UNIVERSITY OF TURKU
Turku School of Economics

Construction and Performance of an NLP–Neural Network Hybrid for Predicting Direction of the EUR/USD

Directional Signals from the Federal Open Market Committee's Statements and Minutes

Accounting and finance
Master's thesis

Author:
B.Sc. Petteri Mikkola

Supervisors:
Prof. Hannu Schadewitz
Prof. Luis Alvarez
M.Sc. Valtteri Peltonen

1.6.2026
Turku

The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the Turnitin Originality Check service.

Master's thesis

Subject: Accounting and Finance
Author: B.Sc. Petteri Mikkola
Title: Construction and Performance of an NLP and Neural Network Hybrid Model for Predicting the Direction of the EUR/USD
Supervisors: Prof. Hannu Schadewitz, Prof. Luis Alvarez, and M.Sc. Valtteri Peltonen
Number of pages: 114 pages + appendices 12 pages
Date: 1.6.2026

Development of transformer-based technologies such as generative Artificial Intelligence (AI) brought about by the seminal work of Vaswani et al. (2017). This presents opportunities for various markets to incorporate unstructured text data to make informed predictions and discover appropriate market prices at profound speed. Research on intraday EUR/USD predictions using machine learning (ML) based sentiment analysis of the Federal Open Market Committee's (FOMC) statements and minutes releases is relatively nascent and shows promise (Osowska & Wójcik 2024).

In light of these developments, this thesis incorporates FOMC-specific BERT models (Gössi et al. 2023; Shah et al. 2023) to predict direction of intraday 5-minute returns during FOMC release events. Both a *hawkishness* score and topic modelling-based topic-sentiment scores, inspired by previous research (Jegadeesh & Wu 2017; Aattouchi & Kerroum 2022; Shah et al. 2023), are implemented.

However, as stated by Osowska and Wójcik (2024, 166-167), the time horizon for predicting EUR/USD returns using BERT-based FOMC sentiment-scores appear brief, while reliable sentiment-score impacts are assessed to be marginal. To address these issues with prediction reliability, this thesis turned towards implementing hybrid ensemble models. This was motivated by ML research (i.e. Breiman 2001) and FX forecasting research on vote-tallying hybrid composite models for predicting the EUR/USD (Ling et al. 2021; Guyard & Deriaz 2024).

Four intraday artificial neural networks and two FOMC-specific sentiment models were employed as component models. Two vote-tallying hybrid models were tested and benchmarked on these component models to understand their ability to predict 5-minute direction of the EUR/USD. Results of this thesis suggest that rather than producing definite accuracy gains over their component models, hybrid models, especially with an equal weighting vote-tallying scheme, offer more consistent forecasting performance during market environments observed at the time of release of FOMC statement and minutes. Additionally, the overall sentiment and hybrid model performance observed at the time of release of FOMC statement and minutes might imply local market inefficiency as FOMC sentiment is slowly absorbed by the EUR/USD spot market.

AI disclaimer: Generative AI was used to assist with both the research for and writing of this thesis, as well as to review and generate draft project code. Scopus AI, Perplexity, and Gemini 3 were used to assist in finding suitable academic literature. While no sentence of this thesis was written by AI, Chat-GPT and Gemini 3 were used extensively to assist in clarifying statements and reviewing grammatical errors. Additionally, project code, particularly as it relates to the figures presented in this thesis, Chat-GPT was used to review code and generate preliminary snippets which were reviewed, tested, and edited by the author before implementation. DeepL was used to assist in translating this abstract from English to Finnish.

Key words: Foreign Exchange Market, Machine Learning, Natural Language Processing, Forecasting.

Gradututkielma

Oppiaine: Laskentatoimi ja rahoitus
Tekijä: KTK Petteri Mikkola
Otsikko: NLP- ja neuroverkkoja hyödyntävän hybridimallin rakentaminen ja suorituskyky EUR/USD-kurssin suunnan ennustamisessa.
Ohjaajat: Prof. Hannu Schadewitz, Prof. Luis Alvarez ja KTM Valtteri Peltonen
Sivumäärä: 114 sivua + liiteet 12 sivua
Päivämäärä: 1.6.2026

Vaswani et al. (2017) työstä alkanut Transformer-mallipohjaisten teknologioiden, kuten generatiivisen tekoälyn (AI), kehitys tarjoaa markkinoille mahdollisuuksia hyödyntää tekstidataan perustuvia ennusteita sekä löytää informaatiotehokkaita hintatasoja ennennäkemättömällä nopeudella. Tämän teknologiapohjaisen kehityksen pohjalta on syntynyt uusi rahoitustieteelle relevantti tutkimusalue, joka käsittelee päivänsisäistä EUR/USD-valuuttaparin ennustamista hyödyntämällä koneoppimispohjaista (ML) liittovaltion avomarkkinakomitean (FOMC) lausuntojen ja pöytäkirjojen sentimenttianalyysiä. Tämä tieteenalue on suhteellisen uusi, mutta lupaava (Osowska & Wójcik 2024).

Näiden kehitysten valossa tämä opinnäytetyö hyödyntää muun muassa FOMC:n sisältöjä varten suunnattuja BERT-malleja (Gössi ym. 2023; Shah et al. 2023), joilla ennustetaan EUR/USD-kurssin päivänsisäisten 5 minuutin tuottojen suuntia FOMC:n julkaisutapahtumien aikana. Tutkimuksessa sovelletaan *Hawkishness*- sekä aihemallintamiseen perustuvaa aihe-sentimentti-pisteytysjärjestelmää aiemman tutkimuksen inspiroimana (Jegadeesh & Wu 2017; Aattouchi & Kerroum 2022; Shah et al. 2023).

Osowska ja Wójcik (2024, 166–167) toteavat, että aikajänne EUR/USD-tuottojen ennustamiseen BERT-pohjaisilla FOMC-sentimenttipisteillä vaikuttaa lyhyeltä ja, että luotettavien sentimenttipisteiden vaikutusten arvioidaan olevan marginaalisia. Mahdollisten ennusteluotettavuuteen liittyvien ongelmien ratkaisemiseksi tässä työssä siirrytään käyttämään hybridejä ensemble-malleja, joita motivoi koneoppimistutkimus (esim. Breiman 2001), sekä EUR/USD-parin ennustamista koskeva valuuttamarkkinatutkimus, jossa on käytetty äänten laskuun perustuvia hybridejä, eli yhdistelmämalleja (Ling et al. 2021; Guyard & Deriaz 2024).

Neljä päivänsisäistä neuroverkkoennustemallia ja kaksi FOMC-julkaisuihin räätälöityä sentimenttiperusteista ennustemallia koulutettiin komponenttimalleiksi. Näiden avulla testattiin ja vertailtiin kahden äänestysperustusteisen hybridimallin kykyä ennustaa EUR/USD-valuuttakurssin viiden minuutin suuntaa. Tämän tutkielman tulokset viittaavat siihen, että nämä hybridimallit eivät niinkään paranna komponenttimalliensa ennustetarkkuutta, vaan erityisesti tasapainotetut hybridimallit tarjoavat luotettavamman ennustuskyvyn FOMC:n lausuntojen ja pöytäkirjojen julkaisemiseen liittyvissä markkinaympäristöissä. Lisäksi havaitut yleiset sentimentti- ja hybridiennustetulokset saattavat viitata hetkelliseen lokaaliin markkinatehottomuuteen FOMC-sentimentin vaikutusten suhteen.

Tekoälyseloste: Generatiivista tekoälyä käytettiin tämän opinnäytetyön tutkimuksen ja kirjoittamisen apuna sekä alustavan projektikoodin tarkistamiseen sekä sen luomiseen. Scopus AI:ta, Perplexityä ja Gemini 3 käytettiin tutkimuskirjallisuuden etsimiseen. Vaikka tekoäly ei ole kirjoittanut kokonaisia lauseita tähän opinnäytetyöhön, Chat-GPT:tä ja Gemini 3:a käytettiin erityisen paljon apuna lauseiden selkeyttämiseen ja kielioppivirheiden tarkistamiseen. Lisäksi projektikoodia, erityisesti tämän opinnäytetyön kuvioihin liittyvää koodia, tarkistettiin sekä alustavia koodinpätkiä luotiin Chat-GPT:n avulla. Tämän tutkielman kirjoittaja tarkisti, testasi ja korjasi koodinpätkät ennen niiden käyttöönottoa. DeepL-palvelua käytettiin apuna tämän tiivistelmän kääntämiseen englannista suomeksi.

Avainsanat: Valuuttamarkkinat, Koneoppiminen, Keinoäly, Luonnollisen kielen käsittely, Ennustaminen.

TABLE OF CONTENTS

1 Introduction 11

1.1 Ever-Better Tools for Decoding the Fed's Impact on the Most Liquid Market 11

1.2 Objectives and Structure 16

2 EUR/USD Currency Markets and the Federal Open Market Committee 18

2.1 Foreign Exchange Markets 18

2.1.1 An Overview of the World's Largest Financial Market 18

2.1.2 Dynamics of Spot Prices and Macroeconomic Fundamentals 21

2.1.3 Forecasting, Technical Indicators and Trading Spot Markets 25

2.2 The Federal Open Market Committee and Monetary Policy Announcements 29

2.2.1 United States Monetary Policy, The Fed, Statements, and Minutes 30

2.2.2 Effects of Statements and Minutes Releases on Financial Markets 34

3 Foundations of Classification, Regression, and Selected NLP Models 40

3.1 Classification, Regression, and The Selected ML Prediction Models 40

3.1.1 Machine Learning in a Nutshell: *Tasks, Learning Algorithms, and Loss* 40

3.1.2 Classification and Model Selection 47

3.1.3 Three Flavors of Artificial Neural Networks: *Vanilla, LSTM, and CNN* 53

3.1.4 Brief Literature Review on FX Predicting CNN and LSTM Models 59

3.2 Natural Language Processing: *From Text to Topics and Sentiment* 63

3.2.1 Understanding Structured Topic Models 63

3.2.2 Meet Bidirectional Encoder Representations from Transformer aka BERT 67

4 Data and Implementation 73

4.1 Assembling the Structured Dataset 74

4.1.1 Macroeconomic and Technical Indicators 74

4.1.2 Structured Inputs: *Labels, Feature Selection, Scaling, and Missing Values* 79

4.2 Engineering NLP Sentiment Features from the Unstructured Dataset 81

4.2.1 The Unstructured Dataset: *FOMC Statements and Minutes* 82

4.2.2 Encoding Sentiment from Fed Speak with STM and BERT 85

4.3 Training and Evaluating FX Prediction Models for the EUR/USD 90

4.3.1 General Intraday Regression ANNs 90

4.3.2 FOMC Release Specialized Sentiment and Hybrid Classification Models 95

4.3.3 Binoculars to a Horserace: *Selected Metrics and Statistical Tests* 97

5 Results **101**

5.1 Generating Sentiment Features: *Encoding Sentiments of the FOMC* **101**

5.1.1 Discovering Latent Topics of the FOMC Statements 101

5.1.2 Discovering Latent Topics of the FOMC Minutes 104

5.1.3 Encoding Sentiments of the FOMC with Sentiment Analyses 107

5.2 Constructing the Hybrid Model Components **111**

5.2.1 Structured Data Predictors: *Performance and Training of Intraday Regressors* 111

5.2.2 Sentiment Feature Predictors: *Parametrization of Release Day Classifiers* 115

5.3 Evaluating Performance During FOMC Statement and Minutes Releases **118**

6 Conclusions **123**

References **125**

Appendices **140**

Appendix 1: Brief Note on Notation and Data **140**

Appendix 2: Illustration of Backpropagation **142**

Appendix 3: Sequential Nature of Global Currency Markets **144**

Appendix 4: Pre-Processing FOMC Documents for STM **145**

Appendix 5: Announcement on the Use of Generative AI **149**

LIST OF FIGURES

Figure 1: Trends in FOMC Statements and Minutes Document Size 33
Figure 2: Splitting the Data 42
Figure 3: Bias-Variance Trade-Off 46
Figure 4: Confusion Matrix Illustration 49
Figure 5: ROC Curve Illustration 50
Figure 6: The Random Forest Algorithm 51
Figure 7: Structure of a LSTM Hidden Layer 56
Figure 8: One-Dimensional Convolution Layer 58
Figure 9: Pooling Layers of 1D-CNN 59
Figure 10: Graphical Illustration of the STM 67
Figure 11: Simplified BERT-like Classification Encoder 69
Figure 12: Overview of Models and Data Implementation 73
Figure 13: Distributions of Cosine Similarity, Minutes & Statements 84
Figure 14: Word clouds by Document Type 85
Figure 15: Pre-Processing of FOMC Documents for STMs 86
Figure 16: Topic Model Cross-Validation for Number of Statement Topics 102
Figure 17: STM Topics from FOMC Statements 103
Figure 18: Topic Word Clouds from FOMC Statements STM 104
Figure 19: Preliminary CTM Results for Number of Minutes Topics 105
Figure 20: STM Topics from FOMC Minutes 105
Figure 21: Topic Word Clouds from FOMC Minutes STM 106
Figure 22: Statement Document-level Topic-Sentiment Scores 107
Figure 23: Minutes Document-level Topic-Sentiment Scores 108
Figure 24: Hawkishness Scores 109
Figure 25: Regression Model Heteroskedasticity and Training Budget 113
Figure 26: Regression Performance Metrics in Training 113
Figure 27: Regression Performance Metrics in Training 115
Figure 28: Sentiment Model Hyperparameter Tuning 117
Figure 29: Giacomini-White 2006 and Mid-*p* test p-values 121
Figure 30: A 3rd Order Tensor Visualized 140

LIST OF TABLES

Table 1: Effects of FOMC Releases on Financial Markets from Literature Review 34
Table 2: Sample of Currency Prediction CNNs and LSTMs from Literature 60
Table 3: Volatility Features in Structured Data 77
Table 4: Moving Average Features in Structured Data 78
Table 5: Descriptive Statistics of Test Data Label Variables for ANNs 80
Table 6: FOMC Documents Descriptive Statistics 82
Table 7: Regression ANN Configurations and Crucial Parameters 94
Table 8: Hyperparameters and Data for Sentiment-Based Classifier Models 95
Table 9: McNemar Test Contingency Table 99
Table 10: Aggregate Test Window Performance of ANNs 111
Table 11: Regression Analyses: Predictive Power of ANNs 112
Table 12: Performance of the MM2023 ANN Models in Test Windows 112
Table 13: Performance of the Macro and Tech ANN Models in Test Windows 114
Table 14: Regression Coefficient Analyses on Sentiment Features 115
Table 15: Sentiment Model Performance by AUC 116
Table 16: Accuracy of EUR/USD Direction Predictions 118
Table 17: Precision of EUR/USD Direction Predictions 120
Table 18: Summary of Giacomini-White 2006 T-tests 120
Table 19: Trading hours of Global FX Market Sessions 144

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
ARCH	Autoregressive Conditional Heteroskedasticity
AUC	Area Under the Curve
BB	Bolinger Bands
BERT	Bidirectional Encoder Representations from Transformers
BIS	Bank of International Settlements
bps	Basis Point, 1bps = 0.01%
BPTT	Backpropagation Through Time
CART	Classification and Regression Trees
CET	Central European Time
CIP	Covered Interest Rate Parity
CNN	Convolutional Neural Network
CTM	Correlated Topic Model
CV	Cross-Validation
DA	Directional Accuracy
DFM	Document Feature Matrix
DST	Daylight Saving Time
ECB	European Central Bank
EMH	Efficient Market Hypothesis
ET	Eastern Standard Time
EMA	Exponential Moving Average
EUR	Euro
EUR/USD	The United States dollar to euro exchange rate; the market value of euros in dollars
EW	Equally Weighted Hybrid Model
Fed	The Federal Reserve System of the United States
FOMC	Federal Open Market Committee
FX	Foreign Exchange
GW	Giacomini-White 2006 Test
HAC	Heteroscedasticity and Autocorrelation Consistent
IFE	International Fisher Effect
IMF	International Monetary Fund
LDA	Latent Dirichlet Allocation

LLM	Large Language Model
LR	Logistic Regression
MACD	Moving Average Convergence Divergence
MHA	Multi-Head Attention
ML	Machine Learning
NLP	Natural Language Processing
OLS	Ordinary Least Squares
PERF	Performance Evaluating Hybrid Model
PPP	Purchase Power Parity
RF	RandomForest
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
RV	Realized Volatility
SMA	Simple Moving Average
SR	Support and Resistance Levels
STM	Structured Topic Model
TSRV	Two Scaled Realized Volatility
UIP	Uncovered Interest Rate Parity
USD	United States Dollar
U.S.	The United States of America
WTI	West Texas Intermediate

1 Introduction

1.1 Ever-Better Tools for Decoding the Fed's Impact on the Most Liquid Market

Recent advancements in Artificial Intelligence (AI) have introduced substantial improvements in our ability to process text data (e.g. unstructured data) at ever faster speeds. In the field of finance, institutions such as Bloomberg (Wu et al. 2023) have created domain specific models to improve efficiency of processing such data. Implementing Artificial (AI) by modeling both key U.S. monetary policy pronouncements with Natural Language Processing (NLP) technologies and training conventional Artificial Neural Networks (ANNs) on structured intraday data, this thesis explores capabilities of hybrid models, to predict intraday directional signals for the EUR/USD exchange rate.

Of the major currency pairs, the EUR/USD has the largest and most liquid foreign exchange (FX) market, boasting an average daily volume of 1,165.2 billion U.S. dollars across various over-the-counter instruments in April 2024 (New York Federal Reserve, 2024). Until recently, researchers have largely focused on structured data in assessing impacts of FOMC announcements on currency pairs: considering factors such as inflation, interest rates, document release date, quantitative easing, and implied volatility (Mueller 2017; Tse 2019; Fassas et al. 2021; Dedola et al. 2021). But such structured data is only part of the story for FX markets. Importantly, high liquidity USD currency pairs and their derivative instruments appear sensitive to both statements and minutes of the FOMC (Rosa, 2013; Tse 2019; Tadle 2022; Osowska & Wójcik 2024).

As the largest economy in the world, monetary policy of the United States' (U.S.) is of great interest to global financial markets. The Federal Reserve System (Fed) is the central bank of the U.S., operating under the dual mandate of price stability and maximum employment. The institution of the Fed responsible for setting monetary policy is the Federal Open Market Committee (FOMC). The committee convenes eight times a year, regularly, to review economic and financial conditions and to set the federal stance on monetary policy. Occasionally, additional meetings are held, for example in face of surprising economic or financial environments. Recently such meetings were held in both 2019 and 2020, due to the Covid-19 pandemic. (Federal Reserve 2025a.)

Pronouncements of the FOMC have been shown to be a major driver of financial market returns, volatility and trading volumes (see e.g. Rosa 2013; Jubinski & Tomljanovich 2017; Tadle 2022; Shah et al 2023; IMF 2023; Osowska & Wójcik 2024). These pronouncements include documents such as, policy statements, the Chair's quarterly press conference, meeting minutes, a summary of Economic Projections, Governors' speeches, congressional testimony by the Chairperson, speeches by bank

presidents, a semi-annual written report to Congress, and news articles produced by the Fed (Osowska & Wójcik 2024). Of these FOMC documents, policy statements and meeting minutes occur regularly and during trading hours of most U.S. financial markets (Jubinski & Tomljanovich 2017). These two documents will be referred to as statements and minutes henceforth.

Arguably statements are the most influential documents produced by the FOMC as reflected by substantial market impacts produced within their wake (Rosa 2013, 72). Statements are typically brief and the very first piece of official information expressing the FOMC's current views on the U.S. economy and the inflation environment, their decisions on target interest rate and employment of policy tools to achieve monetary policy goals, forward guidance, and a summary of votes by its members (St. Louis Federal Reserve 2019). These statements provide, not just information on monetary policy, but also expert views on the health and course of the U.S. and global economies.

In contrast, minutes provide a more comprehensive and detailed account on perspectives of committee members regarding suitable central bank posture, macroeconomic developments, and the recent central bank action and forward guidance. Though minutes are released after statements, this more granular release manages to move markets. (Rosa 2013, 67-68.) Both documents and sentiments contained within are closely surveyed and dissected by financial and economic stakeholders.

Particularly, FX markets have exhibited significant re-pricing, or even price shocks, spanning multiple minutes before settling, preceding the release of these documents. (Rosa 2013, 69-79; Kansoy 2022, 37; Kim et al. 2024, 626-627.) These intervals of market adjustment suggest that there exists an opportunity for arbitrage by exploiting temporally local market inefficiencies. Such opportunities might present themselves to participants who gain superior knowledge by deciphering relevant information and sentiments in a timely manner from these FOMC pronouncements.

The emergence of new Machine Learning (ML) based Natural Language Processing (NLP) technologies, and particularly nascent developments in transformer architecture, have given rise to accessible *context aware* sentiment analysis tools. Unlike previously implemented dictionary- and rule-based methods, transformer-based models account for contextual cues and connotations (Devlin et al. 2019; Araci 2019; Shah et al. 2023). To illustrate, the words "decrease" and "unemployment" in isolation might appear to have negative connotations, unless viewed in the following context "we observe a significant decrease in unemployment."

Accounting for such nuance is crucial in interpreting the concise and context bound FOMC pronouncements (Shah et al. 2023). Recently researchers have trained NLP models and published

research concerning FOMC documents, providing their Bidirectional Encoder Representations from Transformers (BERT) models and expertly labeled training datasets as open source (Shah et al. 2023; Gössi et al. 2023; Kim et al. 2024). Such efforts are laying down groundwork for more accurate and timely interpretations of central bank communication by global financial markets, increasingly so as these sophisticated NLP approaches *democratized* and publicly shared.

Regarding sentiment analysis, transformer models show promise in their ability to classify sentiment when faced with FOMC language. Shah et al. (2023) benchmarked NLP methods, including BERT models, by classifying sentences from FOMC pronouncements as dovish, neutral, or hawkish. Where dovish sentiment signals a loose stance on monetary policy, or that the Fed might attempt to facilitate lending activity and lower interest rates. Neutral sentiment signals the Fed might seek to maintain current conditions. Hawkish sentiment signals the Fed's willingness to temper lending and raise rates. Moreover, Gössi et al. (2023) and Kim et al. (2024) benchmarked models in classifying sentences from FOMC pronouncements as having a positive, neutral, or negative tone regarding financial-economic sentiment. Overall, Gössi et al. (2023), Shah et al. (2023) and Kim et al. (2024) all highlight how transformer models excel over traditional NLP approaches. Implementation of FOMC-specific BERT models for time series prediction is a novel avenue of research, as these models have only quite recently been published. As a first step, Shah et al. (2023, 6669-6672) showed that their model's sentiment-scores could be used to predict treasury rates and profitably trade equity markets.

Additionally, another form of NLP which has reared its head in financial literature is topic modeling (Jegadeesh & Wu 2017; Cerchiello & Nicola 2018; Ferrara et al. 2022; Aattouchi & Kerroum 2022; Koelbl et al. 2024). Topic models, particularly the Structured Topic Model or STM, aims to model co-occurrence of words and their groupings within a collection of texts to create topics similar to those recognized by human experts (Roberts 2016). A topic model infers prevalent collections of co-occurring words or topics within the corpus by training and learning a model of the texts. In a sense, topic modelling can be thought of as dimensionality reduction, like a type of principal component analysis or factor analysis, that allows the analysis of topics instead of individual words, akin to latent factors instead of variable values. This suggests that human-like topic recognition may be automated and importantly accelerated with the use of STMs.

STMs have been successfully implemented in analyzing financial news and announcements, condensing long unstructured data from the word level to the topic level, allowing for efficient analysis of documents. For example, Cerchiello & Nicola (2018) explored the diffusion of banking related news topics nation to nation using a STM in conjunction with granger causality time series

modeling. They extracted common news topics from half a million news articles from Reuters and Bloomberg between October 2006 and November 2013. The prevalence of topics was organized by country and then granger causality for the *diffusion* of topic prevalence nation to nation was estimated. The researchers managed to extract and track the global diffusion of news trends in media.

Meanwhile Koelbl et al. (2024) employed a structural topic model to evaluate prevalence of risk factors in topics within corporate disclosures and their corresponding effects on market pricing behavior. The disclosures published with risk-loaded STM topics resulted in lower return volatility. Pertinently, Ferrara et al. (2022) analyzed monetary policy discussions between European Parliament members and the president of the European Central Bank (ECB), finding supporting evidence for a "Political Phillips Curve" -like dynamic. They found that politicians' speech contained less inflation related topics during high unemployment, suggesting a trade-off where stimulus is prioritized over inflation concerns, potentially facilitating a Phillips Curve type relationship between the two variables. Topic modeling approaches on FOMC texts have been successfully illustrated in works by Jegadeesh and Wu (2017), and Aattouchi and Kerroum (2022, 125-129). Also, by implementing a STM on news titles from the Wallstreet Journal, Cormun and Ristolainen (2024, 26) managed to improve the performance of macroeconomic indicator models with topic-based interaction terms.

Thus, an opportunity to not only ascertain sentiment with *context-aware* FOMC-specific BERT-models, but also to associate that sentiment with relevant topics, such as inflation or economic growth, might be possible. Thus, this thesis seeks to employ both FOMC-specific BERT models and trains document specific STMs to implement sentiment scores similar to Jegadeesh and Wu (2017, 16). These features are engineered or created via a sentiment scoring process and are then used to predict direction of the EUR/USD out-of-sample.

Research on time series prediction for currencies using unstructured data appears novel. Ito and Takeda (2022) and Ito et al. (2021) construct autoregressive models utilizing sentiment indexes from Google Trends data. The researchers found sentiment index factors to improve their models' predictive ability, especially direction of change in exchange rates. Employing a joint sentiment-topic model Yono et al. (2019) found predictive models of movements could be significantly improved in an intraday-dataset. Pertinently, Osowska and Wójcik (2024, 166-169) found that employing sentiment scores produced by a general Fin-BERT model on FOMC statements could lead to profitable trading strategies for the EUR/USD. Though the prediction power for sentiment models was not reliable. Also, work by Ding et al. (2024) modeled the EUR/USD currency pair continuously, utilizing both structured and unstructured data, with the latter consisting of news and market analyses.

Their report concludes that the model could improve predictive accuracy above that of structured data alone within their sample. Thus, the literature appears to support the use of topic models, BERT sentiment, and combined use of both unstructured and structured data for successful and potentially profitable EUR/USD prediction.

The Efficient Market Hypothesis (EMH) posed by Fama (1970, 415-416) suggests that markets should not provide opportunities for profitable predictability on historical data alone. Yet, research shows that even the world's most liquid financial market might fall prey to temporary predictability. This might be particularly true during market dynamics brought about by FOMC releases and sentiment analysis. Particularly, this thesis incorporates and evaluates a subset of such innovative technologies to explore a novel approach to predicting direction of the EUR/USD FX rate during market environments influenced by FOMC text.

The sentiment baked into these texts have become an ever more important policy tool. To highlight this: largest number of dissenting votes since 1992 in the FOMC statement were observed on the 29th of April 2026, where only one dissenting vote was cast over the policy rate decision, while three dissenting votes emerged as members did not support easing bias in the statement. (Federal Reserve 2026). Such sentiment might be key to predicting policy decisions affecting interest rates and inflation of the United States and thus major USD currency pairs.

Empirically, USD currency-pair markets are sensitive with respect to FOMC pronouncements (Rosa 2013; Mueller et al. 2017; Tse et al. 2019; IMF 2023; Osowska & Wójcik 2024). Thus, new opportunities to employ their unstructured data are presenting themselves with NLP technology allowing for better understanding underlying the information structure of FX markets. Two such NLP technologies are implemented in this thesis: STMs and BERT.

However, keeping in mind research by Osowska and Wójcik (2024) not only their potential but also, volatile error rates underlying use such of sentiment analysis for FOMC release-time predictions have been evidenced. Thus, this study employs NLP methods in conjunction with conventional structured data predictors to smooth out volatile predictions. In doing so this thesis explores implementation of hybrid ML models to detect intraday direction of the EUR/USD and asses what advantages such models might have in reducing error over the base predictors alone. This contributes new perspectives and evidence to both the extant body of research on FX prediction and NLP in finance.

Ultimately, such perspectives wish to inform market actors towards better algorithmic price discovery during market conditions produced by the release of FOMC documents. Also, because good

algorithmic trading might improve price discovery and lower volatility (Hendershott et al. 2011, 4, 30; Manahov et al. 2014, 134-135, 153), these perspectives hope to serve actors in the FX markets to foster greater market efficiency and contribute ideas for better text data informed prediction models. Accurate price discovery, particularly in the world's largest FX market, should ideally lend itself to efficiency of international capital allocation (Fama 1970, 383).

1.2 Objectives and Structure

This paper explores the capabilities of a hybrid machine learning model to detect depreciation and appreciation signals of the EUR/USD by utilizing unstructured text data from FOMC minutes and statements, and structured time series data. The hybrid models are constructed as binary classifiers based on component models which employ intraday financial time series data, and NLP generated FOMC sentiment indicators. The objectives of this thesis are threefold: (1) constructing intraday ANN EUR/USD predictors on a structured dataset motivated by FX literature, (2) building FOMC sentiment models motivated by NLP literature, and (3) exploring the performance of hybrid models utilizing both the intraday ANN and FOMC sentiment models.

To encapsulate these objectives, this thesis aims to answer the research question:

> To what extent do hybrid models that combine intraday ANN predictors and FOMC sentiment predictors enhance EUR/USD forecasting performance during FOMC statements and minutes releases compared to singular models?

If hybrid models or possibly sentiment models outperform aggregate ANN predictions, one might ascertain that sentiment modeling does indeed provide some advantage for predicting the EUR/USD. Indeed, hybrid models' ability to smooth-out idiosyncratic variability (Ling et al. 2021, 9, 18) and the volatile but potentially potent sentiment-based prediction (Osowska and Wójcik 2024, 167-168), should provide complimentary capabilities in enhancing the EUR/USD direction forecasts.

Furthermore, if accuracy is high enough, such enhancement could potentially signal local market inefficiency during FOMC statements and minutes release. Such outcomes could suggest arbitrage opportunities for those using NLP methods. For, example, arbitragers implementing such sentiment models could have potentially predicted the direction where other market participants, who might be slower to assess implications of the FOMC's text, were likely to price the EUR/USD currency pair.

Model predictions will be evaluated primarily using the accuracy metric, following the convention set in most FX prediction literature (Galeshchuk & Mukherjee 2017; Yıldırım et al. 2021;

Luangluewut & Thiennviboon 2023; Guyard & Deriaz 2024; El Zaar et al. 2025). To answer the research question more effectively, precision metrics and Giacomini-White (2006) and mid-*p* tests were employed. The empirical analysis used 1-minute close log-returns, as it is assumed that the time from implementing data to creating predictions could take a minute or less to compute inference on, especially when accounting for NLP methods. Mid-price-based log-returns were selected for their use in high frequency settings within FX literature (i.e. Andersen et al. 2003; Rosa 2013, 40; Ben Omrane et al. 2019, 5; Ben Omrane et al. 2020, 89).

Decision to select a 5-minute interval as the prediction target, and in fact the EUR/USD market, was based on Kansoy's manuscript (2022, 37), which suggests evidence of high volatility in EUR/USD returns occurring within 10 minutes after the FOMC's release of statement or minutes. The 5-minute interval is also supported by findings of Rosa (2013, 69-78), showing that volatility of USD FX rates increased rapidly for a short period of time, especially for the EUR/USD during statements and minutes releases with high and significant volatility occurring within 5-minutes.

The decision on selecting directions of returns, which are binary outcomes, meant that less information concerning future returns needed to be estimated by the models. Fundamentally, the magnitude of returns was left out which considerably simplified the forecast task (Hämäläinen 2015, 13). Also, regarding future research on trading FX using such predictors, economic rationale for engaging in the market for 5-minute intervals can be rationalized as long as transaction costs are overcome in aggregate, and because long and short positions are also binary.

Motivation for using hybrid networks stemmed from the brief and somewhat unreliable impacts of sentiments scores reported by Osowska and Wójcik (2024, 166-167). The vote-tallying design for hybrid models was inspired by ensemble models from ML literature (Breiman 2001) and previous research on similar models for EUR/USD prediction (Ling et al. 2021; Guyard & Deriaz 2024).

The rest of this thesis is structure as follows. Chapter 2 reports on a literature review concerning foreign exchange markets, EUR/USD exchange rates, FOMC pronouncements, and FX prediction. Chapter 3 introduces fundamentals of ML and the models implemented. This chapter is parsed into general structured data ML and NLP. Chapter 4 covers data and implementation of models for this study by describing the data, and implementation and evaluation of models. Chapter 5 presents and discusses topic model evaluation, BERT based sentiment variables, intraday performance of ANNs, construction of sentiment models, and results of comparative performance of the models. Finally, Chapter 6 concludes this thesis and discusses limitations and avenues for future research.

