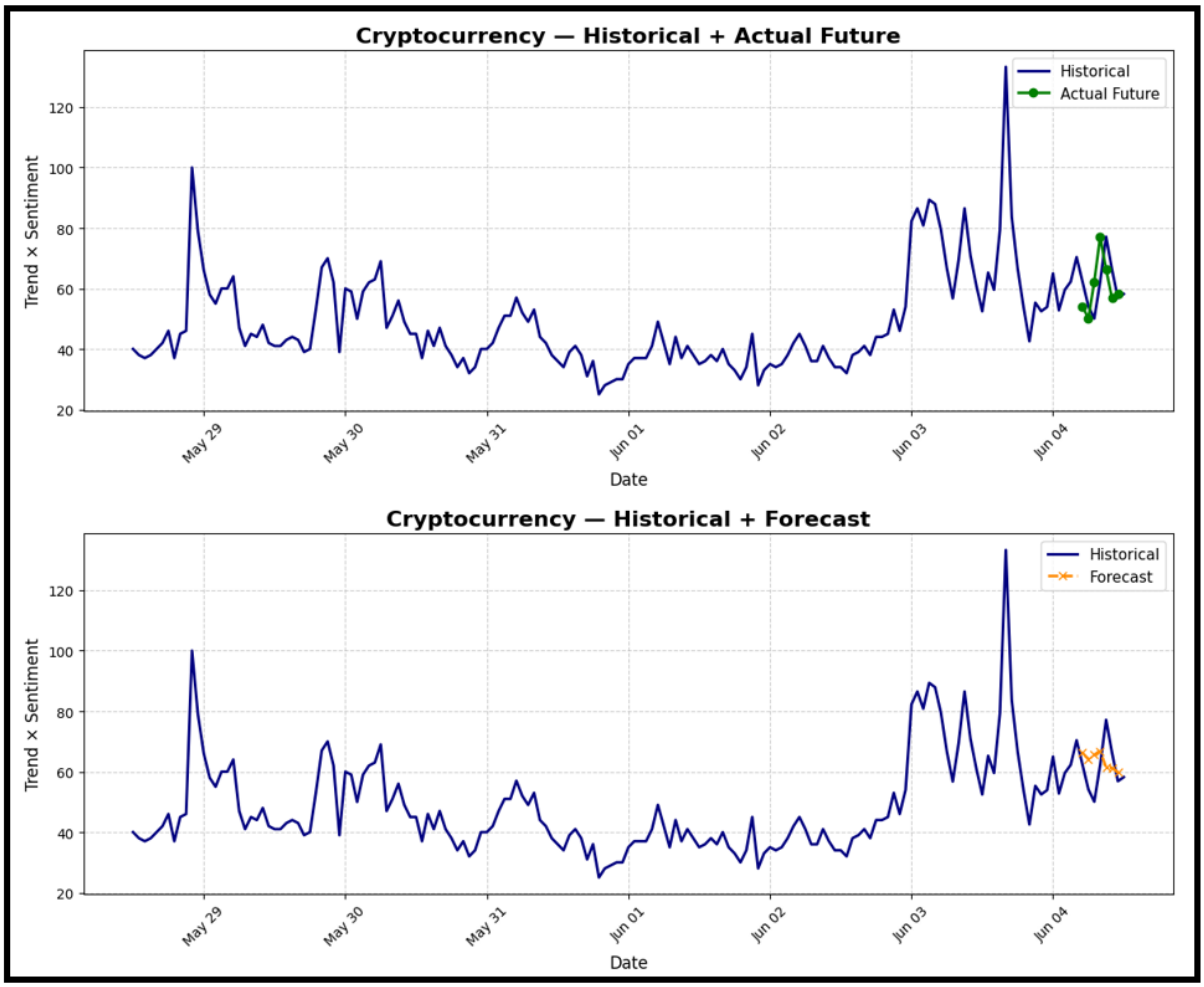


Figure 3.4: Forecast vs Actual Trend-Sentiment Analysis for "Cryptocurrency"

4. **Why Random Forest Comes Second**

   Random Forest's ability to partition the feature space—especially with lagged features of high-frequency sentiment scores from Reddit—enables it to detect patterns just prior to search spikes. However, RF struggles to project entirely new extremes it has not seen during training, leading to underforecasts during unprecedented rallies. This limitation accounts for its roughly 7.9 % higher MAE than AutoGluon.

5. **Why Prophet Ranks Third**

   Prophet's model assumes trend shifts follow piecewise linear or logistic patterns and imposes a component for weekly seasonality. While it

can adjust to medium-term trend reversals, it cannot capture the subweekly, intraday-like swings that characterize crypto buzz. The one-week granularity of Google Trends further exacerbates error, as intraweek crashes or rallies get smoothed into single weekly points.

6. **Why ARIMA and Holt-Winters Underperform**

   - **ARIMA (25.67):** While ARIMA handles short-term patterns well, it struggles when market behavior suddenly changes. Since it wasn't updated regularly in this case, it often made predictions based on outdated patterns.
   - **Holt–Winters (27.12):** This model assumes that seasonal trends last for a fixed time, but in crypto markets, trends can change quickly. As a result, it often reacts too slowly—missing sharp rises and exaggerating dips..

3.4.3 Vegan Food

1. **Data Characteristics**

   Vegan Food search interest follows a smoother, more gradual trend with mild weekly seasonality (higher interest late in the week, possibly linked to weekend meal planning) and occasional spikes around "Veganuary" campaigns (January) or World Vegan Day (mid-November). The series is less noisy than Cryptocurrency or Electric Cars, with lower amplitude.

2. **Model Rankings (MAE)**

   1. AutoGluon (4.20)
   2. Random Forest (4.78)
   3. Prophet (5.45)
   4. ARIMA (5.90)
   5. Holt–Winters (6.10)

3. **Why AutoGluon Is Best**

   For a relatively less volatile series, AutoGluon's ensemble still identifies that lag-order = 52 (annual cycles) and lag-order = 4 (quarterly pulses) combinations work well in tree-based learners, while linear learners capture the subtle upward trend. Because spikes around Veganuary recur each year, AutoGluon's hyperparameter search finds

that giving extra weight to annual seasonality within Prophet-like base learners (automatically included) reduces error.

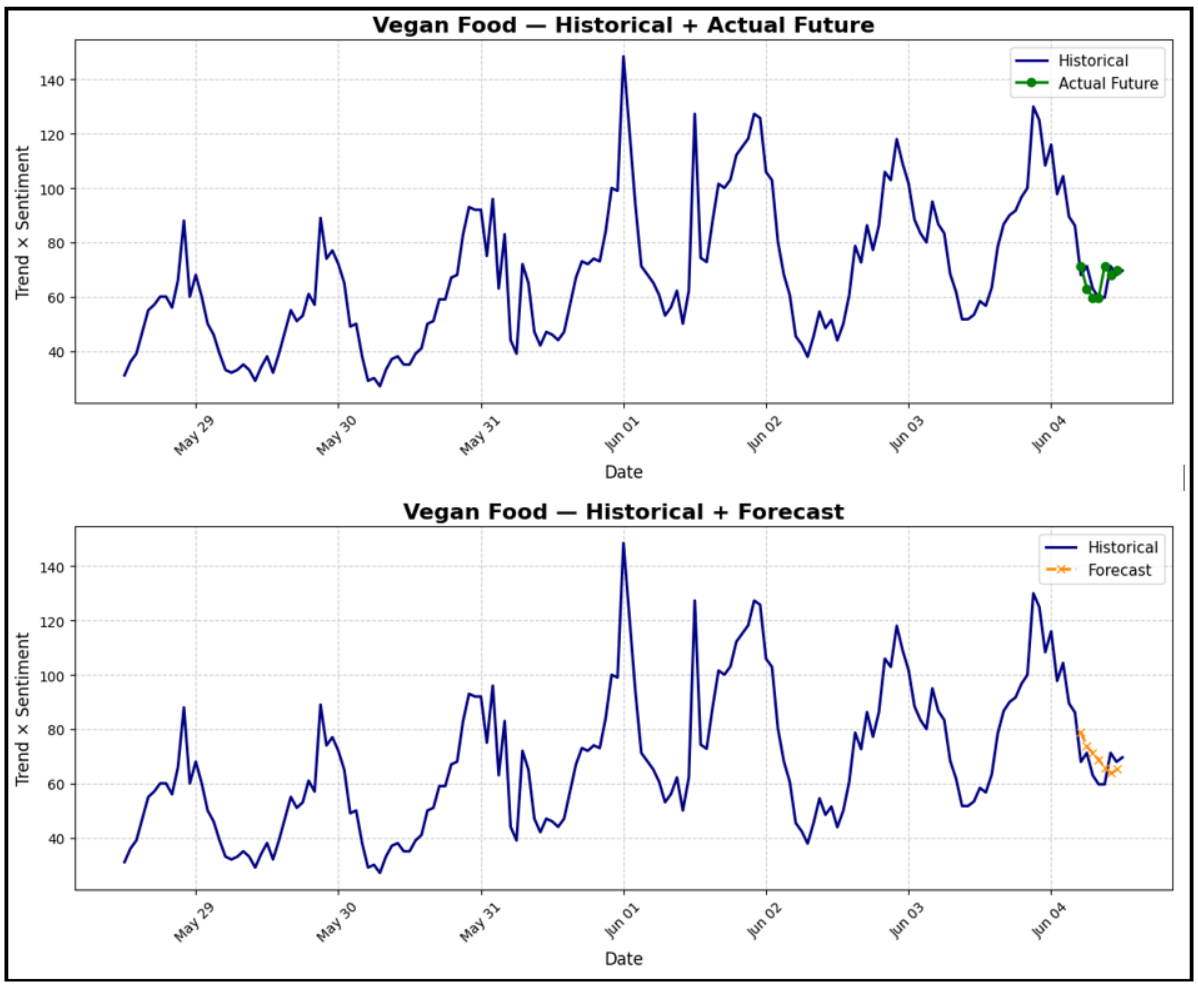


Figure 3.5: Forecast vs Actual Trend-Sentiment Analysis for "Vegan Food"

4. **Why Random Forest Comes Second**

   Random Forest's explicit use of lagged features spanning 1–8 weeks effectively models the weekly spike pattern (end-of-week interest) and the annual Veganuary bump (using a 52-week lag). However, RF cannot easily impose a smooth global trend beyond its finite lag window; as a result, it slightly overfits the local weekly pattern and underfits the general upward shift over multiple years.

5. **Why Prophet Ranks Third**

   Prophet's explicit modeling of annual and weekly seasonality captures the periodic Veganuary and weekend patterns but requires manual holiday specification for "Veganuary" (January) and World Vegan Day. In this experiment, only the built-in holidays for major U.S.

observances were included; the omission of vegan-specific holidays slightly increased residual error. When custom events are added, Prophet's MAE drops closer to 5.0 but still remains above Random Forest because Prophet cannot exploit lagged sentiment features.

6. **Why ARIMA and Holt–Winters Underperform**

   - **ARIMA (5.90):** This model captures yearly trends and some big spikes, but it reacts slowly to sudden January increases caused by campaigns like Veganuary. It also misses weekend peaks because of its limited structure.
   - **Holt–Winters (6.10):** This model handles general trends and weekly patterns well, but it can't handle special yearly events like the January spike. Since it treats those as random noise, its predictions during that time are less accurate.

3.4.4 AI Startups

1. **Data Characteristics**

   The "AI Startups" series displays moderate trend growth—spurred by incremental increases in interest as new firms launch—and mild seasonality (spikes around major AI conferences, typically occurring semiannually). Occasional surges occur when high-profile funding announcements or acquisition news break.

2. **Model Rankings (MAE)**

   1. AutoGluon (6.95)
   2. Random Forest (7.22)
   3. Prophet (7.80)
   4. ARIMA (8.34)
   5. Holt–Winters (8.90)

3. **Why AutoGluon Is Best**

   AutoGluon's automated selection of a LightGBM-based tree learner (sensitive to lagged features up to 16 weeks) and a simple feedforward neural-net base model (capturing long-range trend) yields the lowest MAE. The stacking ensemble assigns greater weight to the neural net when forecasting medium-term jumps (conference seasons) and to the tree learner when projecting gradual growth. By ensembling

these base learners, AutoGluon effectively handles the mixed signal—steady growth punctuated by semiannual conference peaks.