2 EUR/USD Currency Markets and the Federal Open Market Committee

2.1 Foreign Exchange Markets

This thesis concerns forecasting the EUR/USD currency-pair spot market. This section is split into three parts: 1) introducing foreign exchange markets; 2) the dynamics of spot prices and economic fundamentals; 3) forecasting and trading literature. All subsections begin by laying out the background of the subject matter and then presenting relevant findings from literature.

2.1.1 An Overview of the World's Largest Financial Market

The global currency market, also known as the foreign exchange (FX) market, is the world's largest financial market by trading volume, far outsizing global equity markets (Contreras et al. 2020, 2022-2023; Chaboud et al. 2023, 253). The major FX markets recorded trade volumes of 9.6 trillion USD per day in April 2025 according to the latest triennial survey conducted by the Bank of International Settlements (2025, 5), henceforth BIS. The primary purpose of FX market is to allow transfers of purchasing power denominated in one currency to another –this is achieved by exchanging one currency for another (Shapiro & Hanouna 2020, 168). FX markets determine the relative values of global currencies, facilitating international transactions in goods, services, and financial assets (Chaboud et al. 2023, 253), and therefore play a vital role in today's global economy.

The FX currency can be delivered via multiple different types of instruments, the most common are swaps, spot, and forward agreements (BIS 2025, 6). In the spot market currencies are traded for immediate delivery, but the transfer of funds occurs 2 trading days after at a settlement date (Shapiro & Hanouna 2020, 168; Chaboud et al. 2023, 257). The spot market accounted for about a third of FX transactions (BIS 2025, 6). On the FX market, currencies are represented by their ISO 4217 three-letter code. For example, the euro is denoted EUR, and United States dollar is USD. Also, exchange rates are quoted by currency pairs, as a price of the one country's currency in another reference country's currency, with the reference currency listed first (Shapiro & Hanouna 2020, 38). For example, the EUR/USD exchange rate is quoted as dollars per euro (Chaboud et al. 2023, 254).

More than 50 currencies are regularly traded, of which the USD is clearly most dominant, with the EUR trailing at a distinct second place (Chaboud et al. 2023, 253). The world reserve currency, the USD, played part in 89% of all FX transactions in April 2025 and 2022 triannual surveys, whilst EUR accounted for 29% of transactions in 2025 (BIS 2025, 3,8). Understandably, their currency pair, the

EUR/USD, is the most liquid, accounting for 21% of global FX trade volume in April 2025 (BIS 2025, 8; Chaboud et al. 2023, 254-255). As the major currency pair EUR/USD, has slowly declined in status while emerging pairs such as CNYUSD (Chinese Yuan) and BRLUSD (Brazilian real) have increased their share of FX volumes (BIS 2025, 8, 15), reflecting economic growth in the global south.

Central banks play an important role in FX markets as the official national monetary authorities (Ferrara et al. 2022 73-74; Shapiro & Hanouna 2020, 44). For example, the United States' central bank: the Federal Reserve System (Fed) operates under two goals, price stability and maximum sustainable employment, colloquially the "dual mandate" (Chicago Federal Reserve 2020). The ECB's primary objective is to maintain price stability (ECB 2021). Central banks influence FX markets directly with intervention measures or indirectly, particularly when influencing domestic money supply. Monetary policy issues such as inflation and interest rates are crucially addressed by central banks, and these decisions affect exchange rates (Shapiro & Hanouna 2020, 39-55). Today most countries let their currencies float, meaning that the FX rate is chiefly determined by market dynamics. Some countries that peg their currency against other commodities (nowadays other currencies, and previously gold, i.e. Bretton Woods), have adopted currency boards (IMF 2024, 14-15; Shapiro & Hanouna 2020, 44-47). Under today's currency board systems, central banks have no discretion over monetary policy action by themselves, and currency boards issue convertible notes and coins at a fixed rate in the select foreign reserve currency (Shapiro & Hanouna 2020, 47).

Structure of the FX markets could be understood as a decentralized network of interconnected banks, FX dealers and brokers, which bring together buyers and sellers. The FX market operates continuously 24 hours a day, except on weekends, with opening exchanges assuming responsibility as others close for the day (Chaboud et al. 2023, 253; Contreras et al. 2020, 2022-2023). Thus, major FX markets are found around the globe, with most of the trading (81.6%) occurring in Japan, Hong Kong, Singapore, Switzerland, the United Kingdom, and the United States. Notably, the United Kingdom accounted for 37.8% of FX trading volume in April of 2025 and together with the United States 56.4% (BIS 2025, 10,15). Increasingly, most FX trading is electronic, for example, in the United States 60.5% of spot transactions were executed electronically (New York Federal Reserve, 2024) with the comparable figure for the United Kingdom standing at 69.8% (Bank of England 2025).

Most transactions on the FX markets are carried out wholesale in interbank markets by brokers, or between financial institutions and FX dealers (BIS 2025, 5-7; Chaboud et al. 2023, 257). Brokers can help ensure anonymity and sufficient volumes for FX transactions, and grant access to plethora of

markets for non-bank actors. Most FX trade is facilitated over the SWIFT (Society for Worldwide Interbank Financial Telecommunication) system. SWIFT (2005) is a member-owned cooperative, linking together more than 11,000 financial intuitions from over 200 countries, allowing for efficient communication and automated data processing between banks. (Shapiro & Hanouna 2020, 168-175.)

Additionally, the FX market is largely fragmented with an innumerable number of transacting actors, of which the top five transacting entities make up 44.14% of trade volume (Euromoney FX Survey 2022, 1-5). This consolidation might in part be due to large banks being better informed. With a broad base of customer order flows and access to asset managers, banks can plan and implement efficient FX allocation (Bjønnes et al. 2021, 3125). Thus, whilst decentralized in structure, trade volumes portray a high degree of concentration by market and transactor. Market transactions have generally been categorized into two segments by participants: 1) interdealer and 2) dealer to customer (BIS 2022, 3,6; Shapiro & Hanouna 2020, 168-169). This distinction has gradually blurred as new ways of transacting and conducting FX deals emerge. Hence, BIS categorizes FX deals into 1) interdealer, 2) dealer to other financial institutions, and 3) dealer to non-financial customers (BIS 2022, 3, 6).

Notably, from 2000 to 2022 the proportion of deals categorized as dealer to other financial institutions grew from roughly 20% to 53% of trading volume, whilst the proportion of transactions with non-financial declined from 20% to less than 10% (BIS 2022, 3, 6). This group of other financial institutions consists of institutional investors, asset managers, hedge funds, commodity trading advisors, and smaller banks which are not FX dealers themselves (BIS 2022, 3, 6, 18). These developments signal a substantial rise in speculative and hedging FX trades, coinciding with the expansion of sophisticated high-frequency trading (HFT), especially as principal trading firms now account for roughly a third of all electronic FX transactions (Chaboud et al. 2023, 253-257, 264-266).

The rapid adoption of electronic trading, application programming interfaces (API), and declining technology costs have led to substantial proliferation of algorithmic trading. Particularly services such as CME Group's Electronic Broking Services have seen the volume of algorithmic trades surpass human made trades at least a decade ago (Chaboud et al. 2023, 265; Hendershott et al. 2011, 4, 30). Algorithmic trading has shown to both increase liquidity and cross-market efficiency but also adds to the heteroskedasticity of price movements (Hendershott et al. 2011, 4, 30).

With today's data processing technology, emphasis on latency and code optimization has become key to exploiting local arbitrage opportunities–it's not just about the best predictions but about getting to the top of the order book (Chaboud et al. 2023, 266; 2024, 580). Importantly, computers are faster than humans at processing information (Chaboud et al. 2014, 2046).

2.1.2 Dynamics of Spot Prices and Macroeconomic Fundamentals

In macroeconomic theory, the logical and causal structure of price discovery–that is the explicit setting of prices between buyers and sellers–is examined through equilibrium models. Key factors impacting equilibrium models for exchange rate demand and supply are inflation, interest rates, economic growth, purchasing power parity, and political risk. (Shapiro & Hanouna 2020, 38-41.)

Tracing back to the sixteen hundreds, purchase power parity or PPP, has longstanding roots in exchange rate literature (ECB 2020, 5; Rogoff 1996, 647, see also Cassel 1918). Modern frameworks of PPP lean on foundational concepts of finance such as arbitrage and the law of one price. In its absolute rendition, PPP posits that arbitrageurs in free markets enforce the law of one price internationally by swiftly capitalizing on price deviations, hence, identical goods will eventually be priced equally across borders (Shapiro & Hanouna 2020, 102-104). Thus, any unit of currency should have the same purchasing power around the world, and a spot rate (S) would be determined formally as:

$$S_t^{(ij)} = \frac{P_t^{(j)}}{P_t^{(i)}}.$$

Here, $S_t^{(ij)}$ represents the spot exchange rate[1] for a unit of currency *i* per currency *j*, and *P* represents the aggregate price index for countries of currency *i* and *j*, at time *t*.

A more prevalent rendition of PPP is relative purchasing power parity. Relative PPP is less extreme and expects that changes in exchange rates reflect changes in relative price levels, or inflation (Shapiro & Hanouna 2020, 38-41). Formally this relation is described as:

$$\frac{S_{t+1}^{(ij)}}{S_t^{(ij)}} = \frac{\left(1 + \pi_t{}^{(j)}\right)}{\left(1 + \pi_t{}^{(i)}\right)},$$

and thus, one can state,

$$S_{t+1}^{(ij)} = S_t^{(ij)} \frac{\left(1 + \pi_t{}^{(j)}\right)}{\left(1 + \pi_t{}^{(i)}\right)}.$$

1 For example, an EUR/USD spot rate of 1.0809, ($S_{3/3/2025}^{\mathrm{EUR,USD}} = 1.0809$) means that one unit of EUR is exchangeable for 1.0809 units of USD, or 1.0809 U.S. Dollars are required for one Euro in the FX market.

Above, $\pi_t{}^{(i)}$and $\pi_t{}^{(j)}$ represent periodic rates of inflation observed over time t to $t+1$. If one assumes a low rate of inflation, it holds that the rate of change in spot rate is expected to be approximately the inflation differential:

$$\frac{S_{t+1}^{(ij)}-S_t^{(ij)}}{S_t^{(ij)}} = \frac{\pi_t{}^{(j)}-\pi_t{}^{(i)}}{1+\pi_t{}^{(i)}} \approx \pi_t{}^{(j)} - \pi_t{}^{(i)}. \quad (1)$$

This implies that as inflation rates change, exchange rates should converge to model equilibrium over time. Thus, real exchange rates should behave as a mean-reverting stationary process (ECB, 2020, 6).

However, empirically absolute PPP is not consistent short-term, but in the long-run, most deviations in PPP dissipate in 4-5 years at best (Rogoff 1996, 664; Xu 2006, 106; ECB 2020, 6). ECB's report (2020, 6-7) outlines that studies on PPP reveal that the velocity of mean reversion to parity observed was usually very slow and occasionally did not occur at all. Overall, PPP over the long-term is a decently employable equilibrium model for making exchange rate approximations (Lothian & Taylor 1996, 503-505; ECB 2020, 6; Shapiro & Hanouna 2020, 112-113).

In the context of the EUR/USD FX market, research presents mixed findings on the role of relative PPP, post-euro adoption. Zhou et al. (2008, 145-148) discovered evidence for relative PPP following the euro's introduction. However, in later research, Kavkler et al. (2016, 618-620) found no evidence of a stable relative PPP relationship among eurozone countries. Pertinently, when forecasting short-term exchange rate movements (1-5 days ahead), Sosvilla-Rivero and García (2005, 374-376) showed that inflation differential based relative PPP models applied using inflation expectations outperforms the random walk model in the EUR/USD market to a marginal but significant extent.

Another important macroeconomic theory concerning exchange rates is the International Fisher Effect (IFE). IFE builds on PPP and the Fisher Effect. The Fisher Effect states that expected inflation $\mathbb{E}[\pi_t]$ and real interest rate a_t constitute the nominal interest rate r_t, formally:

$$1 + r_t = (1 + a_t) \times (1 + \mathbb{E}[\pi_t]).$$

Thus, it also holds that,

$$1 + a_t = \frac{1+r_t}{1+\pi_t}.$$

The generalized Fisher Effect expands upon this relationship by considering cross-border arbitrage. By assuming equivalent levels of risk between countries *i* and *j*, to prevent arbitrage, the real rates of

return must be equivalent, $a_t^{(j)} = a_t^{(i)}$. Thus, the interest rate differential corresponds approximately to the inflation differential to ensure that $a_t^{(j)} = a_t^{(i)}$ (Shapiro & Hanouna 2020, 112-113). Formally:

$$\frac{1 + \mathrm{r_t}^{(j)}}{1 + \mathrm{r_t}^{(i)}} = \frac{1 + \mathbb{E}\left[\pi_t^{(j)}\right]}{1 + \mathbb{E}\left[\pi_t^{(i)}\right]}.$$

Next, by subtracting 1 from both sides and assuming low interest rates, one obtains

$$\frac{\mathrm{r_t}^{(j)} - \mathrm{r_t}^{(i)}}{1 + \mathrm{r_t}^{(i)}} = \frac{\mathbb{E}\left[\pi_t^{(j)}\right] - \mathbb{E}\left[\pi_t^{(i)}\right]}{1 + \mathbb{E}\left[\pi_t^{(i)}\right]} \approx \mathbb{E}\left[\pi_t^{(j)}\right] - \mathbb{E}\left[\pi_t^{(i)}\right]. \quad (2)$$

The generalized Fisher Effect implies that nominal rates account for inflation expectations to prevent arbitrage. This theory posits that currencies with higher interest rates will depreciate because the higher rates reflect higher expected inflation. Furthermore, IFE, as an extension to the Fisher Effect, expects higher inflation to lead to a currency depreciation as in PPP. (Shapiro & Hanouna 2020, 112-120.) Thus, drawing on Equations 1 and 2, one can state that:

$$\frac{S_{t+1}^{(ij)} - S_t^{(ij)}}{S_t^{(ij)}} = \frac{\mathrm{r_t}^{(j)} - \mathrm{r_t}^{(i)}}{1 + \mathrm{r_t}^{(i)}} \approx \mathrm{r_t}^{(j)} - \mathrm{r_t}^{(i)}.$$

Theoretically, the nominal interest differential $\mathrm{r_t}^{(j)} - \mathrm{r_t}^{(i)}$ between countries *j* and *i* is an unbiased predictor of the change in their future spot FX rate.[2] Importantly, IFE relies on the Fisher effect; and changes in nominal interest differential $\mathrm{r_t}^{(j)} - \mathrm{r_t}^{(i)}$can be the result of a deviation from real interest rate parity $a_t^{(j)} = a_t^{(i)}$, or a change in relative inflations expectations $\mathbb{E}\left[\pi_t^{(j)}\right] - \mathbb{E}\left[\pi_t^{(i)}\right]$. These dynamics pull exchange rates in opposite directions. For instance, an increase in inflation expectations will result in depreciation, but an increase in real interest rates will result in increased capital inflows and appreciation. (Shapiro & Hanouna 2020, 119-120.) However, IFE also ignores investment risk which is sensitive to the stability of monetary policy.

Empirically, IFE like PPP has been shown to hold over the long run, but less so in the short run. Hatemi (2009, 120) analyzed IFE in FX market for the British pound-USD currency pair; the results indicate that interest rate differentials respond to inflation rate differentials positively and

---

2 This relation gives rise to Uncovered Interest Rate Parity (UIP): $E\left[S_{t+1}^{(ij)}\right] = S_t^{(ij)} \times \frac{1+\mathrm{r_t}^{(j)}}{1+\mathrm{r_t}^{(i)}}$.

Covered Interest Rate Parity (CIP) extends this to forward contracts stating: $f_{t+1}^{(ij)} = S_t^{(ij)} \times \frac{1+\mathrm{r_t}^{(j)}}{1+\mathrm{r_t}^{(i)}}$.

significantly, but with a coefficient less than one. A report by the ECB (2009) found mixed evidence for IFE. Employing sophisticated cointegration models, the researchers found that in most cases IFE was not statistically significant but occasionally surpassed the coefficient value of one with significant test values (ECB 2009, 18-19). Additionally, Badillo et al. (2009, 118) found that when testing the United States and the Eurozone for IFE, if the Eurozone was analyzed by considering individual countries instead of the Eurozone aggregate, IFE appeared to hold with statistical significance.

Other notable factors identified in literature to effect FX currency-pair spot prices are volatility, forward premiums, the dollar index, GDP, oil prices, market hours and government policy announcements. Among other findings Yang and Wang (2011, 96-105) showcase that different sequential markets during a trading day (see appendix 3) exhibit varying levels of information share proxied by the intraday variance observed by utilizing realized variance (RV) measures. Building on this, Su and Zhang (2018, 49-53) found that information share proxied via RV measures impact particularly the explanatory power of variables such as returns and macroeconomic news on volatility of the AUDUSD (USD to Australian dollar). Chen and Chen (2006, 400-403) employed panel regression methods to display evidence for the predictive power of oil prices to forecast monthly movements of FX rates. Also, Vochozka et al. (2020, 185-186) found evidence for forecasting power of Brent oil price on the EUR/USD. In contrast, Houcine et al. (2020, 236, 241-242) presented results of a one-way causal relationship from EUR/USD to oil prices by employing Granger causality tests on monthly data. Notably, during domestic market trading hours, home currencies tend to depreciate, intraday, to a statistically significant extent (Zhang 2018, 100, 102-104). This depreciation could be due to an increased supply of the domestic currency as domestic actors engage in the market.

Various studies indicate that macroeconomic news has a significant effect on FX rates (Yono et al. 2019; Ben Omrane et al. 2019; Ben Omrane et al. 2020; Plíhal 2021; Ding et al. 2024). Ben Omrane et al. (2019, 16) found that macroeconomic news influences the returns of the EUR/USD with USD depreciating less if U.S. market risk increases (proxied by the VIX-index). This observation is cogent with the USD's role as a safe haven currency. Interestingly, Ben Omrane et al. (2019, 16) observed that FOMC rate decisions surprise is not consistently significant as a factor for 5-minute EUR/USD returns. In a later study Ben Omrane et al. (2020, 95-99) discovered that news from the United States affected the EUR/USD volatility nearly twice as much as European news. Macroeconomic news announcements have been shown to effect intraday variance of the EUR/USD also (Plíhal 2021, 1388-1389). Effects of the FOMC announcements, which fall under the category of macroeconomic news, are discussed in more detail in sub-section 2.2.3.

2.1.3 Forecasting, Technical Indicators and Trading Spot Markets

Due to high liquidity, no fixed lot size, low transaction costs, and no insider trading, FX markets are a unique subset of the global financial markets (Yıldırım et al. 2021, 1-2). This coupled with the FX markets' economic importance makes them incredibly attractive to investors as portrayed by high portion of trade volumes by principal trading firms in the FX markets (Chaboud et al. 2023, 253-257, 264-266). In today's FX markets, algorithmic trading plays a major role in price discovery, increasing both liquidity and market efficiency (Chaboud et al. 2023, 265; 2014, 2045, 2065, 2075; Hendershott et al. 2011, 4, 30; Manahov & Hudson 2014). With heightened market efficiency and competition, it is estimated that 2% of retail FX investors make money trading (Ayitey Junior et al. 2023, 1-2).

Financial forecasting is a crucial area of financial computing that enables predictive market analysis. Forecasts can be used to derive trading strategies, providing recommendations on market actions to generate profit. In the FX context, forecasting techniques can be categorized into fundamental and technical analysis (Shapiro & Hanouna 2020, 127-130; Yıldırım et al. 2021, 7; Ayitey Junior et al. 2023, 2). Fundamental analysis considers macroeconomic variables, such as those discussed in 2.1.2. Technical analysis considers open, high, low, close, and volume data on bid and ask series to identify patterns and extrapolate buy or sell recommendations. Predominantly in recent FX literature, combinations of both forecasting techniques have been implemented in trading and forecasting model research (e.g. Yono et al. 2019, 6; Ling et al. 2021, 4-7; Yıldırım et al. 2021, 7; Ayitey Junior et al. 2023, 2).

In literature, fundamental models appear to be relatively constrained in their ability to forecast short term but provide reliable forecast models for the long term. This mirrors review findings in section 2.1.2. The most barebones fundamental analysis technique relies on mean-reverting behavior implied by PPP. A report by the ECB (2018, 2-3, 10) found that simple models estimating the gradual adjustment to equilibrium reliably out-perform some more sophisticated econometric methods. Employing a simple model based on inflation expectations, proxied via break-even inflation rates, Sosvilla-Rivero and García (2005, 371-372, 376) obtained forecast accuracy with EUR/USD data surpassing the random walk benchmark. Testing a variety of fundamental models, Hsing and Sergi (2009, 200-201) explored the forecasting ability of macroeconomic variables on the EUR/USD. Hsing and Sergi (2009, 204) found that a multivariate equilibrium estimate (Mundell-Fleming model) and PPP provide some of the best forecasting predictions among macroeconomic fundamental models.

Technical analysis is a methodology which assumes that financial markets exhibit recurring patterns. Consequently, identifying and exploiting these regularities can result in profitable trade strategies. Despite its long history and widespread use, technical analysis has faced severe academic scrutiny and skepticism. Malkiel (1973), author of *A Random Walk Down Wall Street*, expresses in his book:

"*Under scientific scrutiny, chart reading must share a pedestal with alchemy*."

Yet, if one ignores the lack of robust rationale behind most of technical analysis methods, research proves many successful implementations of price data pattern recognition algorithms in FX markets (see e.g. Osler 2003; Loginov et al. 2015; Galeshchuk & Mukherjee 2017; Kampouridis & Otero 2017; Luangluewut & Thiennviboon 2023).

An essential component of technical analysis is technical indicators. A common and simple technical indicator used for prediction category is the moving average. There are three typical moving average indicators used in literature, simple (SMA) and exponential moving average (EMA) and the moving average convergence divergence (MACD) (Luangluewut & Thiennviboon 2023, 1-2). Formally the first two moving average indicators are defined as:

$$SMA_{t,K} = \frac{1}{K} \sum_{k=t-K}^{t} S_k^{(ij)} \tag{3}$$

$$EMA_{t,K} = (1 - \alpha) \times EMA_{t-1} + \alpha \times S_t^{(ij)} \tag{4}$$

Here $S_t^{(ij)}$ represents the spot rate for a unit of currency *i* per currency *j*, and K represents the number of lagged periods. Smoothing factor α is defined as $\alpha = 2/(\mathrm{K}+1)$, and $EMA_0 = S_0^{(ij)}$. When predicting market movement, usually two moving averages are analyzed. The first moving average is lagged at a shorter period K_{lead}, whilst the second moving average at a longer period K_{lag}. Once the lead period indicator surpasses (falls under) the lag indicator, an upward (downward) trend is expected (Luangluewut & Thiennviboon 2023, 1-2). The leading indicator can be replaced with an MACD indicator to measure momentum. An EMA based MACD indicator can be formalized as:

$$MACD_t = EMA_{t,K_{lead}} - EMA_{t,K_{lag}} \tag{5}$$

Additionally, moving averages form the basis for another technical indicator Bollinger Bands (BB). BBs provide an upper- and lower-band for price expectations based on a moving average and volatility (Butler & Kazakov 2010, 505-506). Formally, typical BB-bands can be expressed as:

$$BB = [\hat{\mu}_t \pm b\ \hat{\sigma}_t] = \left[BB_{t,Upper}, BB_{t,Lower}\right] \quad (6)$$

In Equation 6, $\hat{\mu}_t$ denotes a selected estimator for the mean price (e.g. SMA) and $\hat{\sigma}_t$ denotes the estimator for volatility, at time *t*. The scaling factor b is usually set at 2 or 1.96. Pertinently, BBs capture sudden fluctuations, relative to prevailing market volatility. BB's performance, particularly in rule-based trading, is weak (Leung & Chong 2003, 340-341). Butler and Kazakov (2010) found in their experiments that basic BB trading rules tend to underperform buy-and-hold strategies.

BBs belong to a group of indicators that rely on retracement. Retracement is the expectation that prices tend to move between an upper and lower bound indicator (Loginov et al. 2015, 1). The upper bound, called the resistance, is a price point before which price is expected to fall, and the lower bound is called the support, a price point, before which the price is expected to increase (Kampouridis & Otero 2017, 2-3). Essentially, a trend is expected to reverse course before hitting the support or resistance. Osler (2003, 1815-1816) highlighted that clustering stop-loss and take-profit orders around round prices lends support to the base assumptions of retracement. However, some strategies expect that when retracement levels are breached, new pertinent information was realized, and thus prices will increase past resistance or fall below the support level (Kampouridis & Otero 2017, 3).

A simple retracement support and resistance (SR) method is the pivot point. The pivot point uses the average of the previous trading sessions High (H_{s-1}), Low (L_{s-1}), and Close (C_{s-1}) prices, to set SR levels. The pivot point is formalized as:

$$PivotPoint_t = (H_{s-1} + L_{s-1} + C_{s-1})/3 \quad (7)$$

Here s-1 denotes the last sessions index relative to the current session, s (see appendix 3). Using the pivot point, SR levels can be set at multiple levels (Loginov et al. 2015, 3-4). Some common SR levels used in previous studies can be formalized as:

$$ResistanceLevel1_{t,PP} = 2PivotPoint - L_{s-1} \quad (8)$$

$$SupportLevel1_{t,PP} = 2PivotPoint - H_{s-1} \quad (9)$$

$$ResistanceLevel2_{t,PP} = PivotPoint - L_{s-1} + H_{s-1} \quad (10)$$

$$SupportLevel2_{t,PP} = PivotPoint - H_{s-1} + L_{s-1} \quad (11)$$

$$ResistanceLevel3_{t,PP} = PivotPoint - SupportLevel1_{t,PP} + ResistanceLevel2_{t,PP} \quad (12)$$

$$SupportLevel3_{t,PP} = PivotPoint - ResistanceLevel2_{t,PP} + SupportLevel1_{t,PP} \quad (13)$$

Here, time index t belongs to the current trading session s. Another popular class of SR methods are Fibonacci ratios based on the Fibonacci sequence (Loginov et al. 2015, 3-4). This strategy involves identifying previous highs (lows) to gauge support (resistance) levels. Usually, a multiplier is applied to the midpoint of the previous high and low, based on the golden ratio ≈ 1.618, to obtain the respective support or resistance level (Loginov et al. 2015, 2-3). Simple Fibonacci-ratio SRs can be expressed as:

$$H_{max,K} = max\{H_{t-1}, H_{t-2}, \dots, H_{t-K}\} \quad (14)$$

$$L_{min,K} = min\{L_{t-1}, L_{t-2}, \dots, L_{t-K}\} \quad (15)$$

$$ResistanceLevel_{t,Fib} = 1/1.618 \times \frac{1}{2}(H_{max,K} + L_{min,K}) \quad (16)$$

$$SupportLevel_{t,Fib} = (1 - 1/1.618) \times \frac{1}{2}(H_{max,K} + L_{min,K}) \quad (17)$$

Loginov et al. (2015, 2) underline that SR indicators are best employed when they are not used to constructing exact trading rules. Instead, SR should be applied adaptively so that trading rules are revised and possibly conditioned on the prevailing market conditions (Loginov et al. 2015, 2). Overall SR are considered not as specific retracement levels, but as indicators that depict the general areas of support and resistance that might hold if the previous trend were to continue (Schlossberg 2006, 106).

Though technical indicators and analysis methods are peculiar in their lack of scientific basis, previous research revealed that they can still be profitably employed in trading. Especially, the use of technical indicators as inputs for ML methods is increasingly prolific. Using an ensemble of decision trees on trade indicators, Loginov et al. (2015, 9-12) managed to create profitable trades in annual backtests spanning 2010-2013 intraday EUR/USD data. Interestingly, Convolutional Neural Networks (CNN) have been used to read intraday EUR/USD charts and forecast the direction of prices. Comparing technical analysis methods and a time series 2D line plot interpreting CNN, Luangluewut and Thiennviboon (2023, 3-4) created an accurate direction forecast model benchmarked using EUR/USD minute data. To apply the CNN, Luangluewut and Thiennviboon (2023, 3-4) first constructed the plots with 256 lagged prices and then generated their train and test data to run their model.

Another study by Fisichella and Garolla (2021) took a similar approach, combining 20 common technical indicators and a CNN for interpreting candlestick charts with a genetic algorithm to select and amalgamate signals and various trading rules to create a trading system. Fisichella and Garolla (2021) managed to create profitable simulated trades with their trading system when backtesting on EUR/USD data. Additionally, Galeshchuk and Mukherjee (2017, 104-108) found that a deep CNN implemented with one-dimensional convolution (single lagged time series feature) significantly outperformed the next most accurate model, a deep Artificial Neural Network (ANN) trained on moving average features, when forecasting FX rate direction. Galeshchuk and Mukherjee (2017, 109) posit that this may be due to the deep CNNs ability to abstract technical indicators such as SMAs in the initial one feature convolutional layers (implemented with Conv1D layers in Keras).

Pertinent to this thesis, literature highlights the increased use of combined fundamental and technical analysis in forecasting and trading (Yono et al. 2019, 6; Ling et al. 2021, 4-7; Yıldırım et al. 2021, 7; Ayitey Junior et al. 2023, 2; Ding et al. 2024). Yono et al. (2019, 3) constructed topic models on exchange news summaries and compared the FX spot price direction forecast performance of the base price feature models and the combined (hybrid) price feature and topic models. The performance of price feature models was significantly enhanced when news was considered (Yono et al. 2019, 6-7). Constructing a hybrid ensemble method using eight different prediction models, Ling et al. (2021, 12-15) managed to outperform benchmark models in forecasting the USDJPY (USD to Japanese Yen) and USDCAD (USD to Canadian dollar) FX rates. Base models employed by Ling et al. (2021) in the ensemble used both macroeconomic and technical variables to forecast and simulate trades in FX markets. Researchers found that interest and inflation rates play an important role in determining model forecasts (Ling et al. 2021, 16-18). Thus, considering both fundamental and technical analysis appears to be beneficial for designing FX forecasts or trading models.

2.2 The Federal Open Market Committee and Monetary Policy Announcements

The Federal Open Market Committee is a key entity of the United States Federal Reserve Bank. Committee is responsible for directing the monetary policy of Reserve Bank (Federal Reserve 2025a), and thus its releases are closely followed by participants of financial markets (Shapiro & Hanouna 2020, 28; IMF 2023, 774). This Section of the thesis explores these themes and is split into two subsections: 1) United States monetary policy, the Fed, statements, and minutes; and 2) the effects of statements and minutes releases on financial markets.

2.2.1 United States Monetary Policy, The Fed, Statements, and Minutes

As highlighted in the subsections, EUR/USD FX rate is sensitive to both interest and inflation rates which are central elements of monetary policy (Federal Reserve 2025a). Hence, it can be confidently posited that domestic monetary policy impacts the global FX markets. In the United States, the Congress instructs the Federal Reserve System (Fed) via the Federal Reserve Act to conduct monetary policy with the purpose of promoting "the goals of maximum employment, stable prices, and moderate long-term interest rates." Additionally, The Fed follows the 1978 Humphrey-Hawkins Act which identifies maximum employment and price stability, specifically, as the top economic priorities. These two legislative acts are widely cited as the origin of the Fed's dual mandate for maximum employment and price stability (Ireland, 2024, 2.) The Fed carries out its dual mandate primarily by controlling reserve requirements and the federal funds rate, among other policy tools.

The Fed consists of 12 reserve banks and their 24 branches and operates as decentralized system of regional central banks –as per the Federal Reserve Act. The three key entities of the Fed are the Federal Reserve Board of Governors (Board of Governors), the Federal Reserve Banks (Reserve Banks), and the Federal Open Market Committee (FOMC). The Board of Governors is an independent agency accountable on behalf of the Fed to the United States Congress. The Board of Governors resides in Washington, D.C., with Governors appointed by the sitting United States President and confirmed by the Senate. The Board of Governors provides guidance and oversees the 12 reserve banks of the Fed. The Board of Governors has seven appointed Governors, serving 14-year terms. The Board's Chair and Vice-Chair are also appointed by the president and confirmed by senate to serve four-year terms. Reserve banks are chiefly responsible for gathering data on the local economy, businesses, and households from their communities. (Federal Reserve 2021, 2-5, 7-10.)