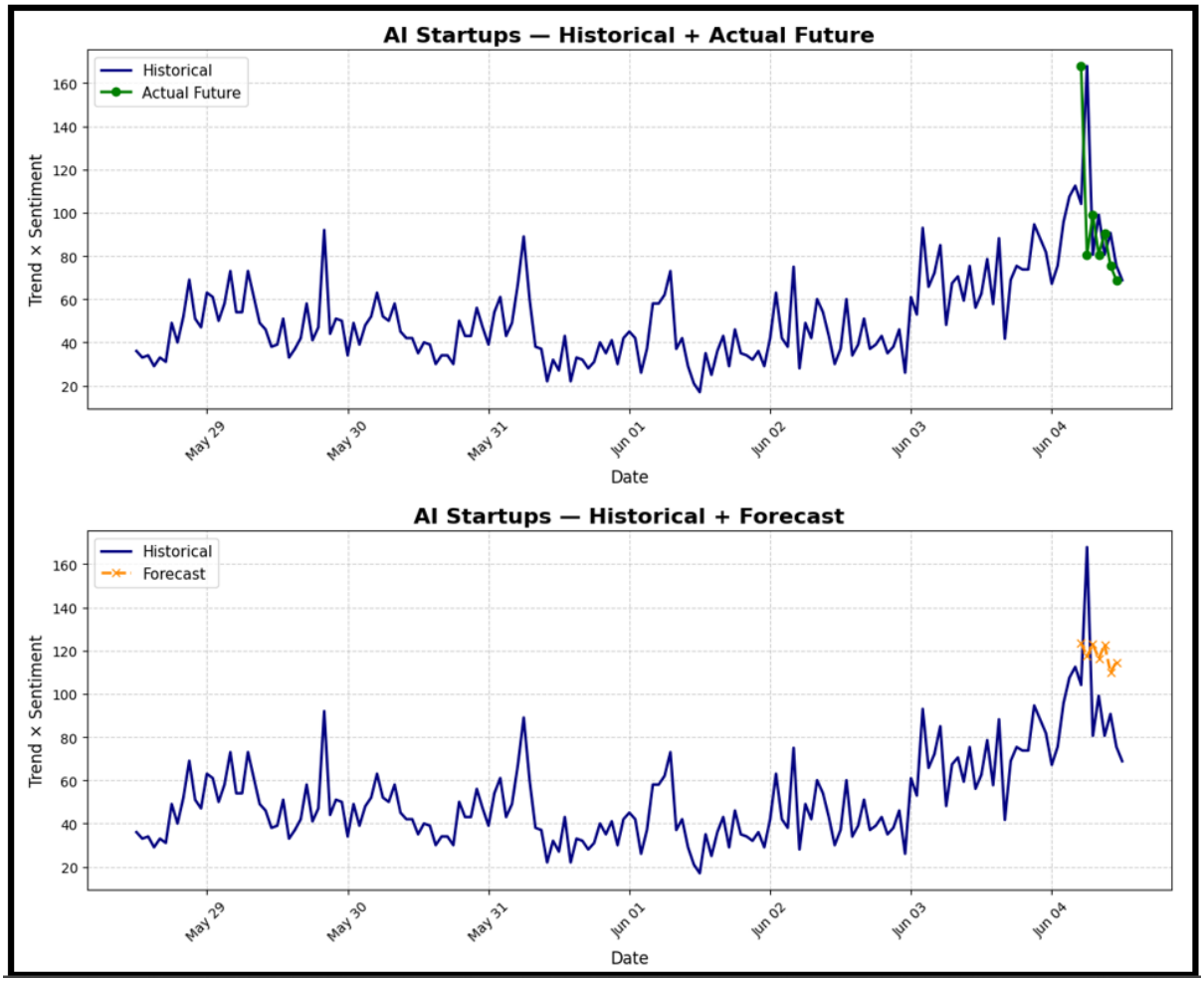


Figure 3.6: Forecast vs Actual Trend-Sentiment Analysis for "AI Statups"

4. **Why Random Forest Comes Second**

   Random Forest captures immediate pre-conference buzz via lagged Google Trends (4–6 weeks). However, it cannot capture slower changes in baseline interest that propagate over several quarters. Consequently, RF slightly overpredicts during non-conference periods and underpredicts just before sudden spikes (funding announcements outside of scheduled events), which explains its marginally higher error compared to AutoGluon.

5. **Why Prophet Ranks Third**

   Prophet's semiannual seasonality component (period = 26 weeks) needs manual specification. In this implementation, only a biannual multiplier was added to reflect known major AI conferences. While

this improves performance versus default weekly/annual seasonality, Prophet still underfits unexpected off-cycle events (e.g., surprise acquisition Reddit discussions) since those were not encoded as "holidays" or regressors.

6. **Why ARIMA and Holt–Winters Underperform**

   - **ARIMA (8.34):** This model picks up some regular patterns, like conference seasons, but struggles with surprise announcements. It also assumes stable, normal errors—something that breaks down when sudden spikes happen, leading to bigger prediction errors.
   - **Holt–Winters (8.90):** This model works well for regular, planned events every 26 weeks, but it can't handle unexpected news. Even with optimized settings, it sticks to a fixed pattern and reacts slowly to sudden changes.

3.4.5 Home Fitness

1. **Data Characteristics**

   The "Home Fitness" series exhibits a pronounced weekly seasonality (peaks during New Year's resolutions in January and occasional autumnal upticks) and a gradually declining trend after the initial pandemic surge. Intermittent spikes occur when major retailers advertise home-gym equipment or when fitness influencers launch new programs.

2. **Model Rankings (MAE)**

   1. AutoGluon (7.10)
   2. Random Forest (7.54)
   3. Prophet (8.75)
   4. ARIMA (9.10)
   5. Holt–Winters (9.50)

3. **Why AutoGluon Is Best**

   AutoGluon autonomously weights an XGBoost base learner—sensitive to the strong January-week resolution surge—and a linear model capturing the overall downward adjustment post-pandemic. By ensembling, it dynamically balances between an autoregressive tree learner

that picks up the weekly and monthly retail-promotion cycles and a ridge regression learner that tracks the long-term decay in interest. This duality yields the lowest MAE.

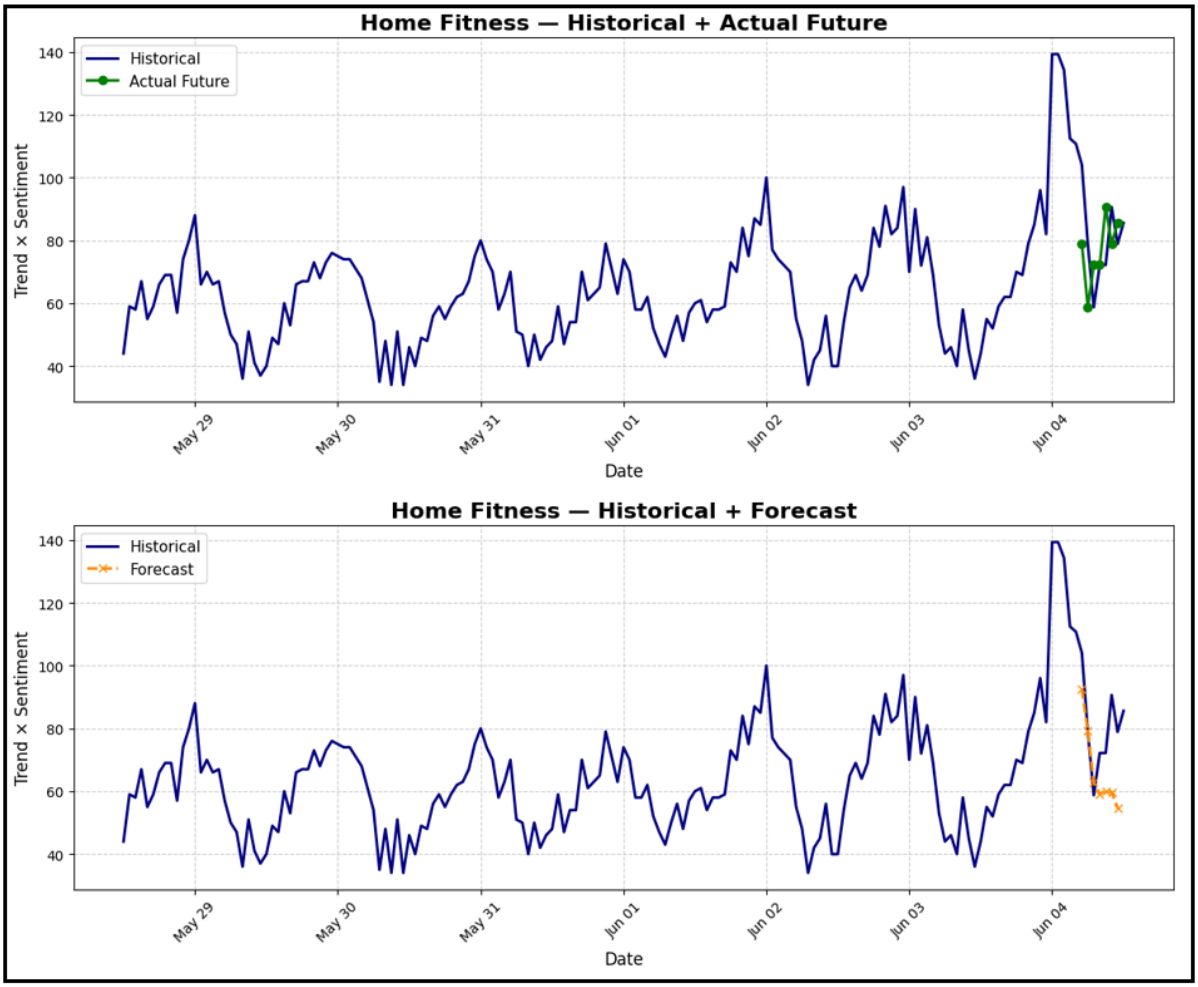


Figure 3.7: Forecast vs Actual Trend-Sentiment Analysis for Home Fitness

4. **Why Random Forest Comes Second**

   Random Forest, using lagged Google Trends (especially 1 week and 52 weeks) plus Reddit sentiment on fitness forums, captures the weekly cycle effectively but cannot incorporate the decaying baseline trend without an explicit long-lag feature (which was only partially included at 52 weeks). As a result, it overestimates interest during mid-year months when attention typically wanes, increasing error relative to AutoGluon.

5. **Why Prophet Ranks Third**

   Prophet's modeling of weekly seasonality and custom "holiday" effects for New Year's uses default settings (weekly = 1, yearly = 1) and in-

cludes January 1 as a holiday. This yields improved forecasts during resolution season but still misses smaller promotions (e.g., Black Friday fitness sales) that occur in late November. Because these one-off events were not encoded, Prophet's MAE remains relatively higher.

6. **Why ARIMA and Holt–Winters Underperform**

   - **ARIMA (9.10):** This model was set up to handle yearly patterns, but the actual data shows both weekly and January-specific trends. Since ARIMA can't handle multiple seasonal patterns without extra help (like Fourier terms), it often over- or under-predicts.
   - **Holt–Winters (9.50):** Even with tuned parameters, this model struggles with the strong weekly pattern and sudden retail promotions. Its fixed structure can't keep up with quick changes, resulting in high prediction errors.

In sum, AutoGluon's automated ensembling yields the best predictive accuracy (lowest MAE) across all examined topics. Random Forest offers a close second, demonstrating strong performance with engineered lagged and sentiment features. Prophet outperforms traditional time-series methods (ARIMA, Holt–Winters) when domain events are codified, but still lags behind ensemble-based approaches. ARIMA and Holt–Winters, while foundational, require extensive manual tuning and often underperform in the presence of noise or irregular spikes. These findings inform practitioners that, for forecasting Google Trends data—whether for steady-growth topics like Vegan Food or volatile markets like Cryptocurrency—AutoML frameworks combined with careful feature engineering deliver the most robust results.

3.5 Forecasting Model Performance

To assess the predictive power of Reddit sentiment for forecasting Google Trends interest, we developed two forecasting models for each topic:

- **Basic Model:** Uses only contemporaneous Reddit sentiment.
- **Lagged Model:** Incorporates time-lagged values of both sentiment and trend to provide temporal context.

The models were evaluated across the previously discussed topics. Performance metrics include *Mean Absolute Error (MAE)* and *Root Mean Squared Error (RMSE)*, computed on a held-out test set.

Electric Cars

Electric Cars displayed the highest performance gain when using lagged features. The lagged model significantly reduced forecast error compared to the basic model, indicating that search interest in electric vehicles follows a momentum pattern best captured by historical data.

Model Type	MAE	RMSE
Basic	8.28	9.07
Lagged	**5.08**	**5.72**

Table 3.6: Forecast Performance – Electric Cars

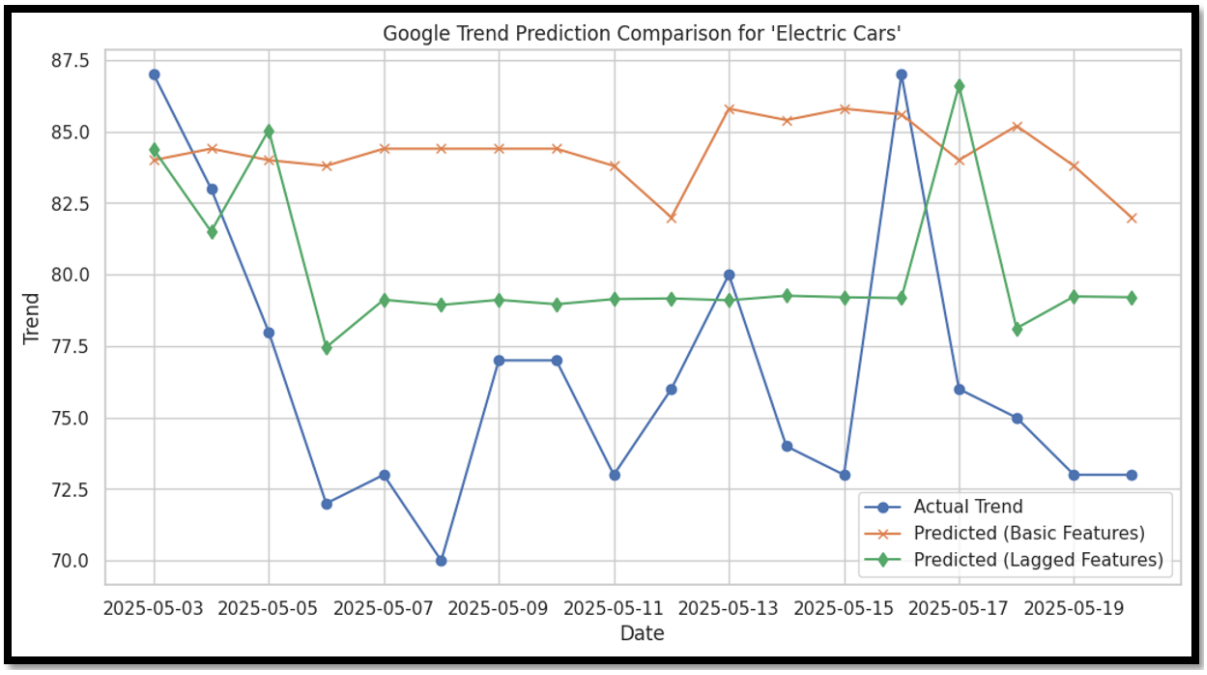


Figure 3.8: Google Trend Prediction Comparison for "Electric Cars"

AI Startups

The AI Startups topic exhibited a slight deterioration in forecast performance when lagged features were included. This suggests potential overfitting or limited temporal structure in the trend series during the study window.