A major institution of the Fed, responsible for directing monetary policy, is the Federal Open Market Committee (FOMC). The FOMC convenes eight times a year, scheduled to be held roughly every six weeks, to review economic and financial conditions and set an appropriate stance on monetary policy (Federal Reserve 2025a). Occasionally, additional meetings are held in face of surprising or difficult economic or financial environments. During its meetings the FOMC discusses trends in the United States economy, reviews reports made by staff and outside sources, and analyze the most recent economic and financial market data to judge an appropriate stance on monetary policy and the risks of implementing the stance (Federal Reserve 2021, 28-29).

Once policy stance is determined, the FOMC ensures that the Fed implements this stance effectively. As part of the stance, the FOMC determines the appropriate target federal funds rates towards which

short-term interest rates are encouraged via policy tools such as reserve mandates, overnight reverse repurchase facility rates, and various other tools. The FOMC consists of twelve members: the seven members of the Board of Governors, the president of the Reserve Bank of New York, and four of the eleven remaining regional heads of reserve banks (who rotate in for 1-year terms). In addition to determining the Fed's policy stance, the FOMC directs the Fed's operations in FX markets and authorizes swap programs with foreign central banks. (Federal Reserve 2021, 2, 8, 12-13.)

Central banks around the world, especially in the United States, have trended towards greater disclosure and accountability (Ferrera et al. 2022, 2-3). Ferrera et al. (2022, 3) argue that this trend stems from both the development of a scientific framework concerning positive effects from disclosing public monetary policy in managing economic expectations, and the increase in accountability and transparency brought by public announcements. Measuring uncertainty using decomposed forecast error for CPI, and Jitmaneeroj et al. (2019, 222, 227-236) showed that monetary policy disclosures, especially in the form of future guidance, reduce forecast-based uncertainty. Evidently, with increasingly certain inflation expectations, actors in FX markets can create more appropriate pricing models. Using panel data on 62 currency pairs, Eichler and Littke (2018, 30-31) found that central bank communications on monetary policy objectives, both with direct numeric target values and indirect communication about models and rationale, resulted in dampened FX volatility. The study by Eichler and Littke (2018, 37-39, 44) particularly highlights the stability of inflation expectations as a conduit for low FX volatility. Employing various linguistic analysis measures, Vyshnevskyi et al. (2024, 4-13) found significant effects of monetary policy announcement complexity and readability on FX volatility. Transparency of announcements on monetary policy impact FX markets. Regarding the EUR/USD, the FOMC is a key source of such announcements. Historically, there have been only two regular scheduled releases by the Fed that occur during trading hours in the United States: the FOMC statement and minutes (Jubinski & Tomljanovich 2017, 702).

The most important disclosure on the United States monetary policy stance is the FOMC statement, released soon or immediately after the FOMC meeting, usually mid-trading day (see Andersen et al. 2003, 40-41; Federal Reserve 2021, 29; Federal Reserve 2024; Federal Reserve 2025a, 2025b). The FOMC statement summarizes the Committees current judgement on the outlook and trends in the United States economy, presents the policy decision and stance, and informs the public about factors that the Committee will consider in conducting future monetary policy action (Federal Reserve 2021, 29). Thus, statements disclose pertinent monetary policy information concisely, in a timely manner.

Though monetary policy statements are meant to provide clarity, the language of the Fed is to an extent strategically obfuscated. The history of obfuscation in monetary policy is long, with the last 30 years representing a shift towards more transparent communication (Ferrera et al 2022, 1-3). The push for FOMCs transparency under the former Chairman Alan Greenspan was initially met with resistance (see e.g. Kansas City Federal Reserve 2016, 3-6). Before FOMC policy statements, Dr. Greenspan was famously quoted in the New York Times (St. Louis Federal Reserve, 2019) stating:

> "*If I turn out to be particularly clear, you've probably misunderstood what I said*."
> (Adam Greenspan, 1988)

It was initially speculated that Dr. Greenspan intended to dampen market overcorrections by obfuscating the Fed's policy intentions (St. Louis Federal Reserve, 2019). This sentiment, though left unsaid, carried over to 2007 with secrecy seen as a policy tool–Dr. Greenspan referred to secrecy then as "constructive ambiguity" (Kansas City Federal Reserve 2016, 4). Nowadays, the legacy of secrecy and obfuscation is increasingly being replaced by transparency (Ferrera et al. 2022, 1-4). Especially the use of forward-looking language, information about future policy inclinations, and the current economic outlook (Lunsford 2020, 2930-2931) reflects a shift towards less obfuscatory language. Yet, in light of this legacy of Fed speak, it can be assumed that the nuanced language inherent in FOMC releases requires sophisticated NLP methods to be accurately modeled.

The St. Louis Fed (2019) lists the six-feature structure of the FOMC statement; these features are:

1. Recent economic developments: What has happened since the last meeting?
2. Monetary policy goals: Focus on the dual mandate.
3. Policy decision: What decision was made at the meeting?
4. Forward guidance: Communication about the likely future course of monetary policy.
5. The decision process: Factors the FOMC plans to consider in upcoming policy decisions.
6. Voting record: Who voted and how?

In 1994 the FOMC began to announce their policy decisions with public statements on target rate on the day of the meeting at irregular times (Andersen et al. 2003, 40-41; Federal Reserve 2021, 29). These statements were initially only released if the FOMC's stance on monetary policy had changed. Until, in January 2000 when statements were release immediately after each meeting (Federal Reserve 2021, 29; New York Federal Reserve 2011, 3-5; Federal Reserve 2025b). Also, in 1995, the

FOMC began to announce the target rate somewhat consistently around 14:15 Eastern Time (ET) on day of the meeting (Andersen et al. 2003, 41). From 2011 until the second meeting in 2013 statements released between 12:30 and 14:15 ET (New York Federal Reserve 2021, 5). After the second meeting in 2013, the FOMC has released statements exactly at 14:00 ET (Federal Reserve 2024).

Another significant piece of Fed communication is FOMC minutes. Whilst statements provide rationale for policy decisions and information about the Fed's outlook on policy decisions, the minutes provide a summary of discussions from the FOMC meeting. Minutes include in-depth descriptions about policy issues and the Committee members' views on the monetary policy stance, their outlook on the United States economy, and short-term policy inclination; among other collective viewpoints on meeting discussions items (Rosa 2013, 67-68; Federal Reserve 2025b). Thus, minutes present a more detailed accounts on FOMC meeting, serving as a rich source of information, especially on the expected health of the United States' and global economy (Jubinski & Tomljanovich 2017, 702). It is likely that minutes present new information concerning FOMC's longer-run outlook and short-run policy path (New York Federal Reserve 2006, 24), direction of interest rates, and other macroeconomic variables (Jubinski & Tomljanovich 2013, 86). Consequently, minutes tend to be lengthier than statements (See Figure 1 below, the text dataset is discussed in subsection 4.1.2).

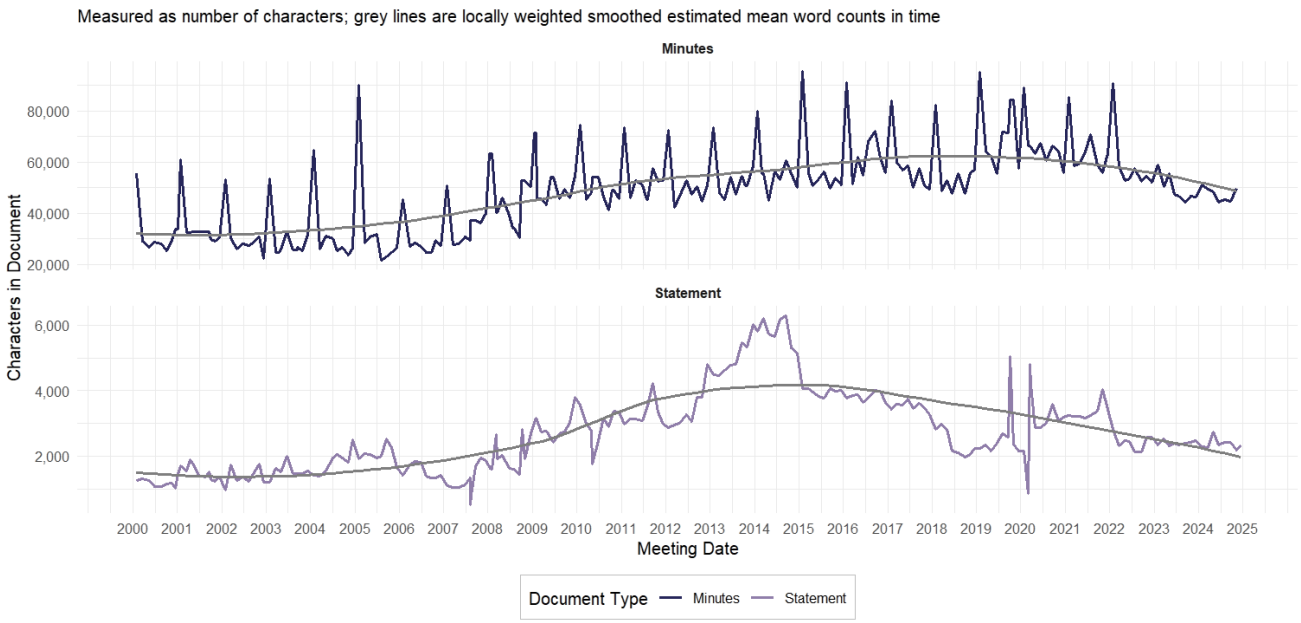

Figure 1: Trends in FOMC Statements and Minutes Document Size (Data from Federal Reserve 2024)

FOMC statements tend to be relatively concise usually less than a page in length and a couple of hundred words, whilst minutes are frequently at least a dozen pages long with thousands of words of information contained within (Jubinski & Tomljanovich 2017, 703; Federal Reserve 2024). Overall,

FOMC minutes, like the FOMC statements are quite structured, but also lengthy. Minutes have routinely consisted of four distinct sections. The first section concerns administrative details of the meeting, and reviews previous open market operations. The second section contains remarks on the pre-prepared staff's review, and the economic and financial market outlook. The third, consists of details about the FOMC members' discussion and presentations of their own additional projections. The fourth section includes the policy decision, decision rationale, future policy inclinations and current policy outlook. (Jegadeesh & Wu 2017, 5.)

Minutes of the first meeting of the year tend to be lengthier by wordcount, due to details on annual administrative issues (such as FOMC appointments) and policy framework reviews (Federal Reserve 2024). This causes evenly spaced spikes in document length, depicted in Figure 1. Before December 14th, 2004, minutes were released within two-three days of the next meeting, corresponding to a six-eight-week lag (New York Federal Reserve 2006, 10). Since 2005, minutes releases have occurred three weeks after the meeting (New York Federal Reserve 2006, 10; Federal Reserve 2025b). Today minutes are typically released three weeks after the meeting also at 2:00 pm (New York Federal Reserve 2006, 10; Rosa 2013, 69; Federal Reserve 2025b).

2.2.2 Effects of Statements and Minutes Releases on Financial Markets

Previous literature reveals that numerous financial markets are sensitive to information contained within FOMC statements and minutes and their release (see e.g. IMF 2023, Rosa 202). Provided the global nature of markets, international investments, lending, and trade activity flows to markets with the best opportunities (Shapiro & Hanouna 2020, 330, 334-340). Thus, news relating to monetary policy concerning the economy of the United States with its liquid world reserve currency, is impactful in today's global markets (Shapiro & Hanouna 2020, 28; IMF 2023, 774; BIS 2025, 3-7). Table 1 summarizes financial markets affected by FOMC statements and minutes releases.

Table 1: Effects of FOMC Releases on Financial Markets from Literature Review

Document / Markets	*Evidence Provided by*
Statements	
Bond Markets	Hansen & McMahone 2016
Equity Markets	Rosa 2013; Hansen & McMahone 2016; Brusa et al. 2020; IMF 2023; Shah et al. 2023
Federal Funds Rate	Mueller et al. 2017; Tse et al. 2019; IMF 2023; Osowska & Wójcik 2024
FX Markets	Rosa 2013; Mueller et al. 2017; Tse et al. 2019; IMF 2023; Osowska & Wójcik 2024
U.S. Treasuries	Hansen & McMahone 2016; Shah et al. 2023

Table 1 continued

Minutes	
Bond Market	Jegadeesh & Wu 2017
Equity Market	Rosa 2013; Jubinski &Tomljanovich 2013;2017; Jegadeesh & Wu 2017
Federal Funds Rate	Aattouchi & Kerroum 2022
FX Market	Rosa 2013
U.S. Treasuries	Rosa 2013

Note*: All FX market findings listed in this table discuss the EUR/USD as a currency pair affected by the particular type of FOMC document.*

The FOMC announcements, chiefly the release of statements, have a significant effect on global and domestic equity risk premia, distinct from announcements of other central banks such as the Bank of Japan, the Bank of England, the ECB, and others (Brusa et al. 2020, 264-266, 301-302). According to a research paper published by the International Monetary Fund i.e., the IMF (2023, 775, 798), investors may generally allocate funds to the United States when the Fed signals weakness about the state of the economy, conjectured as a flight-to-safety, as concerns of a global recession draw investment capital to the United States. On the other hand, investment capital flowed to global equities when the Fed signaled strong performance of the United States economy. (IMF 2023, 798.)

The IMF (2023, 775) paper stated that FOMC announcement surprise can be categorized into monetary policy shocks and central bank communication shocks, whereby a monetary policy shock is disentangled from central bank communications shock by proxy of market reactions. The key theoretical difference between these two is that the former (monetary policy shock) concerns the implementation of unexpected monetary policy, and the latter (central bank communications shock) concerns unexpected assessment of economic outlook which justifies a policy decision (IMF 2023, 774). The IMF (2023, 779) gauged market reactions by measuring changes in S&P500[3] returns while employing the three-month federal funds rate as a policy indicator –both measures used intraday minutely data. For example, monetary policy shock was identified as an unexpected rise in the federal funds rate which resulted in a decline of the S&P500, while a central bank communication shock was identified as an unexpected rise in the federal funds rate which caused an increase of the S&P500. The IMF (2023) paper's event study revealed that a nominal interest rate increase and a monetary policy shock, generally, resulted in an appreciation of the USD against 26 reference currencies, while a nominal interest rate increase paired with a central bank shock caused depreciation.

3 The S&P500 is an index measuring performance of the large-cap segment of U.S. equities markets. Considered to be a proxy of the U.S. equity markets, the index is composed of 500 constituent companies. (S&P Global, 2025, 4.)

A staple piece of literature on the intraday volatility effects of FOMC minutes releases was written by Carlo Rosa in 2013. Rosa (2013, 74-77) investigated the intraday effects of FOMC minutes releases on treasuries, the S&P500, and a couple of currencies. Interestingly, currencies appeared to be sensitive to the minutes' releases, more so than the United States equity index, the S&P500. Volatility appeared to spike at the release around 14:00 ET (Rosa 2023, 70-71). Volatility effects were temporary, suggesting that the FX markets integrate the FOMC releases information into pricing expediently and accurately. Particularly for the EUR/USD, market reactions appeared increasingly intermittent, with significantly high volatility lasting roughly 30minutes (Rosa 2013, 70-71).

Jubinski and Tomljanovich (2013) initially explored the intraday returns and volatility effects of FOMC minutes releases on 2832 United States equities between 2006 and 2007. Their paper highlighted that FOMC minutes impact intraday volatility but not returns significantly. This volatility impact differs across equities when considering factors such as industry and company size. Pertinent to this thesis, the researchers documented asymmetric volatility responses to minutes releases when they occurred after a target rate increase, decrease, or when no change was made. (Jubinski & Tomljanovich 2013, 86-88, 91-93, 96-97.) Jubinski and Tomljanovich (2017, 701, 703) later investigated the intraday effects of FOMC statement and minutes releases again on volatility and returns of 1400 individual United States equities from 2004 to 2015. Akin to earlier findings, their paper found no evidence for equity returns in aggregate being effected by FOMC policy announcements within 5 minutes of document release. Instead, this study showed that 30minutes after document release variance rose sharply, especially for highly traded large firms, with both statements and minutes sparking market movements. Understandably, market volatility increase was reportedly more sensitive to FOMC statements than minutes. (Jubinski & Tomljanovich 2017, 722.) This is largely aligned with the discussion in subsection 2.2.2, where the statements were presented as the first piece of Fed communication on the FOMC policy decisions. Yet both studies provide evidence that FOMC minutes crucially do inform the market participants' intraday trading activities.

According to Mueller et al. (2017, 1214), the USD vis-à-vis different currencies, experienced significant excess returns around FOMC statement releases due to FX market uncertainty. The researchers found that a profitable trading strategy could be constructed, where a trader would short the USD to long other currencies. Specifically, they back tested a strategy involving buying a portfolio of 1-month foreign currency forwards and later selling the underlying currencies at the future 1-month spot, to try and make a profit on the forward discount.[4] This forward discount was expected to occur

4 Without direct data on forward contracts, Mueller et al. (2017, 1221-1222) relied on CIP for contract price estimates.

as a function of policy uncertainty. The strategy and subsequent statistical tests were implemented within high-frequency data spanning 1994 to 2013 on the ten most traded USD currency pairs. The average daily returns on the FOMC announcement day with the above strategy with currencies of low interest rate differentials vis-à-vis the United States was 5.19 basis points (bps), while on currencies with high interest rate differentials it was 14.47bps. When this strategy was modified to only long currencies that have a positive interest rate differential (higher interest rate than the United State), the new strategy resulted in 19.49bps and 22.54bps returns for high- and low interest rate differentials respectively. These results are notable, especially for the making the case for tracking interest differentials, as the average daily returns on non-announcement days were -0.51bps and 1.73bps, respectively. Splitting the announcement day to pre- and post-announcement windows, the trading strategy remained profitable. (Mueller et al. 2017, 1214-1216, 1221-1222, 1234, 1245-1246.)

The paper by Mueller et al. (2017) posited that an increase in uncertainty of the federal funds rate before announcements and the subsequent clarity on monetary policy post announcement, results in higher excess returns for foreign (non-United States) currencies as market participants are rewarded for policy uncertainty risk. Additionally, excess returns for foreign currencies with higher interest rate differentials are significantly better, potentially because financiers' risk constraints limit their actions, and thus FX markets cannot adjust one-to-one to the interest differential. (Mueller et al. 2017, 1214-1216, 1221-1222, 1234, 1245-1246.) I find that this latter argument is aligned with the findings of Hatemi (2009, 120) and the ECB (2009) paper. Interestingly, the paper also highlighted that expansionary (contractionary) monetary policy, codified as a lowering of the federal funds target rate, increased (decreased) returns (Mueller et al. 2017, 1214-1216, 1235-1236).

Expanding on this, Tse et al. (2019) evaluated log-returns on futures contracts made for USD pairs with G10 currencies and six emerging market currencies: BRL (Brazilian real), MXP (Mexican peso), KRW (Korean won), RUB (russian ruble), TRY (Turkish lira), and ZAR (South African rand). The researchers proposed that the increased volatility and market risk related to developing markets would heighten the risk premiums, thus leading to lucrative returns on successful trades under the FOMC statement announcement strategy devised by Mueller et al. (2017). He constructed two portfolios based on past futures price returns, one designated high-yield and another low-yield based on high- or low-futures price log differences. Additionally, he constructed an equally weighted portfolio of currencies for developed countries and for emerging countries. Tse reported that announcement day returns were significant for all currencies except for the JPY. The portfolios had varying degrees of success, the high-yield portfolio generated significant returns whilst the low-yield portfolio did not, also the emerging market portfolio achieved significant returns. Tse discovered as, did Mueller et al.

(2017, 1236), that particularly expansionary monetary policy shocks[5] increase returns. Tse et al. also found evidence that the strategy on emerging market currencies yielded higher returns, but mostly when FOMC policy shocks were expansionary. (Tse 2019, 1590-1591, 1595-1596.)

In addition to event date and target rate surprises brought about by FOMC minutes, and particularly statements releases, various NLP studies reveal that the nuanced textual features of these documents matter to various types of financial markets (Hansen & McMahone 2016; Jegadeesh & Wu 2017; Aattouchi & Kerroum 2022; Osowska & Wójcik 2024; Ko 2024). Computational resources have become readily available, nascent NLP research and methods have substantially contributed to the ever-increasing toolkit and the opportunities arising from testing these newfound tools within the context of FOMC releases empirically. The rest of this section describes studies on the selected NLP methods employed, STMs and BERT, to investigate the effects of FOMC statements and minutes on various markets. NLP models themselves are explored in detail later in section 3.2.

Topic modelling in the realm of FOMC pronouncements has seen some investigation. Hansen and McMahone (2016) explored impacts of sentiment around topics found from FOMC statements with a Latent Dirichlet Allocation (LDA) topic model on various macroeconomic variables. The LDA was implemented with 15 topics with FOMC statements released from 1998 up to March 2015. The researchers selected economic topics found within the FOMC statements for sentiment analysis to gauge the economic situation as reported by the FOMC. To accompany this measure, the researchers also created a measure to analyze the expansionary and contractionary sentiment of the forward guidance weighted by a dictionary method-based uncertainty measure and the portion of words used for future guidance within the statement. The study concluded that the economic situation or forwards guidance reported by the FOMC statements as gauged by the measures had not caused particularly strong effects on real economic variables. (Hansen & McMahone 2016, 115, 116-121.)

Jegadeesh and Wu (2017) investigated the high frequency effects of FOMC minutes contents on bond and equity markets. The researchers employed an LDA model on the paragraph level and calculated the tonality of each paragraph using dictionary-based methods. These two methods were used to score the tonality per topical content identified within the FOMC minutes. The researchers found that for

[5] Shocks were measured as sudden changes to the federal funds rate, following Kuttner (2001) and defined as one-day change in the one-month federal funds rate futures settlement price. Formally measured using the equation:

$$Shock_t = [D/(D-t)](f_t^\circ - f_{t-1}^\circ),$$

where, D denotes the days in the month of the FOMC announcement, t, the event day (of the announcement month), f_t° is 100 minus the settlement price of the nearby federal funds futures on day t. (Tse 2019, 1591.)

their purposes eight topics enabled them to dissect minutes into distinct and meaningful topics. They found that the topics extracted by their LDA had significant relationships on market volatilities. In their conclusion the researchers state that the study supports that the Fed's superior information on market conditions gets transmitted via minutes releases. (Jegadeesh & Wu 2017, 9-12, 16-17, 33.)

Moreover, Aattouchi and Kerroum (2022) implemented a similar topic-sentiment approach to forecast the Fed Funds rates. The researchers extracted topics of interest with a focus on inflation to first reduce the dimensionality of the minutes, then summarized the topic-sentiments with dictionary-based methods. These measures were used to predict the Fed Funds rate with 98.78 percent accuracy using a CNN regression model. (Aattouchi & Kerroum 2022, 125-129.)

Furthermore, financial sentiment research has seen a shift from dictionary-based methods towards more technical transformer-NLP methods. As highlighted by, for example Shah et al. (2023), Gössi et al. (2023), and Kim et al. (2024), these ML methods consistently beat dictionary-based methods in classifying FOMC excerpts as well as human experts. Open-source BERT models specialized for sentient analysis on financial texts have been around at least since 2019 (see e.g. Araci 2019), but open-source FOMC specific models are fairly nascent as demonstrated by the publication dates of the FOMC models. Simultaneously, particularly BERT models are starting to rear their heads in demonstrating the effects of sentiments extracted from FOMC releases on various markets.

For example, Ko (2024) showed that sentiment factors extracted from congressional testimony by the Chairperson, and speeches by bank presidents accounted for 78-88% of the 1-month ahead variance in modelling the affine term structure of U.S. Treasury bonds. The researcher estimated a discreate time affine model using various bond yields, inflation and real activity measures, as well as the BERT sentiment scores. The parameter estimates for the FOMC sentiment factors were all highly significant, lending to the effectiveness of BERT sentiment modelling for FOMC releases. (Ko 2024, 1-3, 5.)

A pertinent piece of research concerning equity and FX market prediction with FOMC statements was published by Osowska and Wójcik (2024). The researchers implemented NLP sentiment analysis to score statement sentiments on a scale from positive (1) to negative (-1) using a pre-trained FinBERT transformer model (Araci 2019). In conjunction with sentiment, researchers implemented macroeconomic variables such as changes in the Fed funds rate, interest rate surprise, inflation measures, and consumer confidence. The sentiment and supporting variables were fed as inputs to various ML models to produce minute-level regression predictions for both the S&P500 and EUR/USD. Random Forest (RF) algorithms appeared to provide the most accurate results, but not very consistently for the EUR/USD. (Osowska & Wójcik 2024, 157, 160-165, 167-168.)

3 Foundations of Classification, Regression, and Selected NLP Models

This section is broadly split into two topics of interest: (1) fundamentals of ML and ANNs, and (2) natural language processing (NLP). Deisenroth et al. (2020, 3) define the goal of ML as finding a good enough model that generalizes well onto yet unseen data that we may care about in the future. Thus, it appears that ML is aligned with goals of FX trading. Yet it is important to note that in the end all models are just models, and as the aphorism attributed to statistician George E.P. Box goes,

> "All models are wrong, but some are useful."

Carrying with this sentiment, this thesis not only implements models, but expands on the reasoning and mechanisms underlying the selected ML algorithms to make proper and informed use of them.

3.1 Classification, Regression, and The Selected ML Prediction Models

To predict the EUR/USD market signals, the empirical exploration of this thesis integrates a variety of ML models with particularly the most advanced ML models relying on feed-forward ANNs. This section briefly overviews some fundamentals of ML and pertinent ANNs and other models used to provide contextual background for the empirical sections of this thesis. The first subsection 3.1.1 discusses the fundamentals of ML with a minor emphasis on regression tasks, the second subsection 3.1.2 discusses classification models and model selection, and the third subsection 3.1.3 discusses the three types of ANNs implemented in this thesis.

3.1.1 Machine Learning in a Nutshell: *Tasks, Learning Algorithms, and Loss*

Mohri et al. (2018, 1) state that ML can be defined as a set of computational methods that use experience to improve performance or make accurate predictions. Conversely, in his dissertation, Botsch (2023, 5) defines ML more holistically as the branch of science focusing on the automatic discovery, or modelling, of regularities in data via the implementation of *learning* computer algorithms. Deisenroth et al. (2020, 2-3) highlights that in the context of ML, algorithms can be understood in two ways: 1) a system that generates predictions using input data or 2) a system that *learns* (via training) the realization of a model that generalizes well onto future unseen input data. Hence, outcomes of good ML algorithms are models that can be viewed as simplifications of real

(unknown) data-generating processes, which capture relevant information and extract hidden patterns within the data to produce reliable results for the task at hand (Deisenroth et al. 2020, 2-5).

In today's data-driven world, there are a plethora of data-generating processes which might be of interest. To guide the implementation of ML and select a suitable model, it is useful to distinguish ML tasks that help frame the approach to modelling data-generating processes of interest. Categories of ML tasks can be discussed within framework of four primary task types, or pillars as Deisenroth et al. (2020, 225) put it. These task types are regression, classification, dimensionality reduction and density estimation. (Deisenroth et al. 2020, 225-227.) For the purpose of this thesis regression, classification, and dimensionality reduction are primarily relevant, though all are used to an extent.

The objective of classification tasks is to find a good approximation of the mapping from inputs $\boldsymbol{x}_t$ to the classification labels $\boldsymbol{c} \in \mathcal{Y}^N$(Botsch 2023, 9). For example, in binary classification labels are observed as either ones or zeros, such that $\mathcal{Y} = \{1,0\}$. The problem of classification is assigning each instance of the data to the correct category. This framework can be applied to multiclass classification where images could be assigned to categories such as *plane, train,* or *car (*Mohri et al. 2018, 3).

In regression tasks the objective is to find a model that maps inputs $\boldsymbol{x}_t$ to a prediction of the real value label $y_t \in \mathbb{R}$ as accurately as possible (Deisenroth et al. 2020, 260; Mohri et al. 2018, 265). Because regression tasks concern real value data, observations of the data tend to contain noise. Therefore, the selected model is required to make predictions that are as close to the correct ones as possible (Deisenroth et al. 2020, 261; Mohri et al. 2018, 266-268). One of the simplest models used for regression tasks is linear regression. From the perspective of ML, the typical Ordinary Least Squares (OLS) linear regression model seeks to minimize squared error by learning the optimal parameters. These parameters are weights $\boldsymbol{w}^* \in \mathbb{R}^D$ and a bias term $b \in \mathbb{R}$, that map input features $\boldsymbol{x}_t$ to the label y_t accurately. Formally $\hat{y}_t = \boldsymbol{x_t}'\boldsymbol{w}^* + b$ (Mohri et al. 2018, 275).

Dimensionality reduction concerns data transformation, where the initial data is projected onto a lower-dimensional representation, preserving or condensing important properties of the data (Mohri et al. 2018, 3). Working with high-dimensional data can be expensive and difficult to interpret. Also, it is not uncommon for high-dimensional data to include unwanted correlations and irrelevances (Deisenroth et al. 2020, 286). Thus, mapping the initial data $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ to a lower-dimensional representation $\boldsymbol{Z} \in \mathbb{R}^{N \times M}$, where $M < D$ can minimize computation time and the risk of overfitting. Dimensionality reduction typically does not require labels and learns the optimal solution on $\boldsymbol{X}$ alone.

The learning component of ML can be viewed as a technique of pattern recognition by optimizing the parameters of a model. (Deisenroth et al. 2020, 3-4). The goal of learning is to adjust the model so that the optimal parameterizations are found which best fits the underlying data generating process. (Deisenroth et al. 2020, 230). The data on which parameters of the model are learned is called the training data, a subset of the available dataset $\mathcal{D}$. After training, models are usually validated on new data points from a validation dataset $\mathcal{D}_{\text{valid}} \subset \mathcal{D}$. Thus, learning is principally about finding suitable parameters using a training data $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ and then testing these parameterizations with a validation data such that $\mathcal{D}_{\text{valid}} \cap \mathcal{D}_{\text{train}} = \emptyset$. (Mohri et al. 2018, 6.) The validation helps to choose the better model which is usually ultimately tested on a final test dataset, not available when training (Mohri et al. 2018, 5; Deisenroth et al. 2020, 255), formally $\mathcal{D}_{\text{test}} \cap (\mathcal{D}_{\text{valid}} \cup \mathcal{D}_{\text{train}}) = \emptyset$, $\mathcal{D}_{\text{test}} \subset \mathcal{D}$.

Figure 2: Splitting the Data, Deisenroth et al. (2020, 255)

The rudimentary data split of 60% training data 20% validation data and 20% test data is visually illustrated above in Figure 2. Other more robust schemes, such as k-fold Cross-Validation (CV) or leave-one-out CV, are commonly used to evaluate and train models (see e.g. Mohri et al. 2018, 72). In k-fold CV, the same data is partitioned several times into folds of training and validation data. The model is retrained and evaluated for each fold. Usually, the data used in CV is not included in a final test dataset. (Arlot & Celisse 2010, 42, 53.) Overall, a common element of data splitting methods is that the training dataset, on which models learn, is the largest (Mohri et al. 2018, 71-72).

There are multiple learning paradigms of which the most common are supervised learning and unsupervised learning (Mohri et al. 2018, 7). The most common form of learning for making prediction models is supervised learning, in which a ML algorithm automatically modifies the model‘s internal parameters to match its predictions to the desired labels (LeCun et al. 2015, 436). Thus, supervised learning inherently relies on labeled datasets, that is to say $\boldsymbol{X}$ is not enough (Deisenroth et al. 2020, 228). In unsupervised learning, the learning is exclusively conducted on inputs $\boldsymbol{X}$ with labels excluded. Dimensionality reduction is an example of a task that typically employs this learning paradigm. (Mohri et al. 2018, 7.)

In practice there are three phases of learning: (1) prediction or inference, (2) parameter estimation, and 3) hyperparameter tuning or model selection. The prediction phase concerns making predictions on new data the model has not been trained on. Parameter estimation concerns adjusting the model based on the training data via, for example, empirical risk optimization or maximum likelihood estimation. (Deisenroth et al. 2020, 230-231.) The first two phases are typically automated in practice. Thus, the third phase requires more discretion from the ML practitioner. Hyperparameter tuning and model selection concerns modelling decisions concerning structure of the model, which are not automatically optimized in second phase (Deisenroth et al. 2020, 231). For instance, the hyperparameter of a basic linear regression model could be how many explanatory variables should be used to model the label i.e. the dependent variable. To complete the learning process, phases one and two are repeated for different models or sets of hyperparameters until an adequate model is found.

In literature, models and their predictions are conventionally represented as functions of the input data $\boldsymbol{X}$, and parameters $\boldsymbol{\theta}$ learned from this data (Deisenroth et al. 2020, 227-230, Botsch 2023, 6-11). For example, in a regression task, the data generating process f for label $\boldsymbol{y}$ can be described as,

$$f: \mathbb{R}^D \to \mathbb{R},\ \boldsymbol{x}_t \longmapsto y_t, \qquad \boldsymbol{x}_t \in \boldsymbol{X}_t,\ \text{and}\ y_t \in \boldsymbol{y}.$$

Thus, a trained model that makes a prediction for the label $\hat{y}_t$ with data $\boldsymbol{x}_t$ could be expressed as:

$$g: \mathbb{R}^D \to \mathbb{R}, \qquad \boldsymbol{x}_t \longmapsto \hat{y}_t.$$

In the case of regression, supervised learning is employed such that the quality of the model $g(\cdot)$ is assessed. Drawing on statistical decision theory, the lack of quality is measured instead of quality, so that a loss function $\ell(y_t, \hat{y}_t)$ assigns a cost to the predictions (Botsch 2023, 6-7). The loss function uses the ground truth label y_t and the prediction $\hat{y}_t$ as input, and outputs a positive cost value. The cost represents the error of the prediction (Deisenroth et al. 2020, 233). The quality of a model $g(\cdot)$ can then be proxied by the *prediction risk* $R(g)$, formalized in Equation 18 (Botsch 2023, 6),

$$R\big(g(\cdot;\boldsymbol{\theta})\big) = \mathbb{E}_{\boldsymbol{x},y}[\ell(y,\hat{y})] \geq 0. \tag{18}$$

Thus, for a continuous label one can state (Botsch 2023, 6-7),

$$\mathbb{E}_{\boldsymbol{X},Y}\big[\ell\big(Y, g(\boldsymbol{X};\boldsymbol{\theta})\big)\big] = \int_{\mathbb{R}^D}\int_{\mathbb{R}} \ell\big(y, g(\boldsymbol{x};\boldsymbol{\theta})\big) p_{\boldsymbol{X},Y}(\boldsymbol{x},y)\, dy d\boldsymbol{x}\,. \tag{19}$$

The prediction risk[6] is thus an aggregate measure of loss. It is measured for each prediction, weighted by the probability of their occurrence, and evaluated over all possible inputs and label outcomes. A theoretical interpretation of this is that a realized value of true prediction risk would require an infinite set of all possible data and labels (Deisenroth et al. 2020, 234). Therefore, in practice, prediction risk is impossible to compute, because the exact joint probability distribution $p_{\boldsymbol{X},Y}$ is unknown. Thus, one must rely on an empirical measure, for a given dataset Ɗ and model g,

$$R_{emp}(g; \mathbf{X}, \boldsymbol{y}) = \frac{1}{N}\sum_{t=1}^{N} \ell\big(y_t, g(\boldsymbol{x_t}; \boldsymbol{\theta})\big). \tag{20}$$

A model and its optimal parameters $\boldsymbol{\theta}^*$ are generally selected to minimize $R_{emp}(\cdot)$,[7] formally,

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\{R_{emp}(g)\}. \tag{21}$$

This is an example of a common approach to training or parameter estimation in ML called empirical risk minimization (Deisenroth 2020, 232-233). Other frameworks for parameter estimation include maximum-a-posteriori, maximum likelihood, and Bayes estimation (Botsch 2023,8-9; Deisenroth 2020, 230). The choice of a loss function is thus a crucial component of training. The most common loss function used for regression tasks is the squared error (Botsch 2023, 9-10).

$$\ell(y_t, \hat{y}_t) = (y_t - \hat{y}_t)^2. \tag{22}$$

Choosing squared error as the loss function leads to MSE for $R_{emp}(g, \mathbf{X}, \boldsymbol{y})$ in Equation 20,

$$R_{emp}(g; \mathbf{X}, \boldsymbol{y}) = \frac{1}{N}\sum_{t=1}^{N} (y_t - \hat{y}_t)^2 \equiv MSE(g; \mathbf{X}, \boldsymbol{y}).$$

This metric is often normalized to root mean square error (RMSE) for reporting purposes. RMSE allows errors to be interpreted with the same units as the label (Deisenroth 2020, 232). For example, if a forecast model fits the asking price of the EUR/USD currency pair, RMSE would also be measured in the USD price of EUR. Formally RMSE is simply defined as the square root of MSE. This normalization is analogous to the relationship between variance and standard deviation.

Moreover, a particularly useful quality of the squared error loss is that it is convex w.r.t to model predictions $\hat{y}_t$, and therefore has a single global minimum in the case of a linear model i.e. linear

6 Prediction risk is sometimes referred to as generalization risk or expected loss.
7 In the case of OLS regression, model parameters are computed by selecting $\ell(y_t, \hat{y}_t) = (y_t - \hat{y}_t)^2$.

regression (Cheridito et al. 2022, 63-64). However, for more complex ML models—such as ANNs—a distinct global minimum for the loss rarely exists.[8]

Consequently, when an analytical solution is infeasible, and a model $h(\boldsymbol{x_t}; \boldsymbol{w}^*)\colon \mathbb{R}^D \to \mathbb{R},\ \boldsymbol{x}_t \longmapsto \hat{y}_t$ is constructed as a differentiable function w.r.t its parameters, a *gradient descent* learning algorithm can be employed to find optimal weights $\boldsymbol{w}^*$. The algorithm first initializes the weights randomly or with a pre-defined expert guess as $\boldsymbol{w_0}$. Next, the algorithm calculates a gradient vector w.r.t loss. (Deisenroth 2020, 204.) The gradient vector is defined below, where in the first step $iter = 0$,

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_{iter}} = \sum_{(\boldsymbol{x_t}, y_t) \in \mathcal{D}_{\text{train}}} \frac{\partial}{\partial \boldsymbol{w}_{iter}} \ell\big(y_t, h(\boldsymbol{x_t}; \boldsymbol{w}_{iter})\big).$$

$\mathcal{L}$ denotes loss over instances of training dataset $\mathcal{D}_{\text{train}}$.[9] Now the gradient vector indicates the amount the loss would increase for a unit increase in that weight for each weight. Next, the gradient descent algorithm updates the weights $\boldsymbol{w}_1$ in the opposite direction to the gradient vector. The magnitude of this update step is determined by a hyperparameter called the learning rate, denoted with α. (LeCun et al 2015, 436-437; Deisenroth 2020, 204.) This update-step can be expressed as:

$$\boldsymbol{w}_{iter+1} = \boldsymbol{w}_{iter} - \alpha \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_{iter}}.$$

Gradient descent exploits the fact that the gradient of h w.r.t the total loss $\mathcal{L}$ decreases increasingly as steps are taken in the direction of the negative gradient. Thus, the total magnitude by the which the weights are updated decreases, and the algorithm converges towards the global minimum.[10] In practice, most modern algorithms use a more advanced class of such procedure called stochastic gradient decent for optimizing ANNs. (LeCun et al 2015, 436-437; Deisenroth 2020, 204.)

Another foundational concept of ML explorable via squared error is the *bias-variance* framework. By incorporating square error as the loss function in prediction risk, as formalized in Equation 22, the decomposition of error into its theoretical components can deliver insight into the well-established *bias-variance* framework (Botsch 2023, 10; Bishop 2006, 148-152). As the name implies, the *bias-variance* framework explains loss via its component pieces variance and bias.

$$\mathbb{E}_{x,y}\left[(y_t - \hat{y}_t)^2\right] = \mathbb{E}_x\left[\underbrace{\left(\mathbb{E}_{\mathcal{D}|x}[\, g(\boldsymbol{x_t}; \boldsymbol{\theta}|\mathcal{D})] - \mathbb{E}_{y|x}[\, f(\boldsymbol{x_t}; \boldsymbol{\theta}^{true})]\right)^2}_{Bias^2} + \underbrace{\mathbb{E}_{\mathcal{D}|x}\left[\left(g(\boldsymbol{x_t}; \boldsymbol{\theta}|\mathcal{D}) - \mathbb{E}_{\mathcal{D}|x}[\, g(\boldsymbol{x_t}; \boldsymbol{\theta}|\mathcal{D})]\right)^2\right]}_{Variance} + \underbrace{\varepsilon_t^2}_{Local\ Error}\right],$$

[8] The landscape of the loss becomes non-convex due to the activation functions of the hidden units.
[9] The gradient operation can be moved inside the summation, because of the summation differentiation rule.
[10] The convergence of gradient descent towards the global minimum depends highly on the learning rate.

where ε denotes the irreducible local error or noise.[11] Bias is the expected difference between the true model's prediction $f(\boldsymbol{x}_t; \boldsymbol{\theta}^{true})$ and the trained prediction $g(\boldsymbol{x}_t; \boldsymbol{\theta}|\mathcal{D})$ over dataset $\mathcal{D}$. Variance represents the expected deviations around the average prediction for the model g, given dataset $\mathcal{D}$, hence proxying the sensitivity of g to the data (Bishop 2006, 149).

In the *bias-variance* framework the problem of overfitting vs. underfitting is illustrated. Overfitting refers to the situation where one either over trains the model or creates an overtly flexible model. Overfit models tend to be very sensitive to deviations in the data, i.e. noise, but their estimates are generally centered around the true label, thus lowering bias (Deisenroth 2020, 243). In contrast, rigid models have low variance, but higher bias. In essence, an underfitting model is not *complex* or rich enough to capture relevant patterns in the data. This relationship between complexity and the bias-variance tradeoff is represented in Figure 3 below.

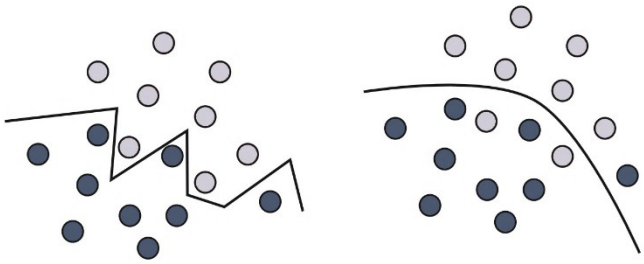

Figure 3: Bias-Variance Trade-Off (Mohri et al. 2018, 8)

Figure 3 illustrates a binary classification problem. The left-side of the image represents a more complex model and the right-side a less complex model (Mohri et al. 2018, 8). The shape of the classifier on the left is more complex, and it clearly fits the training data well, but the better model would predict well on unseen data, not just the training data. For a model performing better on validation data, one might assume that it has learned a more generalizable ruleset of the true process $\boldsymbol{x}_t \mapsto c_t$. Thus, if this process is complex, so should the model be too (Mohri et al. 2018, 7-9, 29-31.)

The trade-off for complexity is illustrated by the learning bound of a finite hypothesis set denoted $\mathcal{H}$. $\mathcal{H}$ is a construct depicting a set of functions h which map features to a label. In training, a selected algorithm, bound by hyperparameters and model structure, considers realizations of $h \in \mathcal{H}$, when learning to model the true mapping $\boldsymbol{x}_t \mapsto y_t$. For a finite $\mathcal{H}$, it can be shown that the upper bound of its true loss $R(h)$ depends on its empirical error $R_{emp}(h)$, which might be lowered by complexity, and on the size of the dataset N, but also on the complexity via number of unique models $|\mathcal{H}|$ in the hypothesis set. (Mohri et al. 5-10.) Formally, the upper bound of $R(h)$ with at least probability $1-\delta$,

[11] The local error term stems from the expected noisiness of the true model $\mathbb{E}_{\boldsymbol{x}}\left[\mathbb{E}_{y|\boldsymbol{x}}[y_t^2] - \mathbb{E}_{y|\boldsymbol{x}}[y_t]^2\right] = \mathbb{E}_{\boldsymbol{x}}[\,\varepsilon_t^2]$.

$$R(h) \leq R_{emp}(h) + \sqrt{\frac{\ln|\mathcal{H}| + \ln\frac{2}{\delta}}{2N}}. \tag{23}$$

The enclosing notation $|\cdot|$ denotes cardinality of the set where, $\mathcal{H} = \{h_1, \dots, h_{|\mathcal{H}|}\}$. Thus, Equation 23 suggests that a smaller $\mathcal{H}$ might help reduce expected error, depending on how the empirical error is affected. Therefore, trade-offs in controlling complexity are implied. (Mohri et al. 2018, 21.)

3.1.2 Classification and Model Selection

Algorithms designed for classification tasks assign a class label from a discrete set of options (Deisenroth 2020, 335). In practice, the aim is to associate instances with their true category. An ML algorithm that constructs or induces a model for a classification task, is called a classifier (Rokach 2010, 1). The simplest group of classification tasks is binary classification, where labels $\boldsymbol{c} \in \mathcal{Y}^N$ can realize as one of two discrete outcomes. In context of this thesis, buy or sell. In literature, the binary label space is commonly denoted as $\mathcal{Y} = \{1,0\}$.[12] (Deisenroth 2020, 335; Mohri et al. 2018, 30).

For binary classification, the relationship between model complexity and expected loss $R(h)$ of an infinite hypothesis set $\mathcal{H}$,[13] can be discussed via the growth function bound. With the probability of at least $1 - \delta$ (Mohri et al. 2018, 35) one can assert the following inequality,[14]

$$R(h) \leq R_{emp}(h) + \sqrt{\frac{2\ln\Pi_{\mathcal{H}}(N)}{2N}} + \sqrt{\frac{\ln\frac{1}{\delta}}{2N}}, \tag{24}$$

where $\Pi_{\mathcal{H}}(N)$ denotes the growth function, i.e. maximum number of unique ways in which N datapoints could be classified using a hypothesis set $\mathcal{H}$. Thus, it could be viewed that flexibility for splitting the dataset, to an extent might increase the prediction risk. The true exploration of this bound becomes very difficult as the complexity of $\mathcal{H}$ grows because it requires computing $\Pi_{\mathcal{H}}(N)$. Nevertheless, the inequalities presented in Equations 23 and 24 aid to illustrate how complexity, which might help to induce generalizable mappings for complex data, can also lead to unwanted outcomes where random noise is captured. (Mohri et al. 2018, 7-9, 29-36; Foster et al. 2020, 2-4.)[15]

---

[12] This could be viewed as a depreciation or appreciation of USD with respect to the Euro.
[13] Sometimes the hypothesis set is equated to a family of functions. (See e.g. Mohri et al. 2018, 10).
[14] Equation 24 stems from Rademacher complexity.
[15] It is common to extend this discussion with the bias-variance trade off illustrated with the U-shaped curve, but considering modern ML practices (Belkin et al. 2019; Nakkiran et al. 2019), this was omitted from the final thesis.

Given the discreate nature of a classification task, the prediction loss (formalized in Equations 18 and 19) for a multiclass classifier $\boldsymbol{g}(\boldsymbol{x}_t;\boldsymbol{\theta})$: $\mathbb{R}^D \rightarrow \mathrm{y}^K$, $\boldsymbol{x}_t \longmapsto \hat{\boldsymbol{c}}_t$, can be expressed as,

$$\mathbb{E}_{\boldsymbol{x},y}\big[\ell\big(\boldsymbol{c},\boldsymbol{g}(\boldsymbol{x};\boldsymbol{\theta})\big)\big] = \sum_{k=1}^{K}\sum_{j=1}^{K}\int_{\mathbb{R}^D}\ell\big(c_{k,t},\hat{c}_{j,t}\big)p_{\boldsymbol{\chi}}(\boldsymbol{\chi}=\boldsymbol{x},y=c_k)d\boldsymbol{x},$$

where, K represents the number of possible classes the true label $c_{k,t}$ and prediction $\hat{c}_{j,t}$ can be assigned to. $\boldsymbol{g}(\boldsymbol{x}_t;\boldsymbol{\theta})$ outputs a vector over possible $c_{k,t}$. For an output space with this characteristic, the loss function discussed in 3.1.2, in theory, could be formalized as (Botsch 2023, 7-10),

$$\ell\big(c_{k,t},\hat{c}_{j,t}\big) = 1-\delta\big(c_{k,t},\hat{c}_{j,t}\big).$$

Function $\delta(\cdot,\cdot)$ compares inputs, and outputs a 1 if they are the same and 0 otherwise (Botsch 2023, 7). The empirical prediction risk implementing this 0/1-loss function becomes,

$$R_{emp}(\boldsymbol{g};\mathbf{X},\boldsymbol{c}) = \frac{1}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\Big(1-\delta\big(c_t,\hat{c}_{k,t}\big)\Big).$$

Thus, classification training involves finding the optimal hypothesis $\boldsymbol{g}^*$ from its hypothesis set $\mathcal{G}$, or in practice, training the optimal model $\boldsymbol{g}(\boldsymbol{X},\boldsymbol{c};\boldsymbol{\theta}^*)$ in a way that minimizes the empirical prediction risk as shown in Equation 21. Following the logic of the bias-variance framework, the trained model $\boldsymbol{g}(\boldsymbol{c};\boldsymbol{\theta}^*)$ should be validated against a dataset previously unseen to the model to test how well it generalizes. As highlighted by the inequalities presented in equations 23 and 24, the choice of the hypothesis set directly prediction risk and thus generalizability meaningfully. Thus, choice of hyperparameters and the ML algorithm is key (see e.g. Shawi et al. 2025).

To diagnose the performance of a classification model and thus make an informed decision on selection the ML practitioner should rely on a set of diagnostic tools. In the context of binary classification tools such as the *confusion matrix* and the *Receiver Operator Characteristic* (ROC) are typically applied in conjunction with metrics such as *accuracy*, *precision* and *recall*, to assist in model selection. Accuracy alone is often a poor metric alone for measuring classifiers' performance (Fawcett 2006, 861). A more detailed assessment of the misclassification patterns is obtained via the confusion matrix. Provided the true labels are known, there are four outcomes to binary classification,

true positives TP, true negatives TN, but also false positives FP, and false negatives FN. These outcomes are typically represented on a confusion matrix, as illustrated in Figure 4 below.

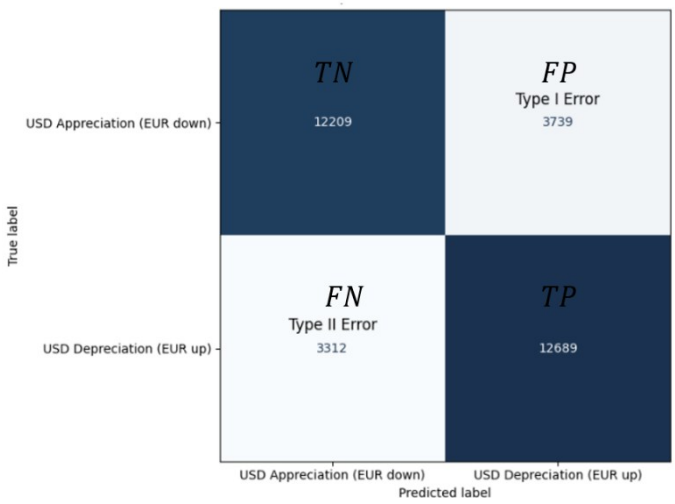

Figure 4: Confusion Matrix Illustration

The confusion matrix as represented in Figure 4, depicts outcomes for binary classification in a two-by-two matrix. The columns depict the model's predicted direction (up or down), and the rows portray the actual realized directions of EUR/USD. Here, the model appears to perform well with an accuracy of 77.93% on the testing data.[16] In addition to accuracy, two crucial metrics can be determined from the matrix, the *true positive rate* and *true negatives rate*, referred to as recall (Fawcett 2006, 862). Formally recall is expressed as,

$$Recall = \frac{Correctly\ assigned\ relevant\ instances}{All\ relevant\ instances}.$$

For example, when trying to predict a depreciation (EUR/USD increase), the model in Figure 4 correctly assigned 12'689 instances out of all 16'001 instances of actual depreciation. Thus, here the recall rate was 79.30%, equally for appreciation, recall was 76.56%. Another important metric is precision, which expresses how often the predictions for a class were correct, formally,

$$Precision = \frac{Correctly\ assigned\ relevant\ instances}{All\ relevant\ instances}.$$

In this example the model's precision was 77.24% and 78.66% for depreciation and appreciation outcomes respectively.

[16] Accuracy is defined as, $Accuracy = (TP + TN) / (TP + TN + FP + FN)$ (Fawcett 2006, 862).

A common metric to gauge model performance via precision and recall simultaneously, is the F1-score. The F1-score is a harmonic mean of precision and recall formalized as,

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}.$$

Furthermore, another important diagnostic tool for binary classification is ROC. ROC is usually represented graphically in a two-dimensional ROC space, defined by a true positive rate y-axis, and a false positive rate x-axis. A discrete binary classifier produces a model represented by a point with coordinates (false positive rate, true positive rate) in the ROC space. Thus, point (0,0) represents models that assign a negative classification for all instances, conversely point (1,1) represents models assigning a positive classification for all instances. The point (0,1) represents perfect classification, where the true positives rate is 100% and false positive rate is 0%. Splitting this space is a diagonal line depicting a random classifier which has a false positive rate equaling true positive rate. (Fawcett 2006, 861-863.)

Typically, binary classifier models output a prediction score $\hat{p}_t \in [0,1]$, and based on this output a positive class {+1} is assigned when the score meets or surpasses a threshold value. Formally,

$$\hat{c}_t = \begin{cases} +1, & if\ \hat{p}_t > threshold, \\ -1, & otherwise. \end{cases}$$

On a ROC curve, the model's predictions are computed with various classification thresholds. For each threshold, the false positive rate and true positive rate are computed typically using test or validation instances. Plotting and connecting these points produces a ROC curve. (Fawcett 2006, 861-863.) Figure 5 below illustrates a ROC curve used to diagnose the same model as in Figure 4.

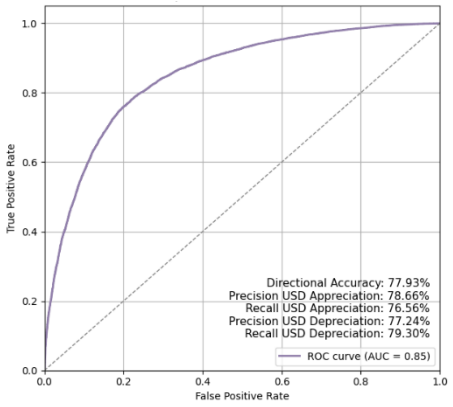

Figure 5: ROC Curve Illustration

In Figure 5, the ROC curve is consistently above the diagonal. Therefore, the model achieves more true positives per each false positive compared to what random guessing should yield with different

configurations of the threshold value. To concisely convey comparative performance, an *Area Under the Curve* (AUC) metric is used. AUC is simply the area of unit square space under the *ROC curve*.[17] AUC is a threshold independent measure, which decouples diagnosis from class skew and error cost issues, contributing to its viability. (Fawcett 2006, 868-873.) Thus, AUC and ROC curve are potent tools for selecting and evaluating the ML models for classification tasks.

In addition to the NLP models explored in 2.2, three families of ML models are employed in this thesis, 1) Random Forest (RF) classifiers, 2) logistic regression and 3) select types of ANNs. RF is a prevalent ensemble model developed by Leo Breiman (2001) in his seminal work at the turn of the century. RF has seen application in virtually every discipline. Ensemble learning is a ML paradigm that integrates multiple models simultaneously to solve the same problem (Rokach 2010, 2). Typically, a collection of individual *weak* classifiers[18] hold a vote on how the instances they are presented with should be classified, and the most popular class is then selected for each instance. (Fawagreh et al. 2014, 602-604, 607-608.) Formally for a classification task, with classes $k \in K$,

$$\hat{c}_t = \underset{k \in K}{\operatorname{argmax}} \sum_{m=1}^{M} \hat{\psi}_{m,k}(\boldsymbol{x}_t)\,.$$

A RF model uses bootstrap aggregation, or bagging, to train the ensemble RF of decision trees ψ_m it employs as its weak classifiers (Breiman, 2001, 5). Each tree $\psi_m \in RF$ is trained on a bootstrapped dataset S_m of size N_B sampled with replacement from the original training dataset $\mathcal{D}_{train}$ (Rokach 2010, 11-13). Pseudo code for creating such an ensemble is depicted in Figure 6.

Algorithm: Simple Random Forest Algorithm

Input : Training data: $\mathcal{D}_{train} = \{\mathbf{X}, \mathbf{c}\}$
Base *weak* learner: Adapted_CART_Classifier(·)
Number of trees: M

Output: Random Forest model: $RF(\cdot)$

Initialize $m \leftarrow 0$
while $m < M$ **do**
  Sample N_B instances with replacement from $\mathcal{D}_{train}$ to form S_m
  Train a tree $\psi_m \leftarrow$ Adapted_CART_Classifier(S_m)
  $m \leftarrow m + 1$
end
return $RF(\cdot) = \{\psi_1, \psi_2, \ldots, \psi_M\}$

Figure 6: The Random Forest Algorithm (Adapted from Rokach 2010, 10-13)

[17] Interestingly, AUC is closely related to the Gini coefficient, $AUC = \frac{Gini+1}{2}$ (Fawcett 2006, 868).
[18] Weak classifiers are simple classifiers that perform slightly better than a random guess would (Rokach 2010, 3).

The decision trees themselves are induced using an adaptation of the *classification and regression trees* (CART) technique. CART is adapted such that at each split, rather than evaluate the node over all features *D*, a randomized subset of the original full feature set—usually of size $\sqrt{D}$—is used.[19] These decision trees are grown until they reach a predetermined maximum size or some other stopping criteria[20] have been met. (Breiman 2001, 11-13, 29-30; Rokach 2010, 12-13; Louppe 2015, 29-31.) The major advantage of RF is that it is fast, robust to outliers, and can handle datasets with many features (Rokach 2010, 13; Louppe 2015, 26). In practice, RF sees a lot of use in feature selection (see e.g. Ding et al. 2024; Botsch 2023, 45). (Fawagreh et al. 2014, 603-608.)

Logistic regression has been widely applied in corporate finance, banking, and investment (Kelany et al. 2020, 121) contexts. For binary classification, logistic regression can be expressed as a sigmoid *activation function* $\sigma(\cdot)$ applied to a linear function z_t of a feature vector $\boldsymbol{x}_t$ to predict the probability of instance t belonging to class c_t (Bishop 2006, 114, 205; Deisenroth 284; Aggarwal 2023, 21),

$$\hat{p}(c_t|\boldsymbol{x}_t) = \sigma(z_t) \in [0,1], \qquad z_t = b + \boldsymbol{w}'\boldsymbol{x}_t \in \mathbb{R}, \qquad \sigma(z_t) = \left(1 + exp(-z_t)\right)^{-1} \in [0,1].$$

The strength of logistic regression is that it is less sensitive to non-normal data (Kelany et al. 2020, 121), and that it operates with a compact set of parameters, one for each of its *D*-features. (Bishop 2006, 205). For multiclass classification, the prediction $\hat{\boldsymbol{p}}$ is a vector, where each element $\hat{p}_k$ corresponds to a class c_k defined as part of the classification task. To map predictions $\hat{p}_k$ for each class, the sigmoid activation function is essentially swapped out for *softmax function* $\boldsymbol{\sigma}(\cdot)$, which normalizes the outputs (Aggarwal 2023, 23; Bishop 2006, 209), this can be expressed as,

$$\hat{\boldsymbol{p}}(\boldsymbol{c}|\boldsymbol{x}_t) = \boldsymbol{\sigma}(\boldsymbol{z_t}), \;\; \boldsymbol{z_t} = \left(z_{1,t}, \dots, z_{k,t} \dots, z_{K,t}\right) \in [0,1]^K,$$

$$z_{k,t} = b_k + \boldsymbol{w}_k'\boldsymbol{x}_t \in \mathbb{R}, \;\; \boldsymbol{\sigma}(\boldsymbol{z_t})_k = \frac{\exp\left(z_{k,t}\right)}{\Sigma_j \exp\left(z_{j,t}\right)} \in [0,1]^K.$$

Therefore, the logistic regression model becomes increasingly more complex w.r.t the feature dimension of the input $\boldsymbol{X}$, as for each class there is an individual linear function mapping inputs to the activation. Also, the constants or bias terms b could be constructed by setting one of the inputs to a constant value of 1 for each instance instead. To find the optimal solutions for the model parameters, maximum likelihood estimation is commonly used (see e.g. Ommen et al. 2011, 99-100).

---

[19] This randomized feature subset size restriction is common for classification tasks.

[20] Common stopping criteria include a predefined maximum tree depth (number of sequential nodes), a minimum number of training instances required to perform a split, and a minimum number of instances required to for a leaf node.

3.1.3 Three Flavors of Artificial Neural Networks: *Vanilla, LSTM, and CNN*

ANNs have become a success story of ML in the age of big data. Growth of computational power has allowed for the training of immensely complex models, which are able to generalize on immense datasets. Deep learning ANNs are unique in their ability to keep increasing in accuracy when the amount of training data increases. Explicitly, ANN algorithms manage to keep improving in accuracy further than traditional ML methods generally would with complex data (Aggarwal 2023, 16.) Additionally, ANNs are claimed to be *universal approximators* (Bishop 2006, 230; Aggarwal 2023 25-28), meaning that with a sufficiently complex model algorithm, theoretically, it is possible for the ANN model to represent any continuous mapping if the necessary data and compute are provided.

For the purposes of this thesis, the ANNs considered will be *feedforward neural networks*, also known as multilayer perceptrons. The *architecture*, or blueprint of the neural network, is that of interconnected nodes, where model inputs are fed forward through layers of nodes to an eventual output layer. Every node of the network holds a variable that is the result of a previous computation or the original input values in the case of the input nodes. The simplest such network is the perceptron or a single layer neural network. A perceptron is quite similar to a logistic regression model, except that its output layer's activation function $\text{sign}(\cdot)$ outputs in $\{+1, -1\}$ based on linear function output. (Aggarwal 2023, 16-19.) A perceptron's label prediction can be expressed formally as,

$$\hat{c}_t = \text{sign}(b + \boldsymbol{w}'\boldsymbol{x}_t) = f(\boldsymbol{x}_t) \in \{+1, -1\}.$$

Indeed, a multilayer network is simply a *chain* of linear mappings and different activation functions. For a L-layer deep multiclass classification task this can be expressed as (Deisenroth 2020, 284),

$$\hat{\boldsymbol{p}}_t = \boldsymbol{\sigma} \circ \boldsymbol{h}_{L-1} \circ \ldots \circ \boldsymbol{h}_1 \in \mathrm{y}^K,$$

where each hidden layer output is defined as (LeCun et al. 2015, 437; Aggarwal 2023, 23-27),

$$\boldsymbol{h}_l = \boldsymbol{g}_l\left(\boldsymbol{W}_{l-1}^{(l)}\boldsymbol{h}_{l-1} + \boldsymbol{b}_{l-1}^{(l)}\right), \qquad \boldsymbol{h}_0 = \boldsymbol{x}_t, \tag{25}$$

where, activation functions at the hidden layers are denoted $\boldsymbol{g}_l(\cdot)$, with $l = 1, \ldots, l \ldots, L-1$ indexing the layers of the network. The final softmax activation is represented with $\boldsymbol{\sigma}(\cdot)$, producing K-dimensional outputs for each category K where $\mathrm{y}^K = [0,1]^K$. Also, $\boldsymbol{W}_{\boldsymbol{l-1}}^{(\boldsymbol{l})}$ and $\boldsymbol{b}_l$ denote the weight matrix and vector bias terms that map outputs $\boldsymbol{h}_{l-1}$ from layer $l-1$ to layer l. The choice of activation function is a key consideration in constructing hidden layers of an ANN. As long as they are nonlinear, the model can be considered to be different from single layer linear model (Aggarwal 2023,

21-23). Pertinently, ANNs are known to perform well with time series data given their ability to learn non-linear patterns and efficiency even within noisy data (Safi et al. 2020, 485).

In practice, for typical ANN[21] classifiers output of the model is often a nonlinearly processed prediction $\hat{\boldsymbol{p}}_t \in [0,1]^{(K)}$. Each element $\hat{p}_{t,j}$ can be thought of as a proxy for how certain the model is of the classification for label $j \in 1, \dots, K$ (Deisenroth 2020, 356). Furthermore, in some applications, predictions $\hat{p}_{t,j}$ are directly mapped so that their sum over instances *t* is equal to one, such as with the sigmoid function $\sigma(\cdot)$, lending outcomes of $\hat{p}_{t,j}$ to probabilistic interpretation (Deisenroth 2020, 336; Aggarwal 2023, 21-23). To pragmatically account for such outputs, the classification loss function is commonly expressed as cross-entropy loss (Aggarwal 2023, 23-25, 83-84) formalized as,

$$\ell(c_t, \hat{\boldsymbol{p}}_t) = -\sum_{k=1}^{K} p_{\mathrm{k,t}} \ln\left(\hat{p}_{k,t}\right) = -\ln\left(\hat{p}_{c_t,t}\right).^{22} \tag{26}$$

The true target value $p_{\mathrm{k,t}}$ takes a value of 1 for true class c_t and 0 for the rest. Also, $\hat{p}_t \in [0,1]$, and thus, loss is always positive and at least zero. For binary classification, $\hat{p}_t$ is a scalar output, with a realization of 1 representing full confidence in class $\{+1\}$, and 0 representing full confidence in class $\{-1\}$. Thus, when implementing Equation 26 into the empirical prediction risk function, loss is incurred by weak certainty for the correct label (Aggarwal 2023, 23, 52, Bishop 2006, 210.) The binary cross-entropy loss function is simply,

$$\ell(c_t, \hat{\boldsymbol{p}}_t) = -\big(p_t \ln(\hat{p}_t) + (1 - p_t)\ln(1 - \hat{p}_t)\big).$$

A useful feature of cross-entropy, like MSE, is that it is differentiable. In fact, ANNs are typically designed modularly, so that it is possible to compute the gradient of R_{emp} w.r.t model parameters in sequence. This allows for the use of the *backpropagation* algorithm.[23] In this modular setting, backpropagation is a practical application of the chain rule for derivatives. The differentiation process tracks how every node affects the empirical error starting at the output layer. By starting at the output layer and defining each layer's gradient's step-by-step as with the previous layers gradients, backpropagation efficiently defines the gradient vector in a recursive manner. (LeCun et al. 2015,

---

[21] Common classification ANNs refer to those using a softmax or sigmoid activation function at the output layer.