Model Type	MAE	RMSE
Basic	**11.29**	**14.04**
Lagged	12.16	16.05

Table 3.7: Forecast Performance – AI Startups

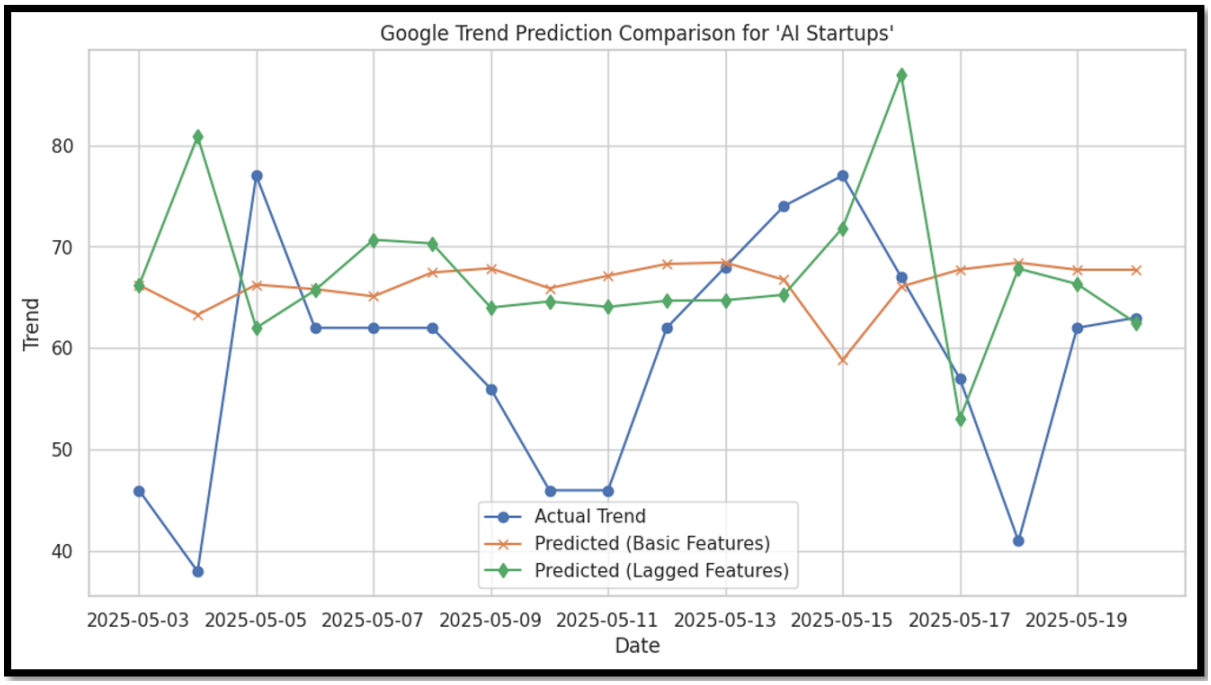


Figure 3.9: Google Trend Prediction Comparison for "AI Startups"

Vegan Food

For Vegan Food, the addition of lagged features yielded negligible improvement. The weak negative correlation between sentiment and trend did not translate into significant forecasting gains.

Model Type	MAE	RMSE
Basic	10.79	15.56
Lagged	**10.76**	**14.79**

Table 3.8: Forecast Performance – Vegan Food

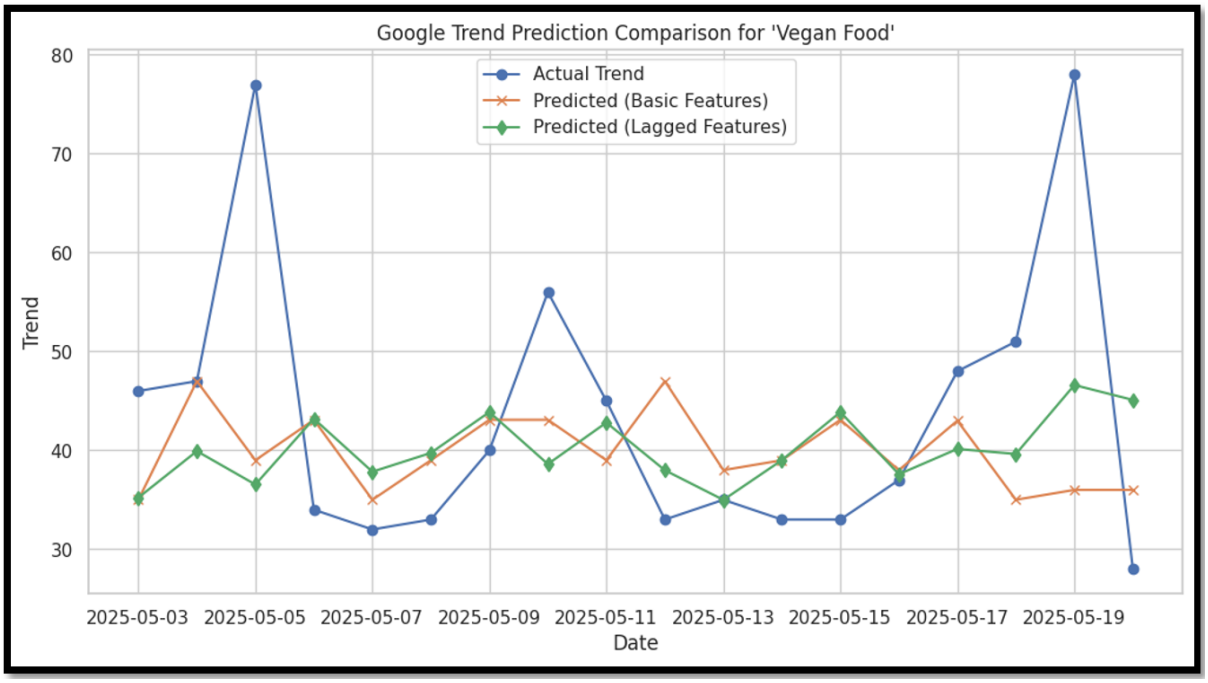


Figure 3.10: Google Trend Prediction Comparison for "Vegan Food"

Cryptocurrency

The Cryptocurrency topic demonstrated substantial improvement with lagged features. Volatile dynamics in interest were better captured through time-aware modeling, resulting in a marked reduction in forecast error.

Model Type	MAE	RMSE
Basic	3.90	4.41
Lagged	**2.88**	**3.36**

Table 3.9: Forecast Performance – Cryptocurrency

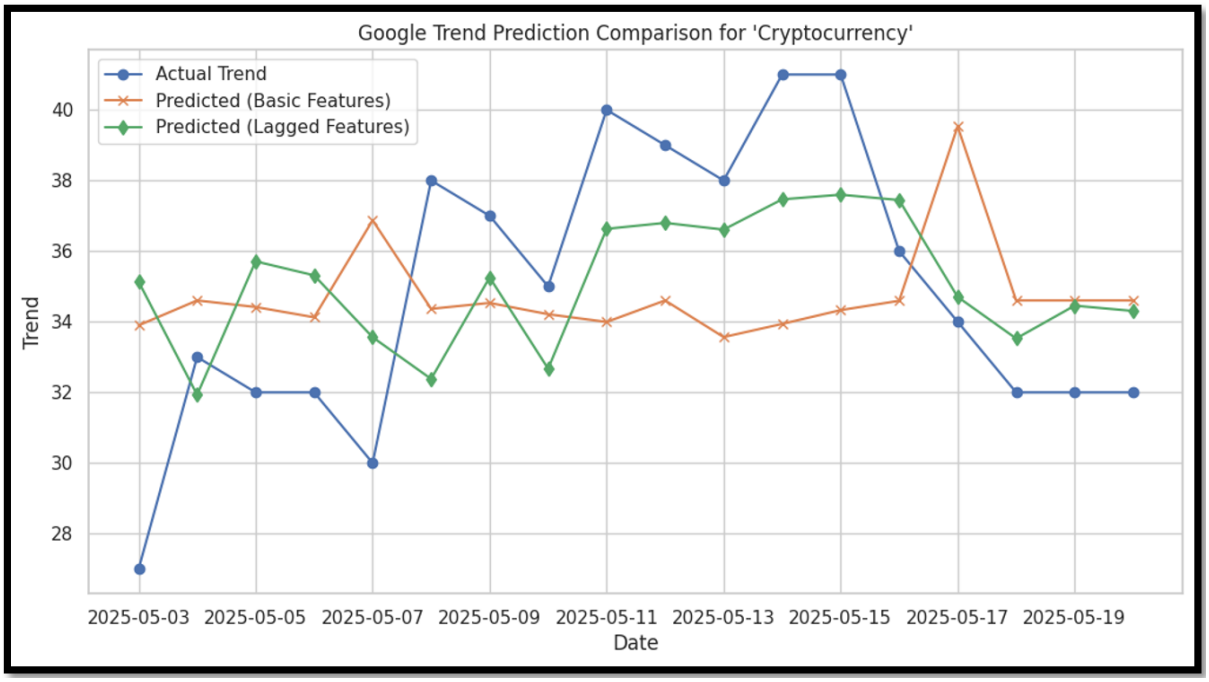


Figure 3.11: Google Trend Prediction Comparison for "Cryptocurrency"

Home Fitness

Home Fitness exhibited very limited forecast improvement using lagged models. This may reflect external, non-sentiment-driven factors influencing trend fluctuations.

Model Type	MAE	RMSE
Basic	10.68	12.19
Lagged	**10.60**	**11.75**

Table 3.10: Forecast Performance – Home Fitness

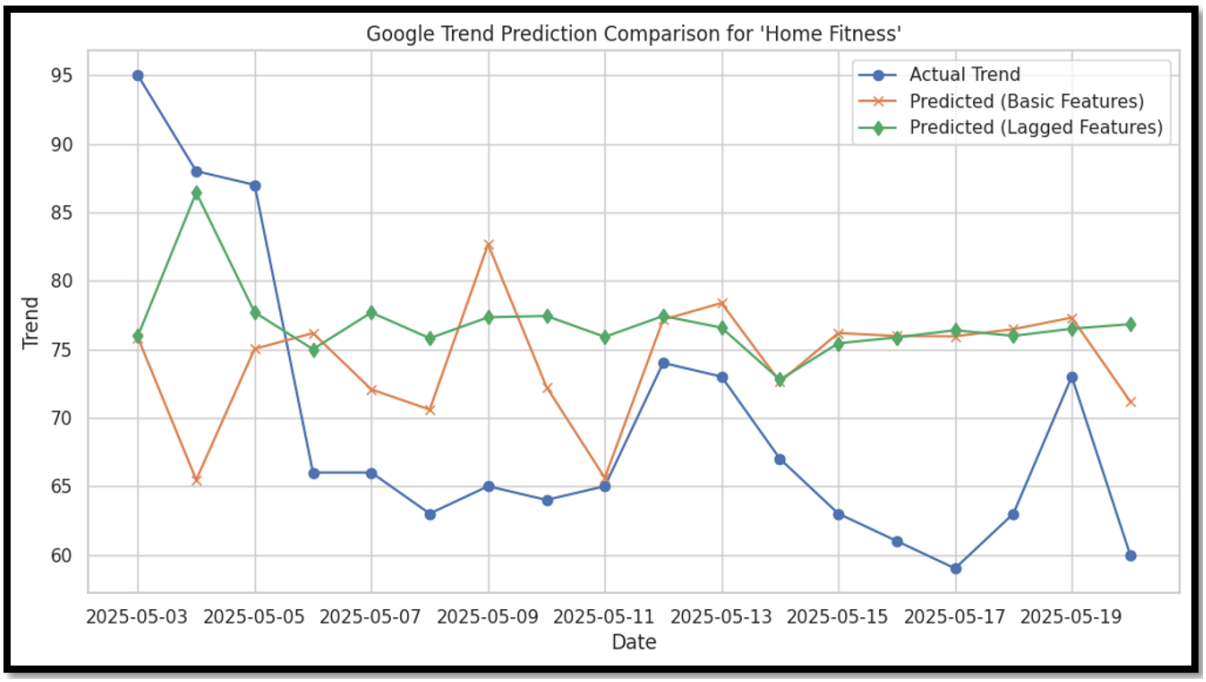


Figure 3.12: Google Trend Prediction Comparison for "Home Fitness"

In summary, the findings show that using lagged features significantly improves forecasting for fast-changing topics like Electric Cars and Cryptocurrency, reducing MAE by 39% and 26%. These topics rely on momentum and historical patterns. In contrast, for more stable topics like AI Startups, Vegan Food, and Home Fitness, complex models offer little benefit and can even overfit, as there's less trend-sentiment interaction. Sentiment by itself is a weak predictor—models that used only sentiment had high error rates, especially for Electric Cars. However, sentiment becomes more useful when combined with past trend data.

These results show that forecasting models work best when they're adapted to the specific topic. Adding sentiment data gives a small improvement, but combining it with time-based patterns makes predictions much stronger, especially for fast-moving trends.

3.6 Evaluation of Large Language Models

The final stage of our project includes a business advisor module powered by Large Language Models (LLMs), designed to generate actionable insights based on forecast outputs and sentiment signals.

3.6.1 Dataset Description and Preprocessing

The dataset used for training and evaluation is "finGPT-fiqa_qa", it focuses on financial question-answering tasks and provides a rich benchmark for assessing the reasoning and language understanding capabilities of large language models in finance-related contexts[30].

The dataset was divided into three standard subsets:

60% for training, 20% for validation, and 20% for testing. While only the training and validation sets were used to fine-tune one of the models (LLaMA), all models were evaluated in a zero-shot setting using a randomly selected sample of 100 examples from the test set. This allowed for a fair and consistent comparison of the generalization abilities of each model.

3.6.2 Evaluated Models

Four large language models were evaluated under the same conditions:

- **LLaMA (Fine-tuned):** This version of LLaMA was fine-tuned specifically on the "finGPT-fiqa_qa" dataset using the training and validation sets. The goal of this fine-tuning was to adapt the base LLaMA model to the financial QA domain, improving its relevance and response quality for domain-specific queries.
- **LLaMA (Base):** The original, unmodified version of LLaMA was used to serve as a baseline and measure the added value of fine-tuning.
- **Qwen3:** A powerful open LLM with multilingual and reasoning capabilities, tested here without any fine-tuning.
- **Falcon:** Another competitive open LLM used in its base form to benchmark performance against the fine-tuned and general-purpose models.

All models were evaluated in a zero-shot setting, meaning they were directly prompted with test questions without having seen those examples during training.