[22] Cross-entropy loss is equivalent to the negative log-likelihood loss. The likelihood function could be defined as,

$$p\big(\boldsymbol{c} = \hat{\boldsymbol{c}}|\hat{\boldsymbol{\theta}}\big) = \Pi_{t=1}^{N}\Pi_{k=1}^{K}\hat{p}_{k,t}^{\,p_{k,t}} \Rightarrow apply\ logarithm\ and\ negate \Rightarrow -ln\left(p\big(\boldsymbol{c} = \hat{\boldsymbol{c}}|\hat{\boldsymbol{\theta}}\big)\right) = -\sum_{t=1}^{N}\sum_{k=1}^{K} p_{k,t}\, ln\big(\hat{p}_{k,t}\big).$$

[23] Interestingly, some of the foundations for backpropagation, i.e. *reverse mode of automatic differentiation*, were conceptualized by a Finnish Master's Student Linnainmaa in 1970 (Schmidhuber 2015, 11; Griewank 2012, 389).

438; Aggarwal 2023, 41-52; Deisenroth et al. 2020, 138-143.) This process for a regression ANN model is illustrated in Appendix 2. The backpropagation algorithm is argued to be one of the major reasons for the revitalized interest in ANNs sparked during the 1980's (Aggarwal 2023, 38).

A prevalent type of ANN for time series data is the Long Short-Term Memory (LSTM). The LSTM was designed on the Recurrent Neural Network (RNN) architecture (Hochreiter & Schmidhuber 1997). Both RNNs and LSTMs were designed to model real data sequences, such as time series (Graves 2014, 1; Aggarwal 2023, 233, 247-249). With an RNN, a binary classifier prediction $\hat{p}_t$ is computed with recurrent dependency. To retain information from past inputs $\boldsymbol{x}_{t-1}$, and values of previous hidden vector sequences $\left\{\overline{\boldsymbol{h}}_{l,t-1} \middle| \, l \in \{1, \dots L-1\}, \; t \in \{1, \dots, N\}\right\}$ collected before the final activation at the output layer are saved by the RNN. The computations are done over sequenced instances $t \in (1,2, \dots, N)$, such that (Graves 2014, 4; Aggarwal 2023, 234-236),

$$\begin{aligned}
\overline{\boldsymbol{h}}_{1,t} &= \boldsymbol{g}_1\left(\boldsymbol{W}_0^{(1)}\boldsymbol{x}_t + \boldsymbol{W}_{h_1}^{(1)}\overline{\boldsymbol{h}}_{1,t-1} + \boldsymbol{b}_0^{(1)}\right), \\
\overline{\boldsymbol{h}}_{l,t} &= \boldsymbol{g}_l\left(\boldsymbol{W}_0^{(l)}\boldsymbol{x}_t + \boldsymbol{W}_{l-1}^{(l)}\overline{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l}^{(l)}\overline{\boldsymbol{h}}_{l,t-1} + \boldsymbol{b}_{l-1}^{(l)}\right), \\
z_{L,t} &= \boldsymbol{w}_{L-1}^{(L)}\overline{\boldsymbol{h}}_{L-1,t} + b_{L-1}^{(L)}, \\
\hat{p}_t &= \sigma\left(z_{L,t}\right).
\end{aligned}$$

In contrast to the structure presented in Equation 25, $\boldsymbol{W}_0^{(l)}$, $\boldsymbol{W}_{l-1}^{(l)}$, and $\boldsymbol{W}_{h_l}^{(l)}$ are weights applied to the input layer, the lower hidden layer, and the previous hidden vector state, respectively (Graves 2014, 4). In training, RNNs typically employ Backpropagation Through Time (BPTT). In this method, the algorithm evaluates loss over instances t, such that loss function $\boldsymbol{\mathcal{L}}(\boldsymbol{p}, \hat{\boldsymbol{p}}) = \Sigma_{t=1}^{T}\ell\left(\boldsymbol{p}_t, \hat{\boldsymbol{p}}_t\right)$ is minimized. Effectively, BPTT adjusts the weights using derivatives of the computed parameters in the network's layers, like in regular backpropagation (see Appendix 2), except with local gradients δ_t^l propagated in a recursive and potentially volatile manner. This can lead to issues with so-called vanishing and exploding gradients for increasing time spans (see e.g. Bengio et al. 1994). (Aggarwal 2023, 243.)

To solve this problem, Hochreiter & Schmidhuber introduced LSTM architecture in 1997. The novelty of the LSTM lies in its *memory cells* that store information and can be updated by using BPTT. With this structure the LSTM is able to model temporal dependencies over relatively large time spans. The memory cells of a LSTM store temporal states values which can be altered with new information and partially reset depending on the circumstances. In an LSTM layer, activation functions are implemented as a composite function constructed by an *input gate* i_t, *forget gate* f_t, *cell state* c_t, and *output gate* o_t. (Graves 2014, 5; Aggarwal 2023, 248-249.)

$$\boldsymbol{i}_t^{(l)} = \boldsymbol{\sigma}\left(\boldsymbol{W}_{l-1,i}^{(l)}\tilde{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l,i}^{(l)}\tilde{\boldsymbol{h}}_{l,t-1} + \boldsymbol{W}_{c,i}^{(l)}\boldsymbol{c}_{t-1} + \boldsymbol{b}_{l-1,i}^{(l)}\right)$$

$$\boldsymbol{f}_t^{(l)} = \boldsymbol{\sigma}\left(\boldsymbol{W}_{l-1,f}^{(l)}\tilde{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l,f}^{(l)}\tilde{\boldsymbol{h}}_{l,t-1} + \boldsymbol{W}_{c,f}^{(l)}\boldsymbol{c}_{t-1} + \boldsymbol{b}_{l-1,f}^{(l)}\right),$$

$$\boldsymbol{o}_t^{(l)} = \boldsymbol{\sigma}\left(\boldsymbol{W}_{l-1,o}^{(l)}\tilde{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l,o}^{(l)}\tilde{\boldsymbol{h}}_{l,t-1} + \boldsymbol{W}_{c,o}^{(l)}\boldsymbol{c}_{t-1} + \boldsymbol{b}_{l-1,o}^{(l)}\right),$$

$$\boldsymbol{c}_t^{(l)} = \boldsymbol{f}_t^{(l)} \otimes \boldsymbol{c}_{t-1}^{(l)} + \boldsymbol{i}_t^{(l)} \otimes \tanh\left(\boldsymbol{W}_{l-1,c}^{(l)}\tilde{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l,c}^{(l)}\tilde{\boldsymbol{h}}_{l,t-1} + \boldsymbol{b}_{l-1,c}^{(l)}\right),$$

$$\tilde{\boldsymbol{h}}_{l,t} = \boldsymbol{o}_t^{(l)} \otimes \tanh\left(\boldsymbol{c}_t^{(l)}\right), \qquad \tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}} \in (-1,1).$$

Here, $\otimes$ denotes an element-wise multiplication. The hidden states at the l:th layer are denoted $\tilde{\boldsymbol{h}}_{l,t}$, with $\tilde{\boldsymbol{h}}_{0,t} = \boldsymbol{x}_t$.[24] Below, Figure 7 illustrates the propagation of inputs at the first hidden layer $\tilde{\boldsymbol{h}}_{1,t}$.

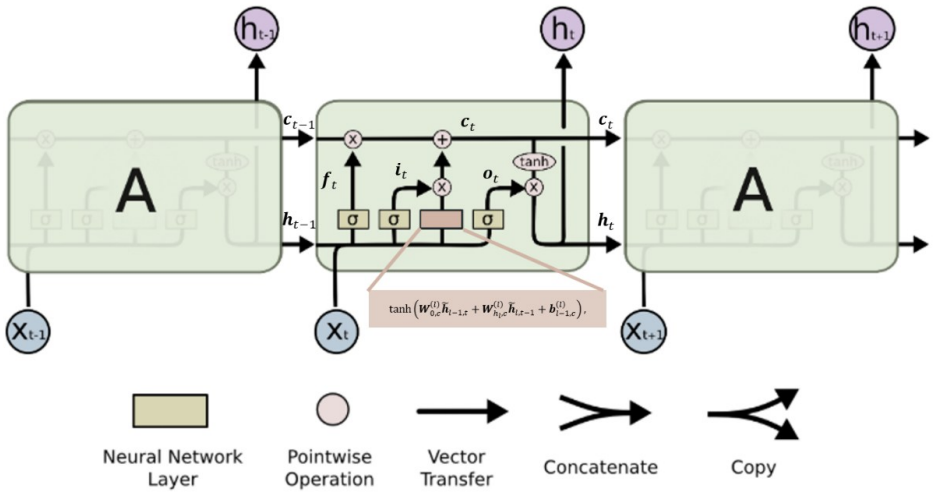

Figure 7: Structure of a LSTM Hidden Layer (Adapted from Colah 2015)

As with vanilla RNNs, LSTMs have a chainlike structure, but instead of having a single activation function for a layer, there are four, all interacting as a composite function. The cell state $\boldsymbol{c}_t^{(l)}$ stores information on previous model structures. *Outdated* information is removed from the cell state by the forget gate $\boldsymbol{f}_t^{(l)}$. New information is updated by incrementation via the input gate $\boldsymbol{i}_t^{(l)}$ by a *proposed* update denoted: $\tanh\left(\boldsymbol{W}_{0,c}^{(l)}\tilde{\boldsymbol{h}}_{l-1,t} + \boldsymbol{W}_{h_l,c}^{(l)}\tilde{\boldsymbol{h}}_{l,t-1} + \boldsymbol{b}_{l-1,c}^{(l)}\right)$. Finally, the hidden state vector $\tilde{\boldsymbol{h}}_{l,t}$ is produced by scaling output $\boldsymbol{o}_t^{(l)}$, element-by-element, with the tanh-transformed updated cell state $\boldsymbol{c}_t^{(l)}$. By implementing this structure, LSTMs can retain information over long time frames and lags robustly. (Aggarwal 2023, 249; Safi et al. 2022, 486.)

[24] The gate equations depict a LSTM with peephole connections $\boldsymbol{W}_{c,gate}^{(l)}$, generally these are not implemented as often.

Another prominent archetype of the ANN algorithm family is the CNN. CNNs are frequently associated with image classification tasks (Taye 2023, 4; LeCun et al. 2015, 440) and have seen application within time series regression and classification tasks. CNN models typically has the following structure: (1) a convolutional layer applied to the input tensor, (2) a pooling layer to condense outputs of the convolution layer, and (3) a so called fully connected layer, or a vanilla ANN -like set of hidden layers (Safi et al 2020, 485).

At the convolutional layers, a CNN learns a hierarchy of features from its input by constructing feature maps -iteratively convolving its input with learnable filters[25] (Taye 2023, 1, 5). These feature maps are passed down as inputs to following layers of network. In essence with two back-to-back CNN layers, the subsequent layer inherits the dimensions of the previous layer's output and process small local regions of it to produce new feature maps. The feature maps have a 3-dimensional grid structure, with height, width, and depth. Here, depth refers to the number of channels (feature maps) produced by the layer. The convolutional layer employs a predefined number of filters with a set spatial size, that is, a filter height and width determined as preselected hyperparameters, usually so that the filter has a square shape. Each filter is convolved across the inputs' spatial shape to produce a new feature map. The convolution operation is conceptually[26] a serial process of summing element-wise products computed by filters to produce output feature maps. The output of a convolutional layer is simply a stack of these feature maps. (Aggarwal 2023, 271-273.)

For time series prediction, one-dimensional CNNs (1D-CNN) have shown strong predictive ability (Zhang et al. 2023). In practice, the 1D filter has a height of 1 and a set kernel size of $\mathrm{K_{size}}$ with which the filter performs the convolutions with. Formally the 1D convolution can be expressed as,

$$\boldsymbol{o}_l^{(c_l)} = \sum_{c_{l-1}=1}^{C_{l-1}} \boldsymbol{w}_{l-1,c_{l-1}}^{(l,c_l)} \circledast \boldsymbol{o}_{l-1}^{(c_{l-1})} + \boldsymbol{b}_l^{(c_l)}. \tag{27}$$

Here, $\boldsymbol{o}_l^{(c_l)} \in \mathbb{R}^{width_l}$ denotes the output vector on layer l for feature map, or channel c_l, so that the layer's output is denoted $\boldsymbol{O}_l \in \mathbb{R}^{width_l \times C_l}$, where $\boldsymbol{O}_0 = \boldsymbol{x}_t$, and C_l represents the depth at layer l. Also, $\boldsymbol{b}_l^{c_l}$ represents the bias term and $\boldsymbol{w}_{l-1,c_{l-1}}^{(l,c_l)} \in \mathbb{R}^{\mathrm{K_{size}}}$ is the filter applied per input-output pair $\{c_{l-1}, c_l\}$, with $\circledast$ denoting the convolution operation. The complete set of filters for layer l is denoted as

[25] Filters are also called kernels for CNNs (Aggarwal 2023, 271-273; Taye 2023, 5).
[26] In practice these operations are optimized with parallelized computed and thus the procedure is not actually sequential (Taye 2023, 5,16).

$\boldsymbol{W}_{l-1}^{(l)} \in \mathbb{R}^{C_l \times C_{l-1} \times K_{size}}$ (Zhang et al. 2023, 3353.)[27] At the input layer, width can be used to capture several lagged instances to be considered at time t for producing a prediction. Thus, when considering Equation 27 at the input layer of a 1D-CNN implementing an input vector $\boldsymbol{x}_t$, one could state that,

$$o_{1,j}^{(c_1)} = \sum_{i=0}^{k_{size}-1} \boldsymbol{w}_0^{(c_1)}[i]\, x_{j+i} + b_1^{(c_1)} \quad \forall\, j = T - (k_{size} - 1), \dots, 0.$$

$\boldsymbol{w}_0^{(c_1)}[i]$ denotes the i:th element of filter $\boldsymbol{w}_0^{(c_1)}$, considering $k_{size} - 1$ values per convolution. The index j denotes *time position* on the feature map $\boldsymbol{o}_1^{(c_1)}$ to which output value $o_{1,t}^{(c_1)}$ is appended.

Because the same filters are passed multiple times over their input feature maps, a CNN can be thought of as network that employs multiple copies of the same neuron on its inputs. With this perspective, a CNN can be thought to represent large and complex processes with a relatively concise set of parameters. A 1D-convolution at the input layer is represented below in Figure 8.

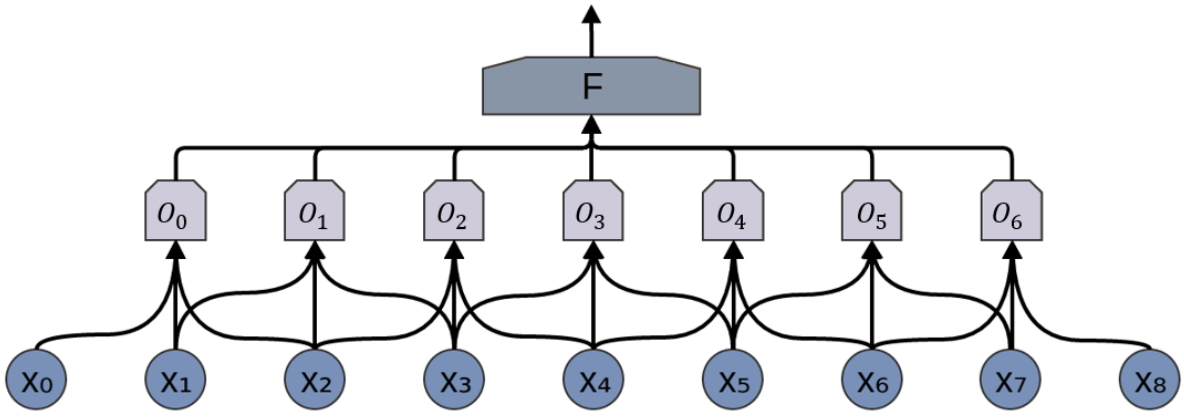

Figure 8: One-Dimensional Convolution Layer (Adapted from Colah 2014)

Here, a filter $\boldsymbol{w_0^1}$ has a height and depth of 1, with a width of 3, i.e., $k_{size} = 3$. Input $\boldsymbol{x}$ has 9 sequential instances, thus, input width T is 9. At the first step, a dot-product operation is performed on the newest inputs $x_6, x_7,$ and x_8, which is then appended to the output vector to produce o_6. The filter then proceeds to the next set of previous instances and repeats the procedure on inputs $x_5, x_6,$ and x_7, continuing to do, while appending values to output vector $\boldsymbol{o}^{(1)}$ until it is of width: $T - (k_{size} - 1)$.[28] In this example the final output width of $\boldsymbol{o}^{(1)}$ is 7.

---

[27] Per feature map at layer l, the set of filters can be denoted $\boldsymbol{W}_{l-1}^{(l,c_l)} \in \mathbb{R}^{C_{l-1} \times K_{size}}$.

[28] In addition to filter size, padding and stride affects the output's width, in practice (Taye 2023, 7-9).

Other common key components of CNN architectures are pooling layers. Pooling helps reduce dimensionality of the input for consecutive layers. A pooling layer aggregates values from predefined windows of the feature map. However, unlike in the convolution layer, the pooling layer *down samples* the feature map by portioning it into non-overlapping sub-grids of predefined size. The aim is to reduce the dimensionality while retaining most of the useful information, simultaneously cutting-down on the compute cost of the network (326-327). Figure 9 depicts two common types of pooling.

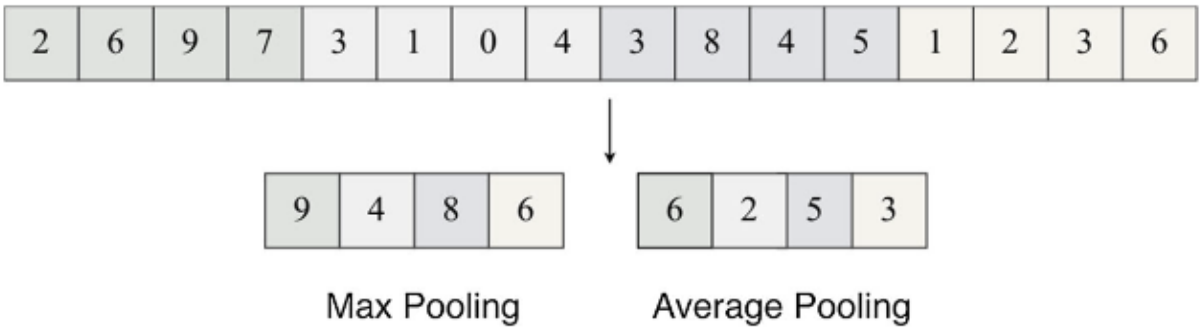

Figure 9: Pooling Layers of 1D-CNN (Bektash & la Cour-Harbo 2023, 327)

In Figure 9, the vector at the top depicts an output vector $\boldsymbol{o}_l^{(c_l)}$ consisting of 16 elements produced by a 1D convolution layer. The 1D-pooling layers both have a kernel size of 4, thus they output a 4-element wide vector. The left-side output vector is the product of a Max Pooling layer, which captures the maximum of the 4 adjacent values for each sub-grid. The right-side output vector is the product of an Average Pooling layer, which computes the average value of each sub-grid.

3.1.4 Brief Literature Review on FX Predicting CNN and LSTM Models

The FX market remains the largest and most traded financial market on our planet (BIS 2025), where algorithmic trading is beginning to dominate transaction volumes (Chaboud et al. 2023, 265). For time series prediction, FX markets pose a significant challenge due to their volatility, complex temporal dependencies, and their sensitivity to a plethora of exogenous factors (Biju et al. 2025). Thus, increasingly sophisticated models are becoming commonplace, with LSTM and CNN based ANNs rearing their heads in today's FX prediction literature. Table 2 summarizes the observed results from a sample of ANN based FX prediction literature surveyed for this thesis. Presented in Table 2 are inputs, empirical results, ML task type, and ANN structure, reported in the reviewed studies.

Table 2: Sample of Currency Prediction CNNs and LSTMs from Literature

Authors & Year	Input Interval	Employed EUR/USD	Prediction Task Type	Reported Metric	Input Features	Model Architecture	ANN Layer Structure
El Zaar et al. 2025	15-Minute	No	Classification	DA: 47.26%	Various Technical Indicators	Transformer-CNN	3 x CNN +GRU +Attention +3NN
Luangluewut & Thiennviboon 2023	Minute	Yes	Classification	DA: 93%	256x256 Pixel Open Price Plots	2D-CNN	3 x CNN + 2 x NN
Yıldırım et al. 2021	Day	Yes	Classification	DA: 64.24%	Various Macro and Technical Indicators	2 Parallel LSTMs	N/A x LSTM
Galeshchuk & Mukherjee 2017	Day	Yes	Regression and Classification	DA: 78.53%	Lagged Rates	1D-CNN	N/A
Markova 2023	5-Minute	Yes	Regression	RMSE: 0.0173	Open, High, Low, and Close Prices	1D-CNN-LSTM	2 x CNN + LSTM
Escudero et al. 2021	Day	Yes	Regression	MAPE: 0.282%	Lagged Rates	LSTM	2x LSTM
Van Hoa et al. 2021	Day	Yes	Regression	MAPE: 29.53%	Lagged Rates	LSTM and Autoencoder	2 x LSTM
Safi et al. 2020	Month	No	Regression	MAPE: 34%	Lagged Rates and Oil Price	1D-CNN	N/A

Below is a brief overview of findings from the literature reviewed. This is used to motivate the ANN models developed for this thesis. Model implementation of this thesis is discussed in section 4.3.

Galeshchuk and Mukherjee (2017) compared multiple classification models on three currency pairs: EUR/USD, GBPUSD (British pounds to USD), and JPYUSD, using daily closing rates from 2010 to 2015. Currency datasets were split into training and validation data at a ratio of 2/3 and 1/3 respectively. These data were used to train and evaluate multiple different models ranging from ARIMA[29] to Support Vector Machines, and different ANNs. The researchers found that of the models they tested, their CNN performed best, achieving about 78.53% accuracy on predicting the direction of change for the next day's EUR/USD rate. Their CNN model was constructed using a 5-day single channel 1D filter and four hidden layers using ReLU activation functions. (Galeshchuk & Mukherjee 2017, 104-106.) Additionally, Luangluewut and Thiennviboon (2023) investigated a EUR/USD trading model implementing a 2D CNN, which used 256x256 images of the currency rate as input. Input images were time series line plots at a 1-minute resolution. The CNN was constructed with three CNN-ReLU-MaxPooling layers and two fully-connected dense layers. This proposed model achieved 93% accuracy within their small set of test data. (Luangluewut & Thiennviboon 2023.)

[29] ARIMA refers to an Autoregressive Integrated Moving Average time series model.

As in section 2.1.2, the impact of oil prices on currency rates has seen investigation in the context of ML predictive regression tasks. For example, Safi et al. (2022) implemented oil prices as inputs to predict Nigerian exchange rates using monthly data that had been transformed by empirical mode decomposition. By applying the Diebold-Mariano (2002) test to various vanilla ANNs, LSTMs, and CNNs, the researchers found that a CNN model utilizing the decomposed time series produced the best results, predicting monthly rates with a RMSE of 0.59 error rate corresponding to a MAPE[30] of 34%. (Safi et al. 2020, 484, 490-491.)

Escudero et al. (2021) compared predictive performance of ARIMA, RNN, and LSTM models on daily EUR/USD rates. They contrasted forecast accuracy of the models for various forecast horizons starting from first day after and ending 5days or up to 55days ahead of the input data. The best predictions were achieved in the out-of-sample (test) data by the researchers when using the LSTM with 22days of min-max normalized EUR/USD as input. Their best model had two hidden layers with ReLU activation functions. This model achieved a RMSE of 0.004 and MAPE of 0.282%. (Escudero et al. 2021, 6-8, 10-11.)

In their paper researchers Yıldırım et al. (2021), investigated a hybrid dual LSTM prediction model where one model was trained on macroeconomic indicators and another on technical indicators. Researchers employed MA, MACD, momentum, rate of change, Bollinger bands and a commodity channel index as inputs for their technical indicators model. Additionally, they used a plethora of macroeconomic input features for their macroeconomic model. The researchers gathered monthly inflation rates, daily Federal Funds Rates, monthly EU interest rates for EU, and daily close prices for the DAX, S&P500, and EUR/USD. All models were trained on the same set of labels denoting price increases and decreases. To improve their models' accuracy, the researchers determined a threshold value to weed-out negligibly tiny changes in price, thus, inducing a favourable balanced dataset with more pronounced signals. The two models were trained as separate LSTMs and their outputs were evaluated separately, and together in rules based hybrid model. The hybrid model outperformed the latter two with a classification accuracy 64.24%, 63.91%, and 68,58% on a 1-day ahead, 3-day ahead, and 5-day ahead evaluation windows. (Yıldırım et al. 2021, 5-6, 9-12, 30-32.)

Furthermore, Markova (2023) employed an LSTM-CNN hybrid ANN to predict the next three instances of open, high, low, and close prices of the EUR/USD using 6 earlier instances. Their final

[30] MAPE refers to a common regression error metric Mean Absolute Percentage Error computed as: $\text{MAPE} = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{\hat{y}_t - y_t}{y_t}\right|$, where $|\cdot|$ denotes absolute value.

model used L2-regularization and Parametric leaky Rectified Linear Unit (PReLU)[31] activation functions for two 1D convolution layers followed by a MaxPooling layer and the LSTM layer. Their model reportedly achieved a RMSE of 0.014 within their test dataset. (Markova 2023, 302-304.) Van Hoa et al. (2021) implemented a novel LSTM approach using a two-model-setup, with an LSTM autoencoder to enrich the inputs for a main predictor LSTM. The autoencoder and LSTM were compiled and trained on 22 sequential daily currency rates as in Escudero et al. (2021, 8, 10). The autoencoder was constructed as an encoder followed by a decoder that aimed to return the same input sequence of 22 daily rates. The compressed representation constructed in the middle of the autoencoder (called a bottleneck) was collected and appended to the input of the main predictor LSTM. This was motivated by the autoencoders' purpose to convert input into output with minimum distortion, such that it filters and retains the necessary information to reproduce an output that is identical to the input. In the framework of ML tasks provided by Deisenroth et al. (2020, 255), the autoencoder's purpose is dimension reduction. The main predictor-LSTM was implemented with same 22 rates and the bottleneck to forecast the value of the next day's closing price. The researchers found that among models tested, their LSTM was the most accurate producing a RMSE and MAPE of 0.00783 and 53.74% for the EUR/USD, respectively. Notably, their novel approach implementing the LSTM-autoencoder achieved a superior RMSE and MAPE of 0.00464 and 29.53% for the EUR/USD, respectively. (Van Hoa et al. 2021, 25-27.)

Recently, El Zaar et al. (2025) evaluated the performance of a 1D-CNN and transformer based model for detecting the directional changes in the crypto currency market sampled at a 15-minute interval. Comparing their novel *Transformer-CNN* hybrid model to a transformer only and CNN only model, the researchers observed that their proposed model consistently outperformed the latter two. Their model achieved an accuracy of 47.26% within their test data. While the market for crypto currency is inherently different to the EUR/USD, the features used were familiar to the FX context: technical indicators, trading volumes, and open, high and close prices. (El Zaar et al. 2025, 1240, 1252.)

Additionally, Dhamija and Bhalla (2010) explored the comparative performance of a variety of autoregressive conditional heteroskedasticity (ARCH) models and ANNs for out-of-sample forecasting on the log returns of five different currency pair time series. The researchers employed models such as ARCH, GARCH, GARCH-M, TGARCH, EGARCH and IGARCH, and benchmarked them against standard multilayer perceptrons and radial basis networks. The ANNs

[31] A PReLU is an activation function called a Parametric leaky Rectified Linear Unit, $\mathrm{PreLU(x)} = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}$.

consistently outperformed the ARCH models. Interestingly, for the EUR/USD dataset, ARCH models performed at levels very close to ANNs with the EGARCH model surpassing the two ANN architectures on one of the three metrics used. (Dhamija & Bhalla 2010, 194-195, 203-206.)

Notably, in literature a variety of error metrics and model structures are used depending on the task at hand. Interestingly the results reported on model accuracy between studies vary significantly. For example, Luangluewut and Thiennviboon (2023) reported a stellar 93% directional accuracy (DA) on a smaller dataset, yet Yıldırım et al. (2021) reported a 64.24% accuracy on their 2015-2018 test sample data. This difference may be partially explained by sampling intervals used: minutes and days, respectively, or perhaps to an extent variance from dataset sizes.

Drawing on sampled literature of ANN prediction models for FX markets, it could be posited that more complex LSTM and CNN might be superior to vanilla ANN models (Galeshchuk & Mukherjee 2017, Escudero 2021) and typical time series models (Dhamija & Bhalla 2010), provided that a stable configuration is employed (Markova 2023, 302-304). Additionally, reported metrics and model architecture in Table 2 seem to suggest that a set of 1-2 CNN layers followed by less complex LSTM and vanilla ANN layers have a productive track record in FX prediction literature.

3.2 Natural Language Processing: *From Text to Topics and Sentiment*

To create inputs for the sentiment-based EUR/USD prediction, two NLP models, STMs and BERT, were implemented on the unstructured dataset of FOMC statements and minutes. The STM is used to dissect documents into distinct topics, granting automated informed understanding of document contents. The BERT model allows transformer architecture to be leveraged for *context-aware* sentiment classification of the documents. This subsection introduces these two NLP models briefly.

3.2.1 Understanding Structured Topic Models

Topic models can be employed in financial document analysis, condensing long unstructured data from the word level to the topic level, allowing for efficient document analysis (Hansen & McMahone 2016; Jegadeesh & Wu 2017; Aattouchi & Kerroum 2022; Ferrara et al. 2022; Koelbl et al. 2024). Thus, within the framework provided by Deisenroth et al. (2020, 255), topic modelling could be seen as a task within the remit of the dimensionality reduction pillar. The topic model implemented in this

thesis is the Structured Topic Model (STM). The STM is a topic model based on Latent Dirichlet Allocation (LDA), which models text data by assuming that text is generated as a mixture of words whose presence is attributed to a set of underlying topics (Blei et al. 2003). The novelty of the STM stems from its ability to model metadata dependencies on both the topic-level and content-level distributions. In practice the STM could account for surrounding market conditions via date metadata variables, contributing to its suitedness for this thesis.[32]

This subsection introduces the STM following Roberts et al. (2016) and Roberts et al. (2019 2-4). Key definitions related to topic models are defined in this paragraph. *Document*: a single piece of text data analyzed using an NLP method, denoted as *d*. For example, a FOMC statement or a minutes release or a single clause within is an example of a document. *Corpus*: a text dataset consisting of a collection of selected documents, denoted as *D*. *Vocabulary***:** Instances of unique words or terms in the documents of the corpus, denoted as *V*. *Metadata*: observable covariates which influence document's text, such as the author, date, and text type. *Token:* a meaningful representation formed from splitting words or sentences (Rai & Borah 2020, 198-199).

Akin to other topic models, the STM architecture defines a data generating process by which the documents it models were created by. To train a STM, a corpus is used to estimate the most likely parameter values for a model of the data generating process which could produce the documents in the corpus data (Roberts et al. 2019, 2.) A STM models the corpus it trains on starting with the document-topic and topic-word distributions. The STM models the data generating process for documents *d,* of a corpus *D* (Roberts et al. 2019, 2). Each document indexed with *d* consists of words (or their positions) *n*. Corpus *D* contains members of the unique set of tokens of interest or terms *V* indexed by *v.* (Roberts et al. 2016, 988-991.) The topics constitute the documents; the number of topics chosen is K. Following Roberts et al. (2016, 988-991), this is typically expressed as:

$$d \in \{1, \dots, D\}, \qquad n \in \{1, \dots, \mathrm{N_d}\}, \qquad v \in \{1, \dots, V\}, \qquad k \in \{1, \dots, \mathrm{K}\}.$$

Importantly, a document *d* consists of words $w_{d,n}$ order by their position *n*. Each term *v* in the vocabulary is unique, but words $w_{d,n}$ of position *n* in document *d* can instantiate multiple occurrences of the same term *v*. Also, for each document *d,* there is a distribution of topics $\boldsymbol{\theta}_d$, and for each topic *k*, there is a topic-term distribution $\boldsymbol{\beta}_k$; both distributions follow a Dirichlet distribution. The STM

[32] STMs extend typical LDA topic-model frameworks by integrating document-level metadata to model the portions of topics in a document and the portions of words used in a topic (Roberts et al. 2016, 990-991).

models the topic proportions[33] so that the sum of topic proportions $\theta_{d,k}$ is one, and the sum of topic-term probabilities[34] is one. (Roberts et al. 2016, 988-991; 2019, 2-3).

$$\boldsymbol{\theta}_d = (\theta_{d,1}, \dots, \theta_{d,\mathrm{K}}),$$

$$\boldsymbol{\beta}_k = (\beta_{k,1}, \dots, \beta_{k,V}).$$

Another important construct is the **B** matrix, which encodes all topic-term distributions associated with topics *k* over terms *v* of the vocabulary, defined as:

$$\boldsymbol{B} = (\boldsymbol{\beta}_1 | \dots | \boldsymbol{\beta}_K).$$

A document *d* is expected to be generated so that for each word position *n*, a word-topic assignment $z_{d,n}$ is generated from a $\mathrm{Multinomial}_K$ parameterized by θ_d. Subsequently, a word $w_{d,n}$ is generated from a $\mathrm{Multinomial}_V$ on the product of the word-topic assignment $z_{d,n}$ and the $\boldsymbol{B}$ matrix, of all topic-term distributions β_k. (Roberts et al. 2016, 988-991; 2019, 2-3). Formally stated as:

$$z_{d,n} \sim \mathrm{Multinomial}_{\mathrm{K}}(\boldsymbol{\theta}_d, 1),$$

$$w_{d,n} \sim \mathrm{Multinomial}_V(z_{d,n}, 1).$$

This framework—so far, presented like other typical LDA topic models—defines topics as a mixture over words $w_{d,n}$, where realizations for each word follow word-topic assignment distributions representing the probabilities of that word belonging to a topic (Roberts et al. 2019, 2-3). The elegance of the STM stems from two features: (1) it allows for correlations between topics,[35] and (2) it allows document-level metadata to affect the content (vocabulary) associated with topics and the portion of a document devoted to a topic (Roberts et al. 2016, 991).

To achieve these two features, the STM models topic-term distributions on the document-level, and topic distributions $\boldsymbol{\theta}_d$ are modeled with a document-level logistic normal distribution. The topic distribution mean $\boldsymbol{\mu_d}$, and topic-term distributions $\beta_{d,k,v}$, are distinct from LDA equivalents, being document-level specific instead of global parameters (corpus-level). (Roberts et al. 2016, 990-991.)

$$\boldsymbol{\theta}_d \sim \mathrm{LogisticNormal}_{\mathrm{K}-1}(\boldsymbol{\mu_d}, \boldsymbol{\Sigma}).$$

---

[33] Portion of a document allocated to topic *k,* or topic prevalence (Roberts et al. 2016, 989).
[34] Probabilities of terms *v* being associated with a topic k are approximated by a proportional measure $\beta_{d,\mathrm{k,v}}$ for a STM.
[35] Topic distributions are assumed to generate from a logistic normal distribution instead from a Dirichlet distribution. This is a defining feature of the correlated topic model (CTM) framework, which the STM expands upon with metadata.

The variance-covariance matrix $\boldsymbol{\Sigma}$ of the logistic normal distribution allows for the core language model to produce correlated topic distributions $\boldsymbol{\theta}_d$. $\boldsymbol{\mu_d}$ is modeled on document-level metadata. Each document *d* has metadata associated with it. Topic-covariates $\boldsymbol{x}_d$ are selected from metadata and can are denoted for each covariate variable *p*. (Roberts et al. 2016, 988-991.) Formally:

$$\boldsymbol{x}_d = (x_{d,1}|x_{d,2}|\dots|x_{d,P}), \qquad p \in \{1, \dots, P\}.$$

$\boldsymbol{\mu_d}$ is defined as a linear combination of the topic-covariates $\boldsymbol{x}_d$ and a regularization matrix $\boldsymbol{\Gamma}$,

$$\boldsymbol{\mu}_d = \boldsymbol{\Gamma}'\mathbf{x}'_d,$$

$$\boldsymbol{\Gamma} = (\boldsymbol{\gamma}_1|\dots|\boldsymbol{\gamma}_{\mathrm{K}-1}),$$

$$\boldsymbol{\gamma_k} = (\gamma_1, \dots, \gamma_{\mathrm{P}}), \qquad \boldsymbol{\gamma_k} \sim \mathcal{N}(0, \sigma_k^2 \boldsymbol{I}_P) \quad \forall\, p = 1, \dots P,$$

$$\sigma_k^2 \boldsymbol{I}_p = \begin{pmatrix} \sigma_k^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_k^2 \end{pmatrix}_{P \times P}, \qquad \sigma_k^2 \sim Gamma(s^\gamma, r^\gamma).$$

Additionally, to account for the effects of the topic-level covariates $\boldsymbol{x}_d$ on the topic-term distributions, the STM $\boldsymbol{B}_d$ matrix is defined as:

$$\boldsymbol{B}_d = (\boldsymbol{\beta}_{d,1}|\dots|\boldsymbol{\beta}_{d,K}).$$

The individual topic-term distributions per topic are defined as (Roberts et al. 2016, 991),

$$\beta_{d,k,v} = \frac{\exp\left(m_v + \kappa_{k,v}^{(t)} + \kappa_{y_d,v}^{(c)} + \kappa_{y_d,k,v}^{(i)}\right)}{\sum_{v=1}^{V} \exp\left(m_v + \kappa_{k,v}^{(t)} + \kappa_{y_d,v}^{(c)} + \kappa_{y_d,k,v}^{(i)}\right)},$$

where, m_v represents the corpus-wide *baseline* distribution for term occurrence. Thus, the deviations from this baseline are expressed with parameters $\{\kappa\}$.[36] When training the STM on a corpus *D,* all parameters (i.e. {s, r, ρ, K}), except hyperparameters, are estimated with an expectation maximization algorithm and initial sampling methods (i.e. Gibbs sampling). (Roberts et al. 2016, 991-993.)

The data generation process modeled by the STM is summarized in Figure 10. The greyed-out components *X, Y,* and *w* represent components directly observable from the training data. The uncolored components $\Sigma, \gamma, \theta, z, \beta$ represent parameters estimated and constructed by the STM. The

[36] Specifically, $\kappa_{k,v}^{(t)}$ denotes topic deviation, $\kappa_{y_d,v}^{(c)}$ represents content covariate deviation, and $\kappa_{y_d,k,v}^{(i)}$ represents content covariate-topic deviation (Roberts et al. 2016, 991).

plates represent replicates. The outer plate represents the corpus, and the inner plate portrays repeated choice of words and topics for each document of the corpus. (Roberts et al. 2016, 990.)

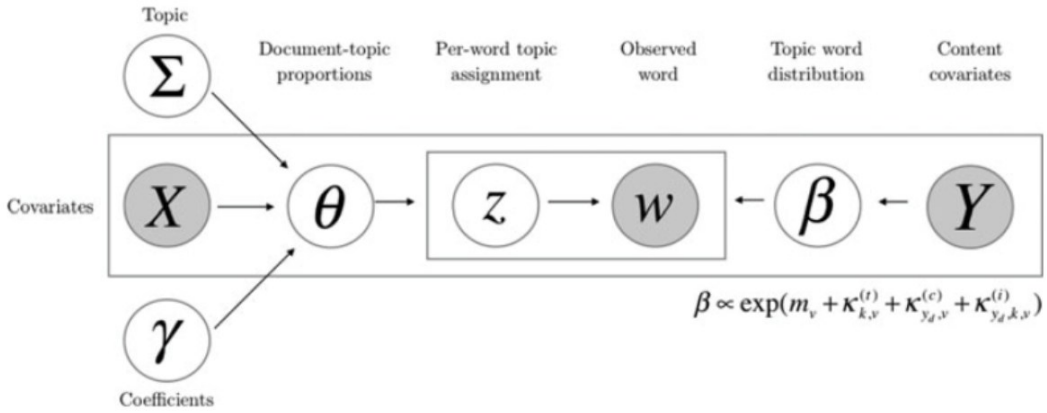

Figure 10: Graphical Illustration of the STM (Roberts et al. 2016, 990)

3.2.2 Meet Bidirectional Encoder Representations from Transformer aka BERT

Bidirectional encoder representations from transformers, abbreviated BERT, is a type of NLP model implemented chiefly for classification tasks. BERT is a powerful NLP tool which has seen increased use in domain of finance, and relevantly, in deciphering the sentiment behind FOMC releases. Built on the same foundational research which enables today's generative *AI-chat bots*, BERT relies on the revolutionary transformer architecture introduced in 2017.

Transformer architecture lays the foundations on which both BERT and today's Large Language Models (LLM) stand. This architecture revolutionized modern AI and was introduced by eight Google employees in their landmark paper "*Attention is All You Need",* hence referred to as Vaswani et al. 2017. The researchers proposed transformers, which employ an *attention* mechanism introduced by Bahdanau et al. (2014) to draw global dependencies between input and output sequences flexibly and regardless of their distance. Both Bahdanau et al. (2014) and Vaswani et al. 2017 are set in the context of neural machine translation, where an ANN model aims to convey meaningful information encoded in one human language to another, as accurately and as correctly as possible.

At the time of Bahdanau et al. (2014), machine translation research was largely conducted with RNN encoder-decoder models which were limited by the rigid fixed-length vector produced by RNN encoders. RNN implementations of the time had an encoder that needed to squash all necessary input information into a fixed length vector, which was then passed to the decoder to be represented in the translated language. This resulted in lost contextual information for longer sentences.

The attention concept introduced by Bahdanau et al. (2014), allowed the models to *attend* to the most informative parts of the input by *soft searching* for a relevant set of input tokens or their mappings produced by the encoder. This enables the attention mechanism to allocate computational resources efficiently, accounting for the surrounding context of each token w_n (Chaudhary 2025, 18-19). In practice, an encoder maps input sequences into hidden states called *annotations* $\boldsymbol{h}_n$. The annotations hold information on the neighborhood of each input *position*[37] n, with a focus on the tokens surrounding token w_n. This can be achieved with an adapted bidirectional RNN as in the original paper. A trainable alignment model then uses the annotations to calculate a context vector. (Bahdanau et al. 2014, 1-4, 7-8.) At each decoder step m, the model scores how well annotations $\boldsymbol{h}_n$ align with the previous step, and defines a context vector as a weighted sum of these scores, formalized as,

$$\boldsymbol{context}_m = \sum_{n=1}^{N} \alpha_{m,n} \boldsymbol{h}_n .$$

Each weight is softmax normalized from the alignment scoring, denoted $\mathrm{score}(\cdot)$, such that,

$$\alpha_{m,n} = \frac{\exp\left(\mathrm{score}(s_{m-1}, \mathbf{h}_n)\right)}{\sum_k^N \exp\left(\mathrm{score}(s_{m-1}, \mathbf{h}_k)\right)},$$

where, s_{m-1} represents the previous RNN hidden state of the decoder. The current hidden state used to predict the current target token y_m, is then computed by a RNN as,

$$s_m = f(s_{m-1}, y_{m-1}, \boldsymbol{context}_m).$$

Building on the foundations of this attention mechanism, Vaswani et al. (2017) proposed the transformer model. Transformers expand the attention mechanism with a self-attention mechanism and replace the RNN structure of previous encoder-decoder models, allowing for dynamic modelling of global dependencies while parallelizing computations. Such advances not only improved accuracy but reduced training time (Chaudhary 2025, 18-19).

The transformer model encoder architecture, as presented in Vaswani et al. (2017, 2-6), is introduced here to provide background for BERT. BERT is essentially built on a refined[38] stack of such encoder blocks. While the original transformer was proposed as a sequence-to-sequence architecture for

[37] In this context, *positions* are representations of each token's location in the sequence.
[38] The difference between original Transformer encoder and the improved version for BERT is bidirectionality.

neural machine translation, BERT models in this thesis use only the encoders to perform sentiment analysis in a sequence-to-classification setting. Consequently, the decoder is not discussed.

The encoder is implemented from the input layer of the transformer. The input tokens are initially embedded or mapped into a continuous vector representation.[39] Because transformers avoid recurrence- and CNN-like sequenced inputs, which inherently preserve the order information, the transformer then injects positional encodings to its embeddings to retain information on relative or absolute positions of tokens. The encoder of a transformer consists of two sub-layers: a Multi-Head Attention (MHA) layer, and a position-wise feed-forward network. Each layer employs residual connections that connect around the sub-layer to be concatenated with that sublayer's output and normalized together before feeding forward to the next layer or encoder block. The encoder is depicted in Figure 11. (Vaswani et al. 2017, 3,5-6.)

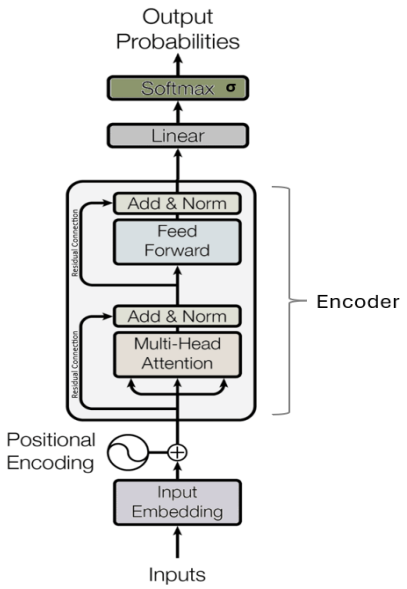

Figure 11: Simplified BERT-like Classification Encoder (Adapted from Vaswani et al. 2017, 3)

The embedding layer maps the input tokens into a vector that captures its semantic meaning, which can then be operated on with the ANN. Because input tokens are processed in parallel, the positional encoding ensures that positional information is preserved to meaningfully represent relative or absolute positions of tokens. The resultant representation produced by these two procedures is then passed on as input to the encoder stack. (Vaswani et al. 2017, 3,5-6.)

[39] A classic example for illustrating the meaning captured by embedding is that subtracting the embedded vector for the token of the word *man* from that of *king* and adding it to the embedding vector of *woman*, should result in an embedding vector representing the concept of *queen*.

The MHA sublayer applies multiple self-attention functions to attend to information from representations of its input to provide *contextual meaning* in its output. At the center of MHA, self-attention extends the attention mechanism introduced by Bahdanau et al. (2014), by calculating the alignment score between different positions of its input sequence, through internal dependencies (Chaudhary 2025, 19). Conceptually, the attention function maps a set of queries $\boldsymbol{Q}$ and key-value $\{\boldsymbol{K}, \boldsymbol{V}\}$ pairs to its output. (Vaswani et al. 2017, 3-5.) The attention score[40] is then calculated as,

$$\boldsymbol{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \boldsymbol{\sigma}\left(\frac{\boldsymbol{QK'}}{\sqrt{d_k}}\right)\boldsymbol{V},$$

where d_k denotes the length of vectors $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$. The output of the function is a weighted sum of values, such that each Value is given a probabilistic scaled weight, determined by comparing the Query and Key vectors. Here, Query and Key compute a compatibility function to determine alignment of its input vectors, which determines the mix of values appropriate for the output vector. In practice, as multiple attention functions are executed simultaneously in parallel, the **Q**, **K**, and **V** are computed with matrices and concatenated into a vector.

After normalization the output vector of the MHA layer is passed to the position-wise feed-forward network. This layer is essentially like a vanilla ANN layer applied to flexibly model and enrich input of the next layer. In the original paper a ReLU activation function was applied at these sublayers. (Vaswani et al. 2017, 3-5.) Overall, transformer encoder blocks are quite a sophisticated combination of MHA and feed forward layers, which first attend to the relationships between input components and second enrich their representations in a sequential manner.

Leveraging a stack of such encoder blocks, the BERT model was developed by Google and released as open source accompanied by the research paper Devlin et al. in 2019. The BERT model uses pre-trained encoder blocks of a transformer, and can be modified with a SoftMax activation function, which could characterize a predicted word, sentence ordering, or categorization, depending on the task at hand (Araci 2019, 3), as visualized in Figure 11. Two BERT models were released as pre-trained encoders trained with unsupervised learning on a very large corpus that could be fine-tuned to suit a wide range of NLP tasks. Indeed, training a BERT model consists of two phases: an intensive pre-training phase, and a less intensive fine-tuning phase, of which the former was carried out by the

[40] This attention score is sometimes referred to as scaled dot-product attention.

Google team to prepare the model to learn semantic meanings in text data. (Devlin et al. 2019, 1,4-5,9.) BERT could be viewed as an almost out of the box ready solution for NLP classification.

Subsequently, soon after the release of BERT, for his master's thesis, Araci (2019) introduced FinBERT for financial sentiment analysis. The researchers modified the original BERT-base with 12 encoder layers and fine-tuned their modified models with a substantial financial corpus provided by Reuters, consisting of 1.8 million news articles published between 2008 and 2010. The FinBERT model results indicated a 15 percent increase in accuracy over the state-of-the-art (SOTA) models of the time for financial sentiment classification. Also, Araci 2019 concludes with a notion that FinBERT is useful for obtaining explicit sentiments, but modelling implicit sentiment is another task completely because such sentiment might not be clear even to the writer. (Araci 2019, 4-6, 9.)

Extending BERT architecture to the context of FOMC statements, Shah et al. (2023) fine-tuned a SOTA BERT model (RoBERTa-large) to construct FOMC-RoBERTa. Their model was trained on three different sources of data: FOMC minutes, press conference transcripts and speeches, from which sentences were extracted and manually labeled to have hawkish, neutral, or dovish sentiment regarding monetary policy. The dataset was collected from these FOMC documents released between January 1996 and October 2022. The researchers compare their model against various BERT models, two LSTMs, dictionary-methods, and ChatGPT 3.5 (Turbo). Their results indicated that a RoBERTa-base and their model FOMC-RoBERTa outperform the rest as measured by F1-scores observed in CV. Also, the researchers designed and implemented an aggregating sentiment measure for the FOMC documents based on sentence-level sentiment, formalized in Equation 28 below,

$$Hawkishness_d = \frac{\#Hawkish_d - \#Dovish_d}{\#Total_d}, \tag{28}$$

where the sentence counts for both hawkish and dovish sentences is normalized by the total number of sentences in the document d. This measure appeared highly correlated with both the consumer and producer price indexes. Also with their measure, the 3-month, 1-year, and 10-year Treasury yields could be predicted at release with statistically significant regression models, lending further validity to their model and proposed measure and BERT model. (Shah et al. 2023, 6669-6672.)

Moreover, Gössi et al. 2023 extended FinBERT by fine-tuning and testing it with 3535 complex sentences extracted from FOMC minutes, which presented contrasting points using conjunctions such as *while*, *though*, and *but*. These complex sentences helped the fine-tuned model to achieve superior results in labeling minutes' sentences as neutral, positive and negative based on their sentiment of the

financial economy. The model fine-tuned by the authors, FinBERT-FOMC, showed remarkable improvement in classifying the financial-economic sentiment of complex sentences with an accuracy of 86.14% compared to FinBERT's 67.33%. (Gössi et al. 2023, 357,361-364.) Continuing research on this model, one of the original authors, Wonseong Kim, co-authored a paper further comparing performance of FinBERT-FOMC against other sentiment classification methods. These other methods included LLMs GPT-4 and Llama-3-70B. Interestingly, on their FOMC minutes dataset, the open-source Llama-3-70B model outperformed other models when prompt engineering was utilized. In their tests, FinBERT-FOMC achieved a F1-score of 0.639, while the same figure for GPT-4 was 0.627 and an impressive 0.784 for their selected Llama model. (Kim et al. 2024, 626, 628-632.)

What is important to note is that all three papers (Shah et al. 2023; Gössi et al. 2023; Kim et al. 2024) employed expert labeled datasets to train for and benchmark the sentiment classification task. Thus, the BERT models used in this thesis attempt to mimic the classifications provided by human experts who created the labeled training data. Indeed, true sentiment of a text, as Araci (2019, 9) in their thesis concludes, which might not be recognized even by its writer, will not -by all likelihood, be fully captured by an NLP model, at least for now.

4 Data and Implementation

This thesis implemented five types of ML models and two data sets. The two data sets were structured and unstructured data sets, which are discussed in sections 4.1 and 4.2 respectively. The structured data set consisted of numeric data such as EUR/USD market data, macroeconomic variables, and technical indicators. Implementing this data, three ANN intraday regression models were developed. The unstructured data set consists of web-scrapped FOMC statements and minutes documents. The documents were preprocessed at the paragraph-level and used to generate sentiment features with STMs and BERT models to the document-level for. Sentiment features were used as inputs for sentiment models. Figure 12 demonstrates use of data and ML models in this thesis.

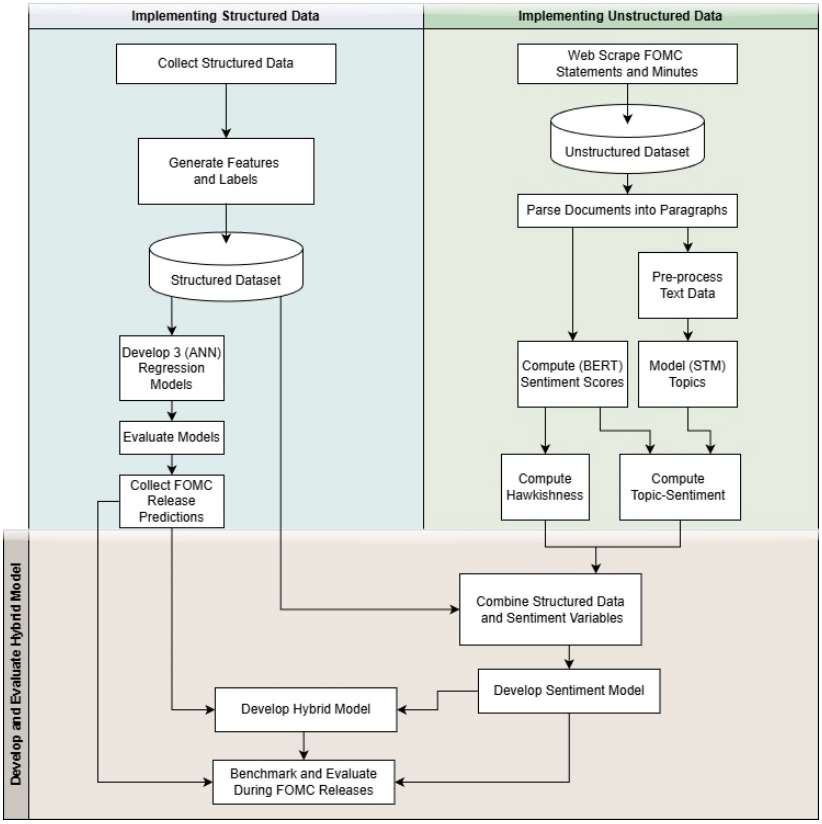

Figure 12: Overview of Models and Data Implementation

To develop and evaluate intraday EUR/USD ANNs, 12 test windows were used, one for each quarter of 2022, 2023, and 2024. For hybrid and sentiment models, the three years of the test windows were used to develop and evaluate models. During 2022, 2023, and 2024 the FOMC released eight statements and minutes each year, producing 48 points for analysis within the selected test data. ANNs trained on a year of intraday data, and sentiment models with minimum 18 years of data depending on test year. Two hybrid models were constructed on top of ANN and sentiment models.

4.1 Assembling the Structured Dataset

4.1.1 Macroeconomic and Technical Indicators

Temporal resolution of all structured data used was scaled to one minute. This initial part of the data used in this study is the structured dataset, containing numeric variables representing fundamental macroeconomic and technical indicators explored in sections 2.1.2 and 2.1.3. Features of this dataset were collated by (1) joining datasets gathered from ECB (2025), United States Treasury (2025), histdata (2025), FactSet (2025), and Dukascopy (2025), and by (2) generating new features from the data. All together the structured dataset had 86 features and more than 7 million instances.

The base of the structured dataset is a EUR/USD dataset consisting of high, low, close, and open prices, as well as trade volume for both the bids and asks. The EUR/USD dataset was gathered from Dukascopy Bank (2025) and spans 31.12.2003 to 1.1.2025. Macroeconomic variables gathered pertain to interest rates, inflation, crude oil prices, and returns of the S&P500. Interest rates were proxied for both economies of the United States and the Eurozone. Interest rates for 3Month, 6Month, 1-Year, 2-Year, 3-Year, 5-Year, 7-Year, 10-Year, 20-Year, 30-Year maturities were proxied by par yields[41] captured at a daily resolution, published by both the United States Treasury (2025) and the ECB (2025). These yields were appended to the minute resolution dataset in a rolling fashion, so that the features' values update at release times indicated on the United States Treasury's and ECBs websites, specifically at 15:30 and 6:00 ET respectively. The time series for expected inflation rates were acquired from data provider FactSet and St. Louis Fed (2025), selected for their availability and comparability. Both series concerned long-term inflation expectations, calculated as 10yr breakeven implied inflation rates derived from treasuries[42] for the United States, and from Euro area inflation swaps[43] for the Eurozone. The inflation data was recorded at a daily resolution spanning 2013-2024. These Implied expected inflation rates were appended to the dataset in rolling fashion, similar to the par yields. The time series for the S&P500 prices were gathered from histdata (2025) at a one-minute resolution, spanning 2010-2024. Crude oil prices were collected for West Texas Intermediate (WTI) and Brent, to proxy costs for this critical energy resource in the United States and Europe respectively, with the findings of both Chen and Chen (2006, 400-403) and Vochozka et al. (2020, 185-186) in mind. WTI and Brent data were collected from histdata (2025) at a 1-minute resolution, spanning

[41] Par yield is a theoretical coupon rate at which the market price of a bond equals its par value (Hull 2017, 105).
[42] Breakeven inflation, as defined by the Fed (2010), is difference between nominal and TIPS yields of same maturity.
[43] Breakeven inflation rates can be defined from par rates of a zero-coupon inflation linked swaps as follows.
If f_t denotes the par rate for a swap maturing at time t, and $I^e{}_t$ represents the expected price level, then the payout from a par swap is: $1 + f_t = \frac{I^e{}_t}{I_0}$. Then, under the risk-neutral measure Q, $1 + f_t = E_0^Q[1 + \pi_t]$. (Gimeno & Ibáñez 2018.)

2010–2023 and 2010–2024 respectively. All instances of data were joined with 1-minute resolution ET timestamps, and the final structured dataset was chronologically ordered from oldest to newest.

The minute-frequency timestamps were used to append release times for FOMC documents and join in sentiment features for hybrid models. Release timestamps were collected from the FOMC and supplemented with previous research such as Handlan's manuscript (2020, 44-45) and Vega et al. (2025, 46). Integrity of timestamps was double checked by observed volatility and trade volume peaks. Sixteen of the 298 timestamps were not appended due to unreliable or missing timestamps.[44]

The technical indicator variables created for the structured dataset were constructed a top a mid-quote price P_t inspired by Rosa (2013 68) and Ben Omrane et al. (2019, 5),

$$P_t = \frac{1}{2}(Closing\ Bid_t + Closing\ Ask_t\ ).$$

The return series for feature generation were created for each instance based on the mid-price. Returns were calculated as simple returns and log-returns, denoted hence as *R* and *r* respectively.

$$r_{t,int} = \ln(P_t) - \ln(P_{t-int}),\ int \in \{1min, 5min\}.$$

Thus, 1-minute and 5-minute log-returns were calculated.

Given EUR/USD's economic significance to countries in European and American time zones, this paper segments a trading day into four *sequential markets* following Wang and Yang (2011, 91-93). Also, the two-hour overlap in trading hours shared between London and New York-colloquially coined "witching hours"-are considered as its own binary feature, in addition to the 3 major sequential market features for Asia, Europe, and the United States (Chaboud et al. 2023, 253). These variables are one-hot encoded for neural networks, and treated as dummy variables, with one encoded variable dropped to represent the reference class for logistic regression. See Appendix 3 for a table of trading hours and designation of *sequential markets* by UTC (summer) time.

To proxy intraday daily, hourly, 15-minute and previous market session currency volatility is proxied by adapting approaches from Menkhoff et al. (2012, 692) and Su and Zhang (2018, 36-37). Consequently, daily, hourly and 15-minute volatility is proxied from minute log-returns as:

[44]Reliable timestamps for documents for meetings held on 06/12/2007, 09/01/2008, 21/01/2008, 10/03/2008, 24/07/2008, 29/09/2008, 07/10/2008, 16/01/2009, 07/02/2009, 03/06/2009, 09/05/2010, 15/10/2010, 01/08/2011, 28/11/2011, 16/10/2013, 04/03/2014, 04/10/2019, 03/03/2020, and 15/03/2020 were not available (some had weekend releases).

$$\sigma_t^{FX_q} = \frac{1}{q} \sum_{m=t-q}^{t} |r_{1\min}|,$$

where, r_{1min} denotes a one-minute log-return, and q depicts number of minutes in the intraday interval defined. For daily, hourly, and 15-minute intervals, q is assigned values 1440, 60, and 15, respectively. Also, motivated by empirical testing of the intraday variance estimators by Liu et al. (2015, 309-310) the 5-minute realized variance (RV) was used. RV is formally defined as:

$$RV_t^{5min} = \sum_{m=t-4}^{t} r_{i,1min}^2 .$$

However, intraday realized variance measures for continuous time series are highly susceptible to noise. Thus, Following Su and Zhang (2018, 36-37) a Two-scale Realized Variance (TSRV) method suggested by Barndorff-Nielsen et al. (2008) was applied as it has been shown to be a consistent estimator for intraday variance. Market session variance sampled at -minute intervals is calculated as realized variance (RV) following Su and Zhang (2018, 37) and Oya (2011, 1292),

$$RV_s = \sum_{i=T_s-m_s+1}^{m_s} r_{i,1min}^2 = \sum_{i \in \mathcal{G}_s} r_{i,1min}^2 .$$

Here T_s represents time index of the closing minute of the session, and m_s the number of 1-minute intervals in the market session. $\mathcal{G}_s$ represents the full grid of indices. RV_s is the sum of squared 1-minute log-returns in session s. To robustly proxy session variance TSRV is constructed as follows:

$$TSRV_s = \frac{1}{5}\sum_{j=1}^{5} RV_{sj} - \frac{m_s - 5 + 1}{5 \times m_s} RV_s.$$

RV_{sj} is the sum of squared 5-minute log-returns within sub-grid j of session s. The sub-grids are non-overlapping constituent elements of the grid $\mathcal{G}_s$.

$$RV_{sj} = \sum_{i \in \mathcal{G}_s^{(j)}} r_{i,5min}^2.$$

The sub-grid sample indices are essentially partitions of full grid $\mathcal{G}_s = t_1^{(s)}, t_2^{(s)}, \dots, t_{m_s}^{(s)}$, with the sub-grids denoted as $\mathcal{G}_s^{(j)} = \{t_j^{(s)}, t_{j+5}^{(s)}, \dots, t_{j+5\times(m_{s,j}-1)}^{(s)}\}$, such that $\mathcal{G}_s = \cup_{j=1}^{5} \mathcal{G}_s^{(j)}$ and $\cap_{j=1}^{5} \mathcal{G}_s^{(j)} = \emptyset$.

It is important to note that this estimator is based on Zhang et al. (2005, 1396-1399) which was originally implemented for data with tick or second length intervals. Thus, this estimator is an adaptation. Additionally, the raw TSRV feature was converted into a volatility estimate by taking the square root as in Plíhal (2021, 1381). Overall, the volatility features appended to the structured dataset were a one session lagged $TSRV_{s-1}$, and one-minute lagged daily, hourly, and 15-minute volatility measures $\sigma_{t-1}^{FX_q}$. All volatility features are summarized in Table 3 below.