3.6.3 Evaluation Results and Analysis

The evaluation was conducted using four widely accepted metrics for natural language generation and QA performance:

- **ROUGE-L:** Measures overlap based on the longest common subsequence.
- **BERT Cosine Similarity:** Computes the semantic similarity between answers using BERT embeddings.
- **METEOR:** Focuses on recall, stemming, and synonym matches.
- **BLEU:** Captures n-gram precision, typically used in machine translation tasks.

The results are shown in the figure below:

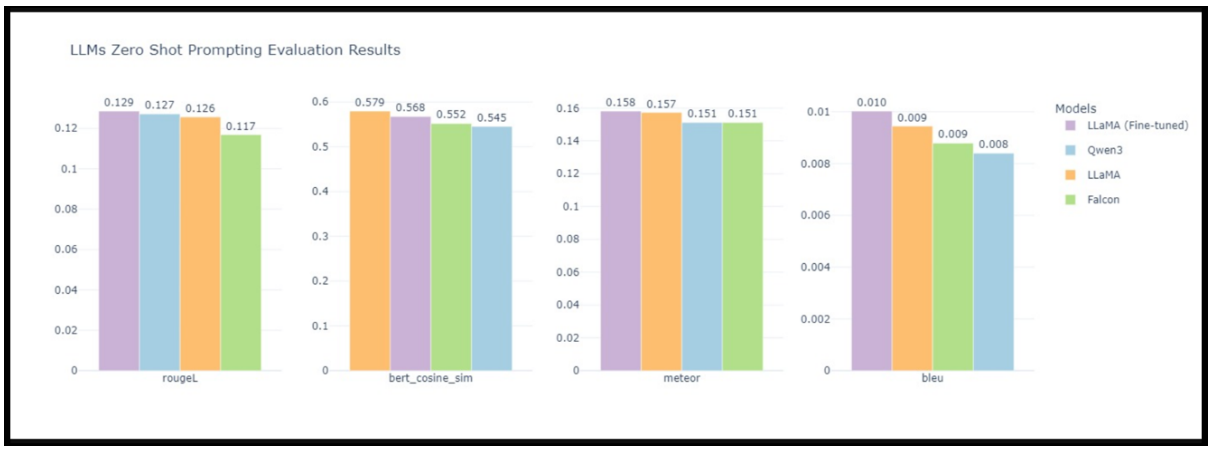


Figure 3.13: Evaluation of LLMs on the FinGPT Dataset

Key observations:

- **LLaMA (Fine-tuned)** consistently outperforms the other models across all metrics, demonstrating the impact of targeted fine-tuning on domain-specific tasks.
- **The base LLaMA** shows reasonable performance but trails behind its fine-tuned counterpart, especially on metrics related to semantic understanding (BERT cosine).
- **Qwen3** performs competitively, nearly matching LLaMA (fine-tuned) in some areas, but slightly lags overall.
- **Falcon** records the lowest scores on all four metrics, indicating limited adaptation to the financial QA context.

Based on these results, **LLaMA (fine-tuned)** was selected for further use. Its superior performance on three out of four evaluation metrics confirms the benefit of adapting general-purpose LLMs to domain-specific

datasets. This model offers a balanced combination of lexical accuracy and semantic understanding, making it the most suitable candidate for financial QA tasks in this study.

3.7 Conclusion

This chapter presented a comprehensive evaluation of forecasting models and large language models across multiple topics and tasks. For time series forecasting, results showed that model performance depends on the volatility and seasonality of the topic, with AutoML frameworks like AutoGluon offering robust, out-of-the-box performance. Sentiment features were helpful mainly when combined with lagged trend data. For language modeling, the fine-tuned LLaMA model outperformed others on financial question answering, confirming the benefits of domain-specific training. Together, these findings highlight the importance of model selection, feature engineering, and domain adaptation in building effective AI systems.

General Conclusion

This thesis explored the intersection of market trend prediction, sentiment analysis, and business strategy advisory through the lens of artificial intelligence. The research began with a comprehensive review of the relevant literature, identifying the theoretical underpinnings and technological advancements that have shaped the current landscape. Emphasis was placed on the growing role of AI particularly machine learning, deep learning, and large language models in transforming the way market behavior and public sentiment are analyzed.

The methodology chapter detailed the design and implementation of a hybrid framework integrating Google Trends data and Reddit sentiment analysis for forecasting market trends. Several forecasting models, including both classical and modern approaches (ARIMA, Prophet, Holt-Winters, LSTM, Random Forest, and AutoGluon), were implemented and compared using different combinations of features. Additionally, we incorporated the evaluation of large language models like LLaMA, Qwen3 and Falcon-7B to assess their potential as financial question-answering agents.

The results revealed that model performance varies significantly depending on the context, data fusion strategies, and the domain of the trend. AutoGluon consistently outperformed other models in accuracy, especially when sentiment data was integrated. Moreover, topic-specific analyses uncovered that certain domains such as cryptocurrency and AI startups exhibit more complex dynamics, benefiting substantially from the fusion of behavioral and sentiment indicators. Large language models also demonstrated promising capabilities, albeit with limitations in domain-specific reasoning and consistency.

Overall, this work contributes to the growing body of research advocating for multi-source, AI-driven approaches to trend prediction and strategic decision-making. By combining quantitative market signals with qualitative public sentiment, and integrating modern forecasting tools with conversational AI, we demonstrate the potential of intelligent systems in guiding businesses through complex, data-rich environments.

Future work may extend this framework with real-time forecasting, reinforcement learning for adaptive strategies, or fine-tuned language models tailored specifically to financial discourse. The integration of explainable AI techniques also remains a promising direction for increasing transparency and trust in AI-driven decision support systems.

Bibliography

[1] Sylvia Jenčová, Petra Vašaničová, Martina Košíková, and Marta Miškufová. A time series approach to forecasting financial indicators in the wholesale and retail trade. *World*, 6(1):5, 2025.

[2] S Radhakrishnan. Ai-driven business intelligence: Forecasting market trends and consumer behavior. *CONFLUENCE OF CURIOSITY*, page 73, 2023.

[3] B Uma Devi, Darshan Sundar, and P Alli. An effective time series analysis for stock trend prediction using arima model for nifty midcap-50. *International Journal of Data Mining & Knowledge Management Process*, 3(1):65, 2013.

[4] Yuling Lin, Haixiang Guo, and Jinglu Hu. An svm-based approach for stock market trend prediction. In *The 2013 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE, 2013.

[5] Kriti Pawar, Raj Srujan Jalem, and Vivek Tiwari. Stock market price prediction using lstm rnn. In *Emerging Trends in Expert Applications and Security: Proceedings of ICETEAS 2018*, pages 493–503. Springer, 2019.

[6] M KRISHNA Kishore, Subhani Shaik, and Bhavani S Panda. Exploration of the sentiment-driven forecasting models for predicting consumer purchase patterns on social media. *IJIRT*, 10(12), 2024.

[7] Bing Liu. *Sentiment analysis and opinion mining*. Springer Nature, 2022.

[8] The Business Research Company. Management consulting services global market report 2025, 2025.

[9] Enmin Zhu and Jerome Yen. Bertopic-driven stock market predictions: Unraveling sentiment insights, 2024.

[10] Mirza Selimovic, Ali Abd Almisreb, and Salue Amanzholova. The role of sentiment analysis in brand management and marketing: A comparative study. *Procedia Computer Science*, 251:579–584, 2024.

[11] Abdullah Önden, Mohammed Alnour, Vladimir Simic, and Dragan Pamucar. The evolution of sentiment analysis across various scien-

tific disciplines: A comprehensive review based on the bibliometric technique. *Decision Making Advances*, 2(1):222–237, 2024.

[12] Batta Mahesh et al. Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9(1):381–386, 2020.

[13] Julianna Delua. Supervised vs. unsupervised learning: What's the difference? *IBM Blog*, 2021.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[15] Amitha Mathew, P Amudha, and S Sivakumari. Deep learning techniques: an overview. *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pages 599–608, 2021.

[16] Shan Masood and Asad Abbas. Neural networks and deep learning: A comprehensive overview of modern techniques and applications, 01 2024.

[17] Weiche Hsieh, Ziqian Bi, Keyu Chen, Benji Peng, Sen Zhang, Jiawei Xu, Jinlang Wang, Caitlyn Heqi Yin, Yichao Zhang, Pohsun Feng, et al. Deep learning, machine learning, advancing big data analytics and management. *arXiv preprint arXiv:2412.02187*, 2024.

[18] Asifullah Khan, Anabia Sohail, Umme Zahoora, and Aqsa Saeed Qureshi. A survey of the recent architectures of deep convolutional neural networks. *Artificial intelligence review*, 53:5455–5516, 2020.

[19] Oona Rainio, Jarmo Teuho, and Riku Klén. Evaluation metrics and statistical tests for machine learning. *Scientific Reports*, 14(1):6086, 2024.

[20] Md Shakhaowat Hossain, Afrina Khan, Pritom Das, Md Sayem Ul Haque, Fnu Kamruzzaman, Sharmin Akter, Adib Ahmed, and Md Rashel Miah. Enhanced market trend forecasting using machine learning models: a study with external factor integration. *International Interdisciplinary Business Economics Advancement Journal*, 6(01):5–12, 2025.

[21] Shunrong Shen, Haomiao Jiang, Tongda Zhang, et al. Stock market forecasting using machine learning algorithms. *Department of Electrical Engineering, Stanford University, Stanford, CA*, pages 1–5, 2012.

[22] Tran Phuoc, Pham Thi Kim Anh, Phan Huy Tam, and Chien V Nguyen. Applying machine learning algorithms to predict the stock price trend in the stock market–the case of vietnam. *Humanities and Social Sciences Communications*, 11(1):1–18, 2024.

[23] Jakob Pallesen and Julius Varlid Bech. Combining machine learning and forecasting to predict real estate prices.

[24] Min-Hsuan Fan, Mu-Yen Chen, and En-Chih Liao. A deep learning approach for financial market prediction: Utilization of google trends and keywords. *Granular Computing*, 6(1):207–216, 2021.

[25] Krishnaveni Katta. Analyzing user perceptions of large language models (llms) on reddit: Sentiment and topic modeling of chatgpt and deepseek discussions. *arXiv preprint arXiv:2502.18513*, 2025.

[26] Sean J Taylor and Benjamin Letham. Forecasting at scale. *The American Statistician*, 72(1):37–45, 2018.

[27] Chris Chatfield. The holt-winters forecasting procedure. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 27(3):264–279, 1978.

[28] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[29] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[30] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

[31] Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

[32] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*, 2023.

[33] Tiago Carneiro, Raul Victor Medeiros Da Nóbrega, Thiago Nepomuceno, Gui-Bin Bian, Victor Hugo C De Albuquerque, and Pedro Pedrosa Reboucas Filho. Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *Ieee Access*, 6:61677–61685, 2018.

Webography

1. https://www.ibm.com/think/topics/encoder-decoder-model
(Accessed: April 15, 2025)

2. https://blog.roboflow.com/what-is-a-neural-network/
(Accessed: March 30, 2025)

3. https://www.aiacceleratorinstitute.com/evaluating-machine-learning-models-metrics-and-techniques/
(Accessed: April 5, 2025)

4. https://www.v7labs.com/blog/confusion-matrix-guide
(Accessed: April 5, 2025)

5. https://www.almabetter.com/bytes/tutorials/data-science/classification-metrics
(Accessed: April 5, 2025)

6. https://www.analyticsvidhya.com/blog/2022/03/a-brief-overview-of-recurrent-neural-networks-rnn/
(Accessed: March 30, 2025)

7. https://www.geeksforgeeks.org/introduction-convolution-neural-network/
(Accessed: March 30, 2025)

8. https://www.studysmarter.co.uk
(Accessed on March 27, 2025)

9. https://www.elastic.co/what-is/large-language-models
(Accessed: April 26, 2025)

10. https://fastercapital.com/topics/the-impact-of-market-analysis-on-business-strategy-and-planning.html
(Accessed: April 28, 2025)

11. https://www.rapidinnovation.io/post/ai-agent-market-trend-predictor
(Accessed: April 28, 2025)

12. https://www.rapidinnovation.io/post/ai-agent-market-trend-predictor#blog-hero
(Accessed: April 28, 2025)

13. https://www.datacamp.com/tutorial/arima
(Accessed: May 17, 2025)

14. https://datascientest.com/en/facebook-prophet-all-you-need-to-know
(Accessed: May 17, 2025)

15. https://nixtlaverse.nixtla.io/statsforecast/docs/models/holtwinters.html
(Accessed: May 17, 2025)

16. https://www.ibm.com/think/topics/random-forest
(Accessed: May 17, 2025)

17. https://tinyurl.com/4wpfe9bv
(Accessed: May 21, 2025)

18. https://auto.gluon.ai/1.1.0/tutorials/timeseries/forecasting-quick-start.html
(Accessed: May 24, 2025)

19. https://auto.gluon.ai/stable/tutorials/timeseries/forecasting-metrics.html
(Accessed: May 24, 2025)

20. https://www.deepchecks.com/glossary/llama/
(Accessed: May 22, 2025)

21. https://tinyurl.com/yrjjnyvm
(Accessed: March 15, 2025)

22. https://bility.fr/definition-visual-studio-code/
(Accessed: May 26, 2025)

23. https://pypi.org/project/pytrends/4.1.1/
(Accessed: May 28, 2025)

24. https://lazarinastoy.com/
the-ultimate-guide-to-pytrends-google-trends-api-with-python/
(Accessed: May 28, 2025)

25. https://asyncpraw.readthedocs.io/en/stable/
(Accessed: May 28, 2025)

26. https://matplotlib.org/3.7.1/
(Accessed: May 28, 2025)

27. https://en.wikipedia.org/wiki/Matplotlib
(Accessed: May 28, 2025)

28. https://www.ibm.com/think/topics/self-attention
(Accessed: May 30, 2025)

29. https://www.techtarget.com/whatis/definition/Hugging-Face
(Accessed: June 02, 2025)

30. https://huggingface.co/datasets/FinGPT/fingpt-fiqa_qa
(Accessed: June 08, 2025)

31. https://huggingface.co/Qwen
(Accessed: June 08, 2025)