Table 3: Volatility Features in Structured Data

VOLATILITY MEASURE	TIME INTERVAL
$\boldsymbol{\sigma_{t-1}^{FX_q}}$	15 minutes
	60 minutes
	1440 minutes
$\boldsymbol{TSRV_{s-1}}$	*Trading Session*[45]

To avoid forward-looking bias, the volatility estimators were lagged by length of one minute and one session or one minute. Thus, in its corresponding instance, the TSRV feature represents the aggregate total volatility of the last market session, and the other volatility measures $\sigma_{t-1}^{FX_q} \ \forall q = 1440, 60, 15$ gauge market volatility as measured a minute ago. This one-minute-lag respects the one-minute computation time constraint considered for fair model performance evaluation. All models and subsequent market direction predictions are expected to take a minute to complete and implement.

The moving averages following the formalizations adopted from literature and presented in Equations 3, and 4 (see section 2.1.3), were appended to the structural dataset. Lags were added based on previous research and the volatility measures chosen. In previous currency research, Hansun and Kristanda (2017, 13)[46] used 10 observation moving averages, and an earlier conference paper by Hansun (2013, 2) employed 5 observations. Additionally, Luangluewut and Thiennviboon (2023) used 13 and 26-minute moving averages. Thus, the lags chosen for moving averages were 5, 10, 13, 15, 26, 60, and 1440 minutes. Based on these lags, SMA and EMA indicators were constructed; furthermore, building on the moving averages, four MACD indicators (see Equation 5, section 2.1.3) were added. Two MACD indicators were constructed for both SMA and EMA with identical lead-lag periods of 13-23 and 15-60 minutes. As volatility measures, these indicators were lagged by one minute. Moving average-based indicators are listed in Table 4 on the next page.

[45] Trading sessions were specified following Wang and Yang (2011, 91-93), see Appendix 3 for details.
[46] This paper highlighted that EMA outperformed SMA when forecasting currencies intraday and included the EUR/USD as a benchmark dataset.

Table 4: Moving Average Features in Structured Data

INDICATOR	LEAD	LAG	INDICATOR	LEAD	LAG
SMA		5	**EMA**		5
		10			10
		13			13
		15			15
		26			26
		60			60
		1440			1440
MACD-SMA	13	26	**MACD-EMA**	13	26
	15	60		15	60

NOTE: *ALL VALUES ARE IN MINUTES*

Notably, BBs are not included explicitly as input features, because node interactions in dense layers of ANNs could create optimal BB-like features from the input return and volatility features.

The structured data was appended with SR indicators, created with both trading session-based pivot points, and intra-session Fibonacci-ratios; following Equations 7-17 (see section 2.1.3) and Loginov et al. (2015). The SR indicators were encoded for support and resistance separately. The encoded features for the support (resistance) levels, denoted $Dist\_Support_{t,Type}$ ($Dist\_Resistance_{t,Type}$), get a positive value if the 1-minute lagged mid-price P_{t-1} is below (above) the support (resistance) and zero otherwise.[47] Thus, this feature encodes direction (positive or negative) and magnitude.

$$Dist\_Support_{t,Type} = \begin{cases} \left(SupportLevel1_{t,PP} - P_{t-1}\right)^{+} & if\ Type = PP_1, \\ \left(SupportLevel2_{t,PP} - P_{t-1}\right)^{+} & if\ Type = PP_2, \\ \left(SupportLevel3_{t,PP} - P_{t-1}\right)^{+} & if\ Type = PP_3, \\ \left(SupportLevel_{t,Fib60} - P_{t-1}\right)^{+} & if\ Type = Fib_{60}, \\ \left(SupportLevel_{t,Fib1440} - P_{t-1}\right)^{+} & if\ Type = Fib_{1440}, \end{cases}$$

$$Dist\_Resistance_{t,Type} = \begin{cases} \left(P_{t-1} - ResistanceLevel1_{t,PP}\right)^{+} & if\ Type = PP_1, \\ \left(P_{t-1} - ResistanceLevel2_{t,PP}\right)^{+} & if\ Type = PP_2, \\ \left(P_{t-1} - ResistanceLevel3_{t,PP}\right)^{+} & if\ Type = PP_3, \\ \left(P_{t-1} - ResistanceLevel_{t,Fib60}\right)^{+} & if\ Type = Fib_{60}, \\ \left(P_{t-1} - ResistanceLevel_{t,Fib1440}\right)^{+} & if\ Type = Fib_{1440}. \end{cases}$$

[47] In the SR Equations, $(\cdot)^{+}$ denotes a maximum function that returns the greater value between its input and 0.

The pivot point SR levels were set at three levels as outlined in Equations 7 to 17, computed from the log-return lows and highs of the previous trading session. Fibonacci-ratios were implemented (based on the golden ratio ≈ 1.618) as described in section 2.1.3. To gauge the reference low and high prices, intervals of previous mid-prices were set at one hour (k=60) and one day (k=1440).

Interest differential features were generated based on the par yields collected from both the Eurozone and United States central banks. Interest differentials were motivated by IFE though the literature review indicated weak empirical results and formalized as:

$$\text{Inflation Differential}_{\text{t,maturity}} = \text{interest rate}^{US}_{\text{t,maturity}} - \text{interest rate}^{EUR}_{\text{t,maturity}},$$

where features were created with $maturity$ of 3 or 6 months, or 1, 2, 3, 5, 7, 10, 20, or 30 years.

Moreover, inspired by both PPE theory and the findings of Sosvilla-Rivero and García (2005, 374-376) concerning the impact of interest expectations on the EUR/USD, an inflation differential feature was created. Staying true to both these sources, the inflation differential was formalized as:

$$\text{Inflation Differential}_{\text{t}} = \pi_t^{(e,US)} - \pi_t^{(e,EUR)}.$$

$\pi_t^{(e,US)}$ and $\pi_t^{(e,EUR)}$ denote the 10year breakeven inflation for the United States and EU economies respectively. This feature was computed with daily frequency given frequency of the breakeven data.

4.1.2 Structured Inputs: *Labels, Feature Selection, Scaling, and Missing Values*

The purpose of the structured dataset was to provide labels and structured inputs for the ML models implemented in this thesis. Classification labels were used in prediction tasks for hybrid and sentiment models (see 4.3.2). The selected ANN models (see 4.3.1) were trained on continues valued return labels for a regression prediction task.[48] To align labels with the 5-minute evaluation windows selected for this thesis, forward-looking or future 5-minute log returns for each instance t were produced as a regression label. The computation of this label was carried out formally as,

$$r_{t,5min} = \ln(P_{t+5}) - \ln(P_t).$$

[48] After preliminary empirical testing using a variety of binary CE-loss, weighted CE-loss, and focal-loss, based models, MSE-loss regression models' directional accuracy (DA) appeared to consistently slightly out-perform the tested binary classification models by accuracy and precision.

To then convert the 5-minute log-returns into a binary directional label for the hybrid prediction models, the directional classification label was computed simply as:

$$LogReturnDirection_{5min} = \begin{cases} USD\ Depreciation, & if\ r_{t,5min} > 0, \\ USD\ Appreciation, & if\ r_{t,5min} < 0. \end{cases} \quad (29)$$

This classification label can be interpreted as the local depreciation or appreciation of the EUR/USD within the 5-minute period. The counts of USD depreciation and appreciation were balanced. The descriptive statistics of the intraday regression and classification labels are reported in Table 5 below.

Table 5: Descriptive Statistics of Test Data Label Variables for ANNs

Test Window	**Q1 2022**	**Q2 2022**	**Q3 2022**	**Q4 2022**	**Q1 2023**	**Q2 2023**
Instances	82'997	84'575	85'795	85'309	82'788	80'227
Mean $r_{t,5min}$ $(\times 10^{-6})$	-1.64	-3.36	-3.76	5.17	0.77	0.40
Standard Deviation $r_{t,5min}$ $(\times 10^{-6})$	349.63	372.43	459.02	466.73	356.13	270.90
Proportion of USD Depreciations	0.4937	0.4927	0.4933	0.5057	0.5041	0.4930
Train and Validation Data	**1.1.2021-31.12.2021**	**1.4.2021-31.3.2022**	**1.7.2021-30.6.2022**	**1.10.2021-30.9.2022**	**1.1.2022-31.12.2022**	**1.4.2022-31.3.2023**
Instances	321'227	323'472	327'701	334'528	338'676	338'467
Mean $r_{t,5min}$ $(\times 10^{-6})$	-1.14	-0.90	-1.91	-2.49	-0.89	-0.30
Standard Deviation $r_{t,5min}$ $(\times 10^{-6})$	226.05	259.68	300.39	364.35	415.76	417.14
Proportion of USD Depreciations	0.5009	0.4991	0.4957	0.4949	0.4965	0.4989

Test Window	**Q3 2023**	**Q4 2023**	**Q1 2024**	**Q2 2024**	**Q3 2024**	**Q4 2024**
Instances	80'237	80'239	77'097	76'924	79'914	81'698
Mean $r_{t,5min}$ $(\times 10^{-6})$	-1.98	2.74	-1.51	-0.22	2.19	-4.42
Standard Deviation $r_{t,5min}$ $(\times 10^{-6})$	275.22	280.03	233.62	232.38	226.35	301.20
Proportion of USD Depreciations	0.4971	0.5077	0.5037	0.5026	0.5132	0.4934
Train and Validation Data	**1.7.2022-30.6.2023**	**1.10.2022-30.9.2023**	**1.1.2023-31.12.2023**	**1.4.2023-31.3.2024**	**1.7.2023-30.6.2024**	**1.10.2023-30.9.2024**
Instances	334'119	328'561	323'491	317'800	314'497	314'174
Mean $r_{t,5min}$ $(\times 10^{-6})$	0.64	1.15	0.48	-0.08	-0.23	0.83
Standard Deviation $r_{t,5min}$ $(\times 10^{-6})$	398.47	353.47	298.16	265.88	256.75	244.29
Proportion of USD Depreciations	0.5021	0.5037	0.5040	0.5039	0.5027	0.5069

Regarding test data (FOMC releases in 2022–2024), for sentiment and hybrid models, USD depreciations account for 11 of 24 statement releases and 15 of 24 minutes releases. In the training sample (2004–2022), the equivalent shares are 82 of 153 and 77 of 145, respectively.

The features of the structured dataset used as inputs for predicting the EUR/USD were not optimal for implementation in their raw form. Thus, three preparation tasks were conducted on the structured dataset. These steps were (1) feature selection, (2) feature scaling, and (3) handling missing values.

First the set of input data from the dataset was selected such that its measurement interval was at least as frequent as the label. For example, a binary prediction model trained on macroeconomic data predicting the coming 5-minute log-return every minute would be trained solely on data sampled at a minute frequency. Thus, daily measurements of interest rates and inflation rates would not be included in its training data. However, daily measurements were accepted as input features for a model predicting the direction of the EUR/USD immediately after a FOMC release.

Following the literature explored in 3.1.4, all feature and label inputs were scaled to domain [0,1] with a Min-Max scaler. Min-Max scaling is a form of normalization achieved by transforming a variable $x_{t,i}$ to a scaled variable $z_{t,i}$ such that (Escudero et al. 2021, 6),

$$z_{t,i} = \frac{x_{t,i} - \min(\boldsymbol{x}_i)}{\max(\boldsymbol{x}_i) - \min(\boldsymbol{x}_i)}.$$

The parameters of the Min-Max scaler (the min and max values) are obtained from the training data. In practice, the parameterized scaler was applied to training, validation, and test data inputs so that validation and test data were also scaled by the minimum and maximum values of the training data.

To handle the missing values surrounding the release events, backfill was used when missing observations would have drastically lowered the available observations. Backfilling is simple procedure were missing values are filled in by earlier observations. For purposes of this thesis backfilling was limited to a maximum of 120 minutes, meaning that a missing value could only be imputed with a value of the same feature with a timestamp no more than 2 hours earlier. Backfilling was conducted around the FOMC pronouncement release events for testing model performances around the release events. For model training, only complete instances with no missing values where used. All instances that remained missing after backfilling were omitted.

4.2 Engineering NLP Sentiment Features from the Unstructured Dataset

In order to create reasonably informed sentiment models to fulfill the second research objective, reliable sentiment features need to be designed and constructed. Subsection 4.2.1 delves into the text data, and 4.2.2 outlines the data preparation and the NLP sentiment feature engineering conducted.

4.2.1 The Unstructured Dataset: *FOMC Statements and Minutes*

The text data of the FOMC minutes and statement releases constitute the unstructured dataset used by the STM and BERT model in this thesis. These documents were gathered from the Fed's (2024) website with the BeautifulSoup package in Python by scraping text data with the associated release- and meeting dates. Altogether 441 FOMC documents were collected concerning FOMC meetings between the 2nd of February 2000 and the 18th of December 2024. All were used for training the STMs. Releases before 2004 and the 16 minutes releases discussed in 4.1.1 could not be joined to the hybrid dataset. The descriptive statistics of all FOMC documents are presented in Table 6.

Table 6: FOMC Documents Descriptive Statistics (with line breaks and excess whitespace removed)

DOCUMENTS	STATEMENTS	MINUTES	TOTAL
AVERAGE TOKEN[49] COUNT	451.00	7967.57	4354.16
STANDARD DEV. TOKEN COUNT	186.43	259.52	4237.46
AVERAGE CHARACTER COUNT	2 666.21	48 114.78	26 266.49
STANDARD DEV. CHARACTER COUNT	1 207.02	16 255.39	25 581.62
UNIQUE TOKENS	1998	12033	12100
AVERAGE LEXICAL DIVERSITY[50]	0.66	0.26	0.46
AVERAGE FLESCH-KINCAID READABILITY SCORE[51]	13.02	16.10	14.62
AVERAGE SMOG READABILITY SCORE[52]	14.95	17.45	16.25
AVERAGE DOCUMENTS PER YEAR	8.48	9.16	17.64
MAX DOCUMENTS PER YEAR	11	14	25
MIN DOCUMENTS PER YEAR	8	7	15
NUMBER OF DOCUMENTS	212	229	441

As highlighted in section 2.2.1, the FOMC statements are more concise than minutes, but their relative lengths vary substantially. FOMC statements condense a lot of information on the committee's decisions, and the variance of statements' length reflect the unpredictable backdrop of the FOMC's decision-making environment. Interestingly, the language of statements seems easier to read than for minutes, as measured by the two readability scores presented in Table 6. Notably, statements have

[49] Tokens include all words, URLs, numbers, and symbols in the text documents. Here terms are limited to words.
[50] Lexical Diversity is proxied by the type-token ratio (TTR), which is the portion of unique terms or tokens used in a document formally: $TTR = |V_d| / N_d$, where $|V_d|$ denotes the number of unique terms in a document and N_d total terms.
[51] A readability metric where a lower score indicates a text is easier to read and a zero is more difficult. This metric provides a grade level readability measure, where a score 3 means that the text is readable by 3rd graders in the U.S. The metric considers longer words and sentences more difficult to read and is formalized as follows,
Flesch Kincaid Readability Score $= 0.39\left(\frac{words}{sentences}\right) + 11.8\left(\frac{syllables}{words}\right) - 15.59$. (Vyshnevskyi et al. 2024, 4.)
[52] SMOG: the Simple Measure of Gobbledygook measures readability, provides a U.S. reading-level score like the Flesch-Kincaid measure. SMOG focuses on word complexity, formally: $\text{SMOG} = 1.043\sqrt{n3}\frac{30}{sentences} + 3.1291$, where n3 denotes the number of words with more than three syllables. (Vyshnevskyi et al. 2024, 5.)

become increasingly more complex by the Flesch-Kincaid readability score -this phenomenon is described by Hernández-Murillo et al. (2014) for example. Also, as statements cannot expand on their subjects, the variability of words used is higher than that of minutes.

Statements are similarly structured. The six-feature structure (St. Louis Fed 2019) appears to hold for most statements, especially for those released after August 2003, with small differences regarding the future guidance (Lunsford 2020, 2930-2931). Minutes are also quite standardized and generally cover topics such as, the list of participants, recent financial developments, analyses and opinions on them and current political issues (Aattouchi & Kerroum 2022, 125). Additionally, all eight scheduled FOMC meetings produce a statement, but for emergency meetings, only a minutes document might be released. Thus, the average number of annual minutes releases is higher than those of statements.

In aggregate, the cosine similarity[53] of statements is also notably higher than that of minutes yet less consistent. This might stem from the brevity of the FOMC statement or because minutes have been produced since the sixties (Kansas City Federal Reserve 2016). Minutes have had time to evolve and gain structure unlike statements which experienced minor changes throughout the sample period (see e.g. Lunsford 2020). Cosine similarity of the documents appears to have increased over time as depicted on the document wise cosine similarity matrices and histograms presented in Figure 13. Breaks in cosine similarity occurred during crises, which is evident especially for statements. The distribution of cosine similarity between all FOMC minutes is relatively flat and centers around 0.78, while for statements the distribution has two distinct peaks around 0.45 and 0.72. Both documents are usually most similar to last released document of its kind. Minutes' cosine similarity exhibits a structural break at the beginning of each year. FOMC membership, appointments, and policy framework are discussed annually during the first meeting of the year. This might cause the first minutes of the year to exhibit higher cosine similarity between each other and lesser similarity when compared with near releases (see section 2.2.1; Figure 13).

[53] Cosine similarity quantifies document similarity with the cosine of the angle between their term-frequency vectors $\boldsymbol{r}$. The rows of Quanteda (CRAN 2015) package's DFM was used to map each document $d \in D$ to a vector $\boldsymbol{r} \in \mathbb{R}^{|V|}$, where $|V|$ is defined as the number of unique terms in corpus D. Consequently, cosine similarity is formally defined as: $Sim(d_i, d_j)^{COS} = \frac{\boldsymbol{r}_j \cdot \boldsymbol{r}_i}{\|\boldsymbol{r}_i\| \times \|\boldsymbol{r}_j\|}$ (Mohana & Suriakala 2018, 43-45; Sohangir & Wang 2017, 4.), $\|\cdot\|$, denotes Euclidian norm.

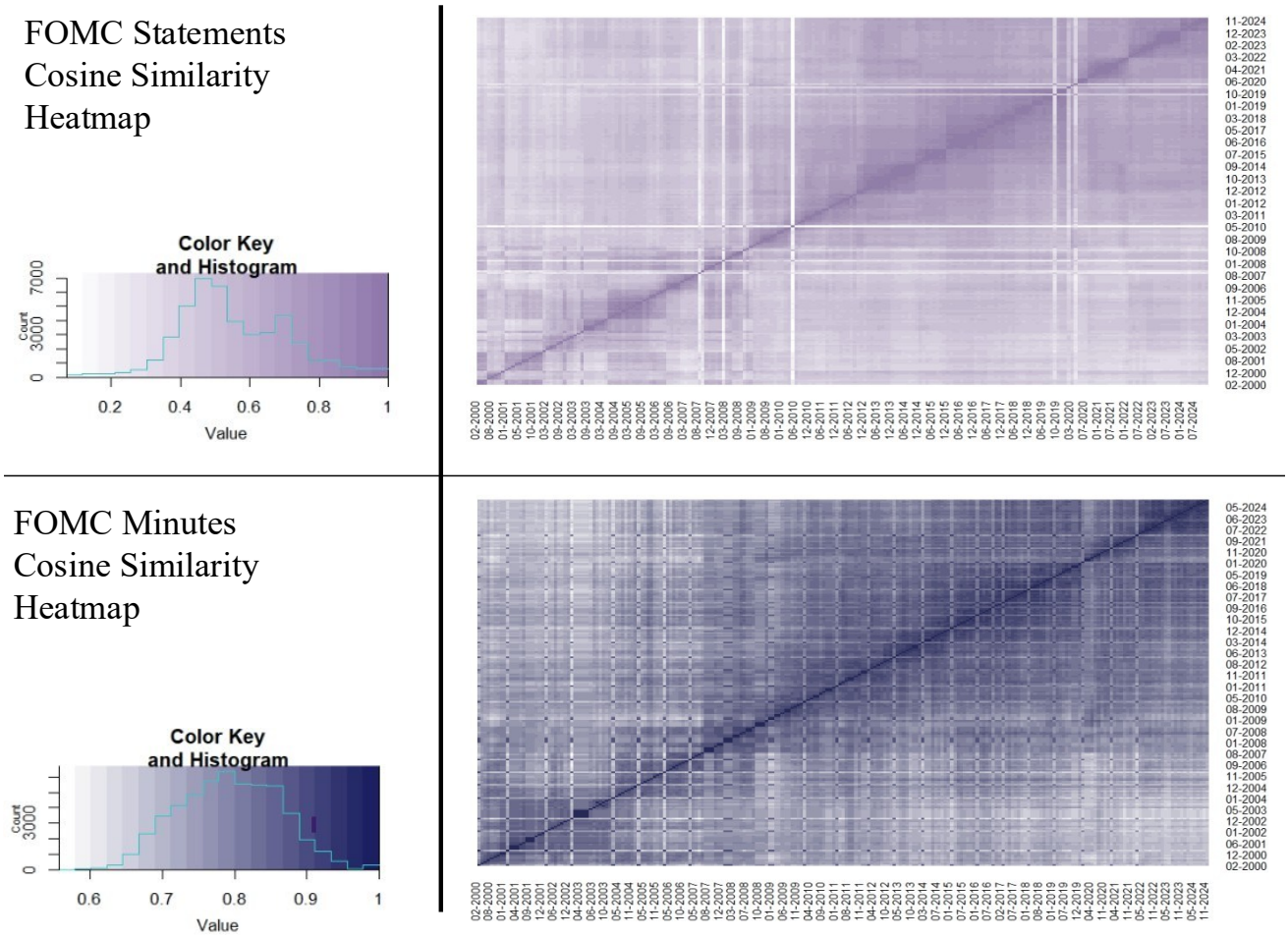

Figure 13: Distributions of Cosine Similarity, Minutes & Statements

After going through a handful of randomly selected FOMC minutes and statements, two notable contrast points arise. (1) In-depth accounts about the FOMC's views on the international financial and trade environment are discussed in minutes, but not in FOMC statements. For example, recently, FOMC minutes concerning the 2025 March 19th meeting discussed, in a couple paragraphs, the effects tariffs by the Trump administration on the global capital rotation away from United States equities and dollar denominated assets and how this effected USD currency pairs[54] (Federal Reserve 2025c). The statement concerning the same meeting makes no mention of this phenomenon, focusing on economic activity, policy stance and implementation instead (Federal Reserve 2025d). (2) Unlike statements, minutes routinely mention the monetary policy stances of several other central banks such as the ECB, Swiss central bank, bank of Canada, and others (Federal Reserve 2024).

[54] Though this release is outside the unstructured dataset, it highlights the content disparity particularly well.

Furthermore, the most frequent stop-word[55] filtered lemmas, that is words in their dictionary form, observed in the statements and minutes were quite similar. If words clearly pertaining to the FOMC they were omitted (i.e. federal, market, and committee) the most frequent lemmas appeared to diverge slightly. Both document types shared the highly prevalent lemmas: inflation, economic, rate, policy, continue, price, and remain. Yet statements disproportionately featured lemmas: *fund*, *security*, and *condition* compared to minutes, while minutes disproportionately featured lemmas: *bank*, *growth*, and *financial* compared to statements. Figure 14 presents word clouds for both statements and minutes, where the most common words are represented with larger letters. The word clouds were limited to the top 120 most frequent words for their respective document type.

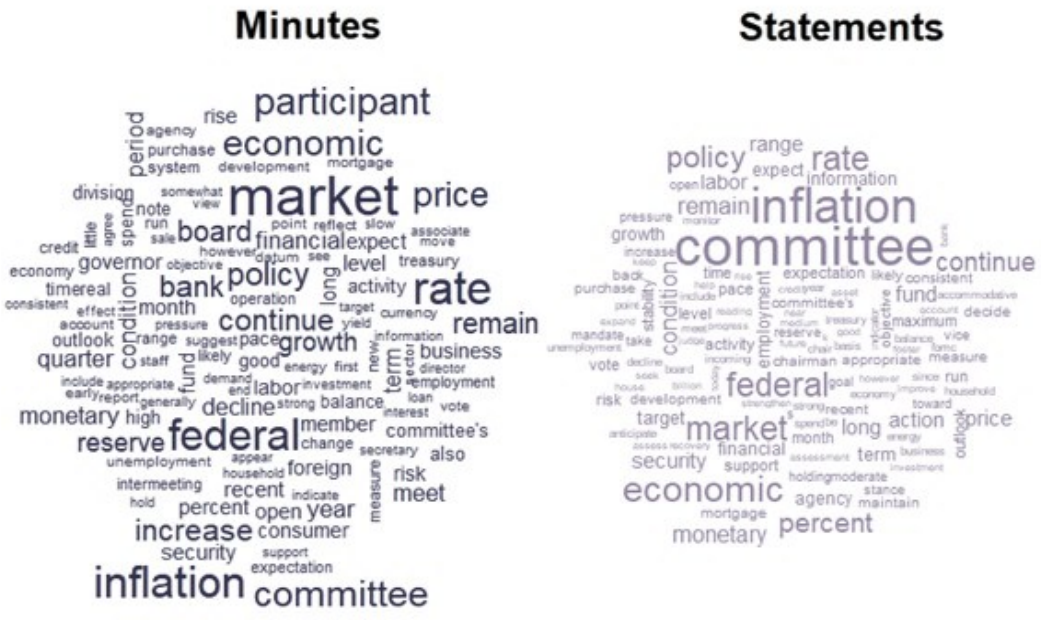

Figure 14: Word clouds by Document Type

4.2.2 Encoding Sentiment from Fed Speak with STM and BERT

Following Hansen and McMahone (2016), Jegadeesh and Wu (2017), and Aattouchi and Kerroum (2022), this thesis employed topic modelling coupled with sentiment analysis to create a better-informed sentiment score for the topics of interest. This thesis builds on these two papers with the more complex topic model STM for topic discovery and the FinBERT-FOMC for analyzing the sentiment of the discovered topics. Using a scoring technique adapted from Jegadeesh and Wu (2017,

[55] Stop-words are words that are filtered out, as they are deemed inconsequential, such as articles (e.g. *a*, *an*, and *the*).

16-17), topic-sentiments are computed for each release. Additionally, following the work of Shah et al. (2023, 6670) a *hawkishness* score adapted from Equation 28, was also calculated for each release.

The STM was used to generate topic loadings as inputs for EUR/USD predictors per document type. In practice, the STM created predictor inputs are topic distributions $\boldsymbol{\theta}_d$ assigned by the model for each unseen document in the testing data. Two STMs were trained for the testing windows, one per FOMC document type. The training data was gathered from releases between 2000 and 2021.

The inputs for the STMs were corpuses consisting of 212 FOMC statements or 229 minutes. To preprocess the raw FOMC text document, first, all document text was set to lower-case and tokenized into so called unigrams, or one-word terms (as in Ristolainen et al. 2024), to capture the word-level nuance of FOMC language. Second, these tokens were cleaned of special characters, punctuation, numbers, names,[56] URLs, and English stop-words (e.g. *very, the, is, in, have, from, and,* and *about*). Third, all tokens related to temporally local events, such as Covid19 and Russia's invasion of Ukraine, were removed to facilitate formation of general and timeless topics. Fourth, the remaining tokens were lemmatized[57] using Quanteda's[58] built-in lemma-mappings. Fifth, the processed and lemmatized tokens were collected into their own respective document-feature matrices (DFM) by document type: statement or minutes. An example of pre-processing steps 1,2,4 applied to a FOMC statement released on the 10th of June 2007, amidst the global financial crisis, is presented in Figure 15 below.

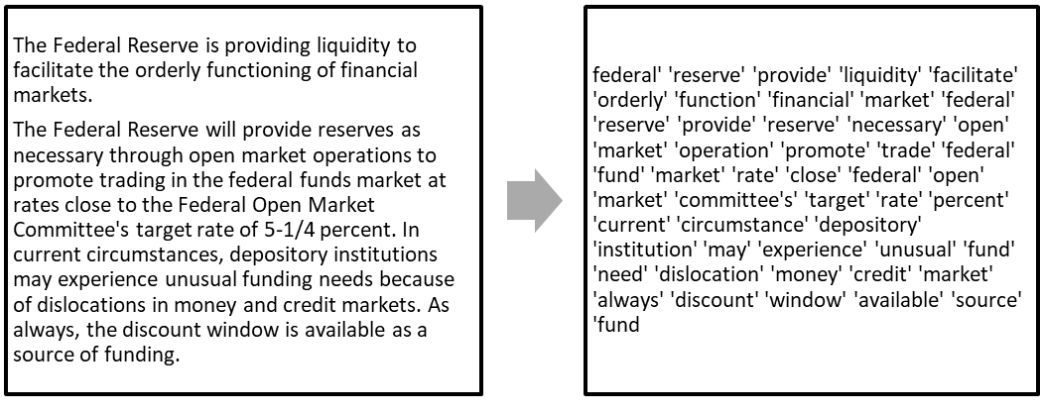

Figure 15: Pre-Processing of FOMC Documents for STMs

[56] Names were removed in a semi-automated fashion by: 1st removing a list of names recognized by the *en_core_web_sm* language model from spaCyr; 2nd manually going through and removing remaining names found (See Appendix 4).
[57] In practice lemmatization converts words to their base forms, for example words: "tax", "taxing", "taxed", "taxed", and "taxation" are all converted to their lemma or base form, "tax" (CRAN 2015, 39).
[58] Quanteda is a R package for managing and analyzing text data.

Special attention was given to the number of topics *K* in training, as choosing the appropriate number topics *K* is central for successful modelling (Blei & Lafferty 2009, 11). Blei[59] and Lafferty (2009, 11-12) advocate for using either CV to search for the optimal hyperparameter value *K,* or relying on outside sources that can specify *K.* In the extant body of research, studies have implemented LDA-based topic models on FOMC statements with 15 topics (Hansen & McMahone 2016, 116), and on FOMC minutes with 8 topics (Jegadeesh & Wu 2017, 39; Aattouchi & Kerroum 2022, 129).

Common criteria for evaluating the fit of the topic models are semantic coherence and exclusivity, as topics which are both coherent and exclusive are probably more useful, however, even with today's efficient and descriptive measures, human judgement is irreplaceable (see e.g. Mimno et al. 2011; Roberts et al. 2014; Roberts et al. 2019; Ristolainen et al. 2024). Exclusivity of the terms that constitute topics were evaluated by raw exclusivity and FREX (a term exclusivity and frequency measure), and the coherence or cogency of topics was evaluated by semantic coherence. Human judgment[60] was also applied to select the optimal number of topics for the STMs. Semantic coherence is defined following Mimno et al. (2011, 265). Semantic coherence is built on the assumption that words belonging to a single concept tend to co-occur. For nonsensical topics, only a few words might co-occur (Roberts et al. 2019). According to Mimno et al. (2011), Semantic coherence correlates well with human judgement. Semantic coherence of a topic can be computed as,

$$SemCo_k = \sum_{m=2}^{M} \sum_{i=1}^{m-1} \ln \frac{D\left(v_{i,k}, v_{m,k}\right) + 1}{D\left(v_i^{(k)}\right)}, \qquad v_k \in V_k,$$

where, $D(v)$ is the number of documents with term v_k drawn from a vocabulary V_k for topic k of M terms. For the summation operation, V_k is ordered from least to most probable by topic-term distribution terms. Thus, as m increases the number of term co-occurrences $D\left(v_{i,k}, v_{m,k}\right)$ is evaluated over a more comprehensive set of terms for increasingly *important* constituent terms of the select topic. Therefore, semantic coherence increases as the most probable terms in a topic co-occur more frequently. Importantly, Roberts et al. (2019, 10) noted that high semantic coherence can be attained with models with few topics dominated by common terms, thus additional measures are needed.

Raw exclusivity, hereafter exclusivity, is the relative usage rate of a term within a topic relative to a set of comparison topics. Bischof and Airoldi (2012, 201) state that a nonexclusive term in a document

[59] Blei originally proposed the LDA topic model framework in Blei et al. (2003)
[60] Judgement was provided by the author: a master's degree student with a B.Sc. in Business Administration.

will probably contain less topic-specific information. Thus, if a topic consists of terms which appear exclusive to it, it could indicate a more distinct topic. Exclusivity usually is formalized as follows,

$$\phi_{k,v} = \frac{\beta_{k,v}}{\sum_{j=1}^{K} \beta_{j,v}}.$$

FREX as adapted by Roberts et al. (2019, 11), is a composite measure based on Bischof and Airoldi (2012; 2016). FREX gauges both exclusivity and frequency of a select term v in a topic k. FREX is defined as the weighted harmonic mean of a term's rank by exclusivity and frequency, formalized as,

$$FREX_{k,v} = \left(\frac{0.7}{\hat{F}(\phi_{k,v})} + \frac{0.3}{\hat{F}(\beta_{k,v})} \right)^{-1},$$

where $\hat{F}$ denotes the empirical cumulative distribution function, and $\beta_{k,v}$ denotes the topic-term distribution as modelled by the STM (Roberts et al. 2019, 11). Also, $\phi_{k,v}$ represents exclusivity of term v in topic k, while $\beta_{k,v}$ represents this term's frequency (Bischof & Airoldi 2012, 203).

Employing both semantic coherence and exclusivity, the relative performances of different numbers of topics K were evaluated. Initially, both exclusivity and semantic coherence were aggregated by averaging over all topics and documents to provide an overview of STM performances by K. FREX was used to assess the cogency of topics by examining terms with the highest FREX. Generally, when applying human judgement, special attention was given to the words associated with topics. The most prevalent words for a topic (gauged by term frequency and FREX) needed to appear coherent and needed to sensibly characterize the topic to produce a reasonable label for the topic itself. For practical reasons, the characterizing words representing their topics had to capture meaningful phenomena related to FX markets or U.S. economy for a topic to be deemed of interest and then implemented for sentiment analysis. For instance, the macroeconomic environment, implemented monetary policy, future monetary policy, exchange rates and trade would be acceptable topics to proceed with.

With an appropriate STM selected the sentiment of the topic text was calculated using an approach similar to Jegadeesh and Wu (2017, 16) to gauge aggregate financial-economic sentiment. Specifically, per document for every topic of interest k, a score is computed as a sum over all paragraphs on the products of each topic loading and that paragraphs sentiment, formally,

$$\begin{aligned} Score_{\mathrm{Doc},k} &= \sum_{d \in \mathrm{Doc}} \theta_{d,k} \omega_d \ \in (-1,1), \\ \omega_d &= (1 \quad 0 \quad -1)\hat{\boldsymbol{p}}_d, \\ \hat{\boldsymbol{p}}_d &= (\hat{p}_{d,\mathrm{Positive}} \quad \hat{p}_{d,\mathrm{Neutral}} \quad \hat{p}_{d,\mathrm{Negative}})'. \end{aligned} \tag{30}$$

$\widehat{\boldsymbol{p}}_d$ denotes the soft-max normalized sentiment score outputs produced by FinBERT-FOMC for each paragraph. The model outputs a vector of scores $\widehat{\boldsymbol{p}}_d$ containing scores for the Positive, Negative, Neutral classifications. Thus, the dot product ω_d in Equation 30 indicates how positive the financial-economic sentiment was for paragraph d. The paragraph sentiment ω_d was then assigned to the topics via topic loadings $\theta_{d,k}$ observed for each topic k in the paragraph. In the spirit of Hansen and McMahone (2016, 117), economic and inflation topics were given importance, but also topics concerning the global events, and dealings between national and international banks were focused on, given the EUR/USD context of this thesis.

Additionally, using FOMC-RoBERTa (Shah et al. 2023) an adapted version of the *hawkishness* score was calculated on the paragraph level also. This meant that the number of texts evaluated per document was lower, thus, such sentiment was analyzed at a more aggregate level. Formally, the expression remained much the same,

$$Hawkishness_{\text{Doc}} = \frac{\#Hawkish_{\text{Doc}} - \#Dovish_{\text{Doc}}}{\#Hawkish_{\text{Doc}} + \#Dovish_{\text{Doc}} + \#Neutral_{\text{Doc}}}.$$

Unlike in Shah et al. 2023, for this thesis, the number of paragraphs per document of hawkish, dovish or neutral classification by FOMC-RoBERTa was evaluated instead of the number of sentences.[61]

Notably, the context window, or maximum number of tokens the two BERT models could take as input, was 512 (see Devlin et al. 2019, 8). Therefore, when the context window was exceeded by a paragraph, the paragraph was split with a sliding window or chunked into overlapping pieces of text of 512 tokens. The sliding windows for long paragraphs were constructed with one-word strides. For example, if a paragraph was 514 tokens, the first window extended from token 1 to token 512, the next window from 2 to 513, and the last window from 3 to 514. The average of the output vectors $\widehat{\boldsymbol{p}}_{window}$ scores per category were averaged to create a proxy for $\widehat{\boldsymbol{p}}_d$. Of course, this meant bearing the risk that contextually relevant information was not captured within the windows for analysis.

Additionally, the two BERT models did not require preprocessing, which besides might have disrupted the tokenization process. Therefore, the original paragraphs were indexed per each document and globally, to match the STM generated documents loadings, which required preprocessing, to the BERT sentiment scores. Thus, the text data of the paragraphs evaluated by the STMs and BERTs were essentially the same, but the STMs got fitted on the preprocessed text.

[61] BERT models were applied to perform inference on the unstructured data with Python via the Transformers library.

4.3 Training and Evaluating FX Prediction Models for the EUR/USD

This section outlines how the EUR/USD predictor ML models were implemented and tested on the data just discussed in 4.1 and 4.2. In this study different types of predictors are employed to answer the research question and create models outlined in the research objectives introduced in section 1.2. First, subsection 4.3.1 explains the role, and training procedures for the intraday ANNs. Next, subsection 4.3.2 delves into the training of sentiment models and the development of two hybrid models. Finally, subsection 4.3.3 briefly motivates the choice of performance measurement tools used to evaluate the models and help answer the research question.

4.3.1 General Intraday Regression ANNs

To benchmark the sentiment and hybrid models against ANN predictors, three different ANNs trained on different parts of the structured dataset were constructed, trained and evaluated. Combined with the sentiment models, constructed classifications from theses ANNs served also as inputs for the hybrid models. Three types of ANNs were trained on three distinct subsets of the structured dataset. The first referred to as MM2023, was based on the CNN-LSTM encoder-decoder design outlined by Markova (2023). The tested *MM2023* models were trained solely on the EUR/USD volume, high, low, open, and close data for both bid and ask prices. The second and third type of models were inspired by the work of Yıldırım et al. 2021. Specifically, these types of models were trained on either macro or technical indicators and are referred to as *Macro* and *Tech* respectively.

All three models were created as regression models trained to predict the next 5-minute log returns of the EUR/USD. Though their primary task was regression, DA of these models was evaluated, as the information encoded regarding directionality would be passed down to the hybrid model. Regression targets instead of classification targets where selected to ensure that the high performing architecture provided in Markova (2023) might serve as a relevant foundation. Thus, model outputs were regression estimates for the future 5-minute return, formally,

$$G: \boldsymbol{X}_t \rightarrow \hat{r}_{t,5min}, \qquad r_{t,5min} = \ln(P_{t+5}) - \ln(P_t), \tag{31}$$

where, $\boldsymbol{X}^t$ denotes the input variables at time t, and G the ANN regression model. The classification predictions for these ANN regression models were computed from log-return predictions as follows:

$$\hat{c}_t = \begin{cases} USD\ Depreciation, & \hat{r}_{t,5min} \geq \text{threshold}, \\ USD\ Appreciation, & \hat{r}_{t,5min} < \text{threshold}. \end{cases} \tag{32}$$

In the above, c_t denotes the true classification label and $\hat{c}_t$ a prediction constructed from the regression output. The threshold was selected simply as 0.[62]

Drawing on Markova (2023) all three ANN types used some variant structure of CNN encoder fed forward to a LSTM decoder, inspired by the comparative results w.r.t previous research presented in Table 2 (see 3.1.4). To leverage the CNN and LSTM time series capabilities, inputs were formatted as tensors of shape $(N - T + 1,\ T,\ D_{sub})$, where T denotes the lagged instances of features for each minutely instance which there are N in the original dataset and $N - T + 1$ in the tensorized input. The structured dataset had D features and the tensor input had a subset denoted D_{sub} for each ANN.

Also, PReLU activation functions were implemented after CNN layers to circumvent the dying ReLU problem as discussed in the paper by Markova (2023). The dying ReLU problem occurs when the ReLU-activation function stops its neuron from learning when constantly receiving below zero values from the linear transformation, therefore, outputting zero for all inputs (Lu et al. 2020). The PreLU is a non-linear activation function that helps to combat the dying ReLU problem by allowing below zero values to pass through scaled (Markova 2023, 299-300), formally,

$$\text{PreLU(x)} = \begin{cases} x, & x > 0, \\ \alpha x, & x \leq 0. \end{cases}$$

Even so, classical ReLU and Tanh activations were employed in LSTM layers of the networks when optimal. To find optimal configurations, the Keras Tuner library was heavily employed, and only the best models by MSE and classification metrics were selected for further implementation.

The models were built and trained with the Keras deep learning library. All ANN models in this study are trained with the Adam optimizer[63] as in Markova's (2023) paper. The Adam optimizer, like other stochastic gradient descent methods, employs mini-batches as one of the mechanisms to introduce randomness. At each iteration, a random subset called a batch is drawn from the training data, in this study, without replacement. Abstracting away from Adam's moment update, the batch is then used to compute the gradient and update the model parameters. Drawing on the background laid out in 3.1.2 and Prince (2023), this stochastic gradient update can be represented as,

$$\boldsymbol{w}_{iter+1} = \boldsymbol{w}_{iter} - \frac{\alpha}{|\mathcal{B}_{iter}|} \sum_{t \in \mathcal{B}_{iter}} \frac{\partial \ell_t}{\partial \boldsymbol{w}_{iter}}.$$

[62] This could have been further optimized by using the validation data to estimate the optimal threshold value.
[63] A common optimizer for training deep neural networks, based on stochastic gradient descent (Prince 2023, 88-90), which is built on top of mini-batch stochastic gradient descent, briefly discussed here.

Updates to the model are applied after each batch $\mathcal{B}_{iter}$, until all instances from the training data have been used. A complete pass through of the training data is called an *epoch*. (Prince 2023, 85.)

The ANNs were evaluated solely on the high frequency inputs of the structured dataset using all available full instances (no missing feature entries) from the 12 test windows. The data used for each window concerning the ANN models were collected as 16-month subsets of the structured data. This data was implemented to form separate training, validation, and test datasets. The test windows were constructed such that all 12 quarters between beginning of 2022 and end of 2024 were covered. To develop the models, the four previous quarters leading up to the test quarter were used. This four-quarter dataset was further split into training and validation instances with an 80/20 ratio, such that the first 80% instances constitute the training set and the last 20% instances the validation set. During years 2022, 2023, and 2024 the FOMC released eight statements and minutes each year, producing 48 points for analysis within the designated test windows.

All models were re-estimated in a rolling fashion, every quarter, with a year's worth of instances collected prior to the test windows quarter long test data. Training was informed with validation data to assess the generalizability of predictions and identify suitable epochs to stop model training. Primary early stopping methods were employed to determine when to stop training.

Early stopping based on the validation dataset was used to prevent overfitting. The procedure monitored a selected metric and stopped model training once no improvement was observed for a predefined *patience* interval. For example, if a model was configured with early stopping using a patience value of 10, then after the model had not improved for 10 consecutive epochs, training was stopped and best weights encountered before stopping w.r.t the select metric were restored.

In implementing early stopping, MAPE, MSE, and the adjusted R2 score were utilized. The *MM2023* models implemented MAPE and MSE, and due to challenges in training reliable *Macro* and *Tech* models, the adjusted R2 scoring was employed.[64] The adjusted R2-Score, hence R2, was computed with the default Keras regression metrics. The R2-Score can be formalized as (Keras 2026),

$$R^2 = 1 - \frac{\sum_{t=1}^{N}\left(r_{t,5min} - \hat{r}_{t,5min}\right)^2}{\sum_{t=1}^{N}\left(r_{t,5min} - \bar{r}_{t,5min}\right)^2},$$

where $\bar{r}_{t,5min}$ denotes the mean five-minute log-return in the selected dataset of the test windows.

[64] These decisions were made by the Author when testing models and implementing the Keras Tuner package.

The ANNs were trained with MSE as the loss function minimized by an Adam optimizer. Outputs $\hat{r}_{t,5min}$ at each FOMC release were collected in test data for benchmarking and hybrid model inputs.

To evaluate the ANN regression models' performance, primarily, three metrics were focal: MAPE, RMSE, and DA. Because constructing and training reliable *Macro* and *Tech* models was challenging, the adjusted R2 and DA scores were closely monitored. To further assess the degree of directional information encoded, precision and recall were evaluated based on the classification labeling computed following Equations 31 and 32. These metrics from ANNs and previous literature were also implemented for benchmarking the hybrid classification models. To examine the optimal thresholds for computing labels from ANN regression outputs using Equation 32, confusion matrices were employed.

Confusion matrices and classifications metrics were used to decide on slight alterations to selected features, types of layers, and parameters of the layers initially selected with Keras Tuner. Human judgement and All model configuration decisions were made based on validation data results. Using test data would have defeated the purpose of creating a prediction model, as it would have been fishing or intended data snooping (White 2000, 1997-1998). However, slight risk for unintended look-ahead bias (Yae 2024, 953) remains, as previous research, which included the experiment conducted within the test data, was reviewed before training the ANN models.

When testing training procedures for the MM2023 model, two different approaches to training were implemented, producing two predictive regression models. The first was trained for a maximum of 50 epochs with early stopping configured on validation MAPE using a patience of 10 epochs. This training procedure produced the MAPE-stopped short-budget model, hence abbreviated to MAPE-SB-MM2023. The second model trained for a maximum of 240 epochs and was configured with early stopping based on validation loss with a patience of 15 epochs. This training procedure produced the MSE-stopped long-budget model, hence abbreviated to MSE-LB-MM2023.

Save for the different early stopping algorithms, the MM2023 ANNs were identical up to the early stopping point of MAPE-SB-MM2023. The long-budget model had a longer training time budget than the short-budget model. Both models were trained with the same input EUR/USD bid and ask features. These features were the open, close, high and low prices, trading volumes for both the bid and ask prices, and the mid-price. The inputs were lagged with $T = 16$.

The Macro and Tech models were also trained with early stopping configured based on the R2 score. The Macro models' early stopping routine was configured with a patience of 15, while the Tech

models implemented early stopping with a patience of 25. The input features of the Macro model were the EUR/USD mid-price, one-hot encoded trading session indicators,[65] the log-returns of the S&P500, Brent, and WTI prices. These features were lagged with $T = 16$. Simultaneously, the Tech models were trained with all technical minute-wise input features, the Fibonacci-ratio-based SR indicators, volatility features (with the addition of TSRV), all moving average features, the mid-price, bid and ask volumes, and one-hot trading session indicators. These features were lagged $T = 22$. The final model configurations for the ANNs are summarized in Table 7.

Table 7: Regression ANN Configurations and Crucial Parameters

	MM2023	***Macro***	***Tech***
CNN	C_1=2048, $K_{size} = 6$	C_1=428, $K_{size} = 2$	C_1=512, $K_{size} = 2$
Activation Function	PReLU	PReLU	PReLU
CNN	C_2=2048, $K_{size} = 3$	C_2=428, $K_{size} = 3$	C_2=512, $K_{size} = 3$
Activation Function	PReLU	PReLU	PReLU
Max Pooling	Stride = 2	Stride = 2	Stride = 2
LSTM	256 Hidden Units	256 Hidden Units	256 Hidden Units
Activation Function		Tanh	PReLU
LSTM		32 Hidden Units	24 Hidden Units
Activation Function		PReLU	ReLU
Output Layer	Linear	Linear	Linear

As in Markova (2023), causal padding was used for all CNN layers. Additionally, various regularizations were applied based on Keras Tuner's validation data outcomes. The regularization parameters as well as some of the selected features were occasionally dropped depending on validation data outcomes when retraining models within the twelve test windows. The parameter values in Table 7 were implemented exactly as shown for the reported ANN models.

[65] An indicator based on if markets are open in Asia, Europe, the U.S., or the Witching hour occurs (see Appendix 3).

4.3.2 FOMC Release Specialized Sentiment and Hybrid Classification Models

With the structured dataset (4.1) and sentiment features (4.2) prepared, sentiment models could be developed and trained. For each of the three years in the testing windows (2022, 2023, and 2024), a logistic regression (hence LR) and a RF model were trained and evaluated on the FOMC statements and minutes, separately. This procedure resulted in six datasets and twelve models being produced. Labels for training the sentiment predictors were binary depreciation and appreciation signals as expressed in Equation 29. Input data were the topic-sentiment and hawkishness scores generated by implementing STMs and BERTs. Datasets were split into training and testing data, with model hyperparameters selected using five-fold CV. This decision was made due to the small number of available training instances.[66] To combat the low number of training instances, the previous testing data were appended to next year's models' training data. The model hyperparameters and sizes of training and test data by number of instances are summarized in Table 8.

Table 8: Hyperparameters and Data for Sentiment-Based Classifier Models

		2022	***2023***	***2024***
Statements		145/8	153/8	161/8
	LR	L2-Reg	L2-Reg	L2-Reg
	RF	MinSamples, MaxDepth	MinSamples, MaxDepth	MinSamples, MaxDepth
Minutes		145/8	153/8	161/8
	LR	L2-Reg	L2-Reg	L2-Reg
	RF	MinSamples, MaxDepth	MinSamples, MaxDepth	MinSamples, MaxDepth

Note: *the number of instances per document type are stated as training data / testing data.*

To train LR and RF models with suitable complexity for the classification task, hyperparameters were tuned using CV. The hyperparameters tuned were L2-regularization (L2-Reg) for LR, and the minimum number of samples per leaf-node (MinSamples) and maximum depth (MaxDepth) for RF. The number of trees for the RF models was set as 10'000. For both families of model, the best model as measure by AUC was selected as the final predictor model.

L2-Reg adds a quadratic value-based penalty for weights to loss function used to train the model. This shrinks the weights, but the extent of shrinkage depends on the hyperparameter λ selected. Essentially, the loss function for the LR model becomes (Park & Hastie 2008, 5),

[66] While this decision did not compromise validity of the reported results, it raised risk of overfitting sentiment models hyperparameters based on older data points.

$$\mathcal{L}(\sigma(\boldsymbol{z}); \boldsymbol{X}, \boldsymbol{c}) = -\sum_{t=1}^{N} \big(p_t \ln(\hat{p}_t) + (1 - p_t)\ln(1 - \hat{p}_t)\big) + \frac{\lambda}{2}(\boldsymbol{w}'\boldsymbol{w}).$$

Likewise, both RF hyperparameters MinSamples and MaxDepth help to control the complexity of RF by limiting overfitting by limiting how the weak learners or trees ψ_m of the forest can be grown. Minsamples force the tree-building algorithm to consider a node split only if a minimum number of training data instances will land in the leaf nodes. MaxDepth limits the maximum depth to which individual classification trees can grow. (Dael & Avvad 2025, 462, 464).

Additionally, the directional labels in the training datasets were somewhat imbalanced. This was addressed by stratifying the sampling when generating the folds of the CV to prevent partition induced dataset shift (Fontanari et al. 2022, 627). This meant that instances were selected such that each fold of the dataset had a similar representative proportion of classes as the original dataset (Arlot & Celisse 2010, 69). Furthermore, weighted loss functions were implemented in training the sentiment models on the imbalanced datasets. Following conventional approaches, a weighting factor $\alpha \in [0,1]$ was added to the loss function (Lin et al. 2018, 3). The weighting factor α was set as the inverse proportional frequency of class 1. Formally, the loss functions were updated to,

$$\tilde{\ell}_t = \begin{cases} \alpha \ell_t, & c_t = 1, \\ (1-\alpha)\ell_t, & otherwise. \end{cases}$$

Inspired by Ling et al. (2021, 6-7) and RF models, the hybrid models were built as ensemble models tallying votes from ANNs and sentiment models to make a final prediction on direction of the EUR/USD. The first hybrid was constructed as an equally weighted (EW) model. EW weighed label predictions for each model equally. Because six models were trained, EW gave each model an equal weight of 1/6. The second hybrid model weighed votes based on past performance. This model is hence referred to as PERF. PERF employed the last eight out-of-sample release prediction accuracies to weigh the models' votes for classification. For the first test year, 2022, training data accuracy was employed as out-of-sample measurements were not available for ANNs. With this approach, each model's weight was its share of previous correct classifications among all models, formally,

$$w_{t_{type},mod}^{(PERF)} = \frac{\sum_{t=0}^{t_{type}-8} TP_{t,mod} + TN_{t,mod}}{\sum_{t=0}^{t_{type}-8} \sum_{m \in M} TP_{t,m} + TN_{t,m}}.$$

In this definition, m denotes the models belonging to the set of models used M, and mod is the model to which the weight $w_{t_{type},mod}^{(PERF)}$ is assigned to by PERF. Also, t_{type} is the document release event index for document of $type \in \{Statments, Minutes\}$.

Moreover, hybrid models EW and PERF, resolved tied votes by producing an additional vote using sentiment models. This tie-breaker vote was constructed by gauging the confidence of the models' predictions by taking the square root of the product of their scores to form a new prediction, formally,

$$\hat{p}_{t,tie} = \sqrt{\hat{p}_{t,Logistic\ Regression} \times \hat{p}_{t,RF}}, \qquad \hat{c}_{t,tie} = \begin{cases} 1, & \hat{p}_{t,tie} \geq 0.5, \\ 0, & \hat{p}_{t,tie} < 0.5. \end{cases}$$

4.3.3 Binoculars to a Horserace: *Selected Metrics and Statistical Tests*

To ultimately report on and compare model performance (5.3) the accuracy metric was employed. Accuracy was selected for its prevalent use in previous FX forecasting literature as seen in 3.1.4 and other FX direction prediction papers (e.g. Guyard and Deriaz 2024). The accuracy metric expresses the number of predictions which were correct out of all predictions made by a model, formally,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}.$$

Accuracy allows the models of this study to be contrasted with other research findings. To examine the performance from the point of view of quality and reliability, precision is also reported. These metrics by themselves would not be sufficient to answer the researcher question outlined in 1.2.

Thus, to scientifically compare classification performance of EUR/USD direction predictors, multiple statistical tests were initially considered. However, given the nested-like[67] nature of hybrid models the standard Diebold-Mariano (1995) tests were forgone, with Giacomini-White (2006) tests (GW) conducted as a special case of Diebold-Mariano tests, robust to nested models. The key advantage of this test is that it handles forecast comparisons between a variety of modeling techniques (Giacomini-White 2006, 1570-1571). GW test (2006, 1553) is used to test whether one model can outperform another conditional on $\mathcal{F}_t$ via the null hypothesis,

$$H_0\colon\ \mathbb{E}[\Delta L_{t+1} \mid \mathcal{F}_t] = 0,$$

where $\mathcal{F}_t$ denotes information available and ΔL_{t+1} the difference of loss between models, $g(\cdot)$ and $f(\cdot)$ at time t. Formalized as,

$$\Delta L_{t+1} = \ell_{t+1}^{(g)} - \ell_{t+1}^{(f)}.$$

[67] Strictly speaking, hybrid models are linear combinations of most models they are compared with, not nested models.

In this study, the loss function was selected as 0/1-loss as specified in (see 3.1.2). This was necessitated due to limitations from ANNs regressions predictions. For this study the trivial information set $\mathcal{F}_t = \{\emptyset, \Omega\}$ was chosen following Nonejad (2021, 5-7), reducing the hypothesis to,

$$H_0\colon\ \mathbb{E}[\Delta L_{t+1}] = 0.$$

That can be evaluated with a test similar to a Diebold-Mariano t-test (Giacomini-White 2006, 1557),

$$t_{GM} = \frac{\Delta\bar{L}_n}{\frac{\hat{\sigma}_n}{\sqrt{n}}}, \tag{33}$$

where n is the number of out-of-sample observations, $\hat{\sigma}$ is a Newey and West (1987) Heteroscedasticity and Autocorrelation Consistent (HAC) correlation matrix estimator, and $\Delta\bar{L}_n$ denotes the mean difference in out-of-sample loss between $g(\cdot)$ and $f(\cdot)$ (Giacomini-White 2006).

To predict which of two models produce a lower mean out-of-sample loss, a useful property of the GW test (2006, 1558), $\mathbb{E}[\Delta L_{t+1}] \approx \hat{\beta} h_t$, is utilized for the following regressions (Nonejad 2021, 6),

$$\Delta L_{t+1} = \hat{\beta} + \varepsilon_{t+1}.$$

Thus, by implementing HAC, a t-test statistic as presented in Equation 33 can be evaluated to determine whether a constant coefficient $\hat{\beta}$ can explain the loss differential. Then, a positive $\hat{\beta}$ suggests that model f is superior to g. Conversely, a negative $\hat{\beta}$ suggests model g is superior to f. (Nonejad 2021, 6; Giacomini-White 2006 1558.)

As the dataset size was limited and 0/1-loss function was selected, results of GW the tests are to be interpreted as suggestive evidence of comparative predictive performance, rather than definitive proof for outperformance of one algorithm over another for the EUR/USD prediction task. Especially, as the 0/1-loss function can realize highly variable or skewed results with small datasets.

Moreover, to deal with the smaller dataset size, other tests from ML literature where considered. Inspired by Dietterich 1998 (1905-1907) and Costa et al. 2018 (1,7), modern mid-p McNemar tests were motivated to assess whether misclassifications differ significantly between models. The McNemar test is a non-parametric contingency test that can be employed to test whether two binary classifiers performance differently in the same test data. The test was advocated by Dietterich (1998, 1907) for comparing models in situations where algorithms are run once. Given the computational expense of developing ANN models, this was the case. In this study two models, $g(\cdot)$ and $f(\cdot)$, are

applied to the same FOMC release test data to generate label predictions $\hat{c}_{t,g}$and $\hat{c}_{t,f}$ for the direction of the EUR/USD. These predictions are then compared against the real label c_t to produce a contingency table as shown in Table 9. (Dietterich 1998, 1901.)

Table 9: McNemar Test Contingency Table

[A] $\#(\hat{c}_{t,g} = \hat{c}_{t,f} \neq c_t)$ Both models misclassified	[B] $\#(\hat{c}_{t,g} \neq \hat{c}_{t,f} = c_t)$ $g(\cdot)$ misclassified
[C] $\#(\hat{c}_{t,f} \neq \hat{c}_{t,g} = c_t)$ $f(\cdot)$ misclassified	[D] $\#(\hat{c}_{t,g} = \hat{c}_{t,f} = c_t)$ Both models were correct

Under the null hypothesis that the two ML algorithms perform identically in the test set, the values of cells B and C of the contingency table should be equal. Thus, with a continuity correction term applied (Edwards, 1948; Raschka 2018, 38), the test statistic in Equation 34[68] approximately follows the $\chi 2$ distribution with one degree of freedom. (Dietterich 1998, 1901.)

$$\frac{(|B - C| - 1)^2}{B + C} \sim {\chi_1}^2. \tag{34}$$

Though the test datasets for out-of-sample label predictions were small, the class predictions used for McNemar tests were gathered model-vs-model for statements and minutes separately. Moreover, following available scientific guidelines the minimum number of discordant pairs $(B + C)$ between models needs to be 25 (Sundjaja et al. 2025) before the McNemar test can be reliably conducted.[69] Otherwise, a conservative two-sided exact McNemar test (Raschka 2018, 38) should be conducted.

The exact McNemar test provides directly the p-value via an adapted binomial test, with proportion 0.5 (Raschka 2018, 38; Fagerland et al. 2013, 3). The exact McNemar test can be formalized as,

$$p_{exact} = 2\sum_{i=0}^{k} \binom{n}{i} 0.5^n, \qquad k = \min(B, C), \qquad n = B + C.$$

Yet, given the conservative nature of the exact test, it is not expected to provide robust evidence of non-significant relationships. In fact, in well cited biomedical methodological research it has been stated that they would not recommend use of the test for any situation and instead recommend using

[68] Here, $|\cdot|$ denotes absolute value.
[69] Thus, provided with test datasets of 24 observations, only the exact tests would have been run.

the McNemar test statistic without continuity correction, and advocate for another test for small samples: the mid-*p* value (See Fagerland et al. 2013, 3-4, 6-7.) The mid-*p* value is, formally,

$$\text{mid-}p = p_{exact} - \binom{n}{k} 0.5^n.$$

Consequently, mid-*p* test results were reported instead of conventional McNemar test results. This decision was cautiously encouraged by its empirical testing and recommendation by Mohammadi et al. (2019) in the context of ontological matching ML systems, and the use of mid-*p* tests for comparing prediction accuracies of models for medical research, published in Nature's Scientific Reports (Abhishek et al. 2021, 12). Additionally, Fagerland et al. (2013, 7-8) find that the mid-*p* test did not violate the nominal level of type I errors in the almost 10'000 scenarios they tested.

Additionally, to test the comparative accuracy of the RF sentiment models for the statement's compared to the minutes' release events, a simple test for difference of proportions was selected. The underlying assumptions of the data with this test are easily violated in such contexts (Dietterich 1998; Raschka 2018, 34-35), thus a significant result with this test should be considered as preliminary evidence of the statements and minutes sentiment RF models having different accuracies. Given that both datasets had 24 observations, the z-statistic for this test was computed as (Raschka 2018, 35):

$$z = \frac{Accuracy_{RF_2} - Accuracy_{RF_1}}{\sqrt{Accuracy_{RF}(1 - Accuracy_{RF})/24}} \sim N(0,1), \tag{35}$$

Where the term in the denominator is defined as,

$$Accuracy_{RF} = (Accuracy_{RF_2} + Accuracy_{RF_1})/2.$$

With the z-value used to assess the null hypothesis,

$$H_0\colon\ Accuracy_{RF_2} = Accuracy_{RF_1}.$$

5 Results

This chapter discusses the (5.1) NLP analysis used to create sentiment scores for FOMC sentiment models, (5.2) training and initial observations of performance of the two types of component models, (5.3) out-of-sample model performance results in the FOMC statements and minutes release data, and (5.4) robustness of the methods used and validity of the obtained results. All analyses and experiments seen here were conducted on a Windows laptop PC.[70]

5.1 Generating Sentiment Features: *Encoding Sentiments of the FOMC*

This section discusses results of the NLP modeling of the unstructured data set with STM and BERT on their respective datasets. 5.1.1 and 5.1.2 discuss and describe topics discovered by STMs from FOMC statements and minutes, respectively. 5.1.3 reports on topic-sentiment, and hawkishness scores.

5.1.1 Discovering Latent Topics of the FOMC Statements

The results of the FOMC statements STM topic hyperparameter (K) search are outlined in this subsection accompanied by the descriptions of selected topics employed for further analysis.

When training the STMs, special attention was placed on selecting the number of topics K. To initialize modelling, the selection of candidate values of K were evaluated with statistical criteria and personal judgement as described in section 4.2.2. Hyperparameter selection was evaluated for FOMC statements and minutes within the training data spanning from 2000 to 2021. Additionally, the STMs were trained such that each pre-processed document was broken down by paragraph, with topic loadings aggregated to back to the document level. Additionally, due to the relatively high computational expenses of modeling minutes documents compared to statements, K for the statements STM was evaluated more thoroughly with 10-fold CV using the selected statistical criteria.

[70] All experiments were implemented with Python or R languages using open-source frameworks such as Keras, STM, spaCyr, and Quanteda. Keras was implemented on a Windows laptop with an Intel i7-12700 Central Processing Unit, 16GB of Physical Random Access Memory, and a NVIDIA GeForce 4060 RTX-series Graphical Processing Unit (GPU). Keras was run in a WSL environment, with Tensorflow, and Python 12 to enable more stable backend use of the GPU.

To select the appropriate number of topics, semantic coherence and exclusivity were initially observed, and human judgement was used to further consider topics coherency, interpretability and distinctness. The number of topics was initially narrowed down via a CV analysis, considering semantic coherence and exclusivity for a base Correlated Topic Model (CTM). The CTM was otherwise the same as the final STM, but no metadata covariates were considered.[71] The available documents in the training corpus were randomly split into ten folds for CV. The semantic coherence and exclusivity for each CTM realization were computed per K. The mean and standard deviation of metrics were aggregated for each K. In this initial step, the number of topics benchmarked spanned from 3 to 30, and CV results were used to select candidate values of K to be further assessed with human judgement. The CV results for the FOMC statements STM are illustrated in Figure 16.

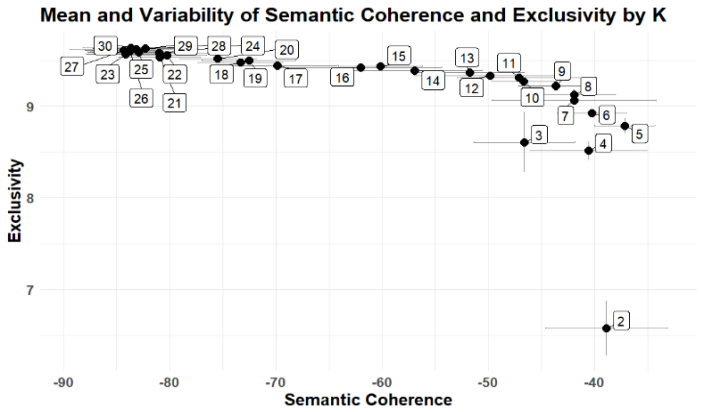

Figure 16: Topic Model Cross-Validation for Number of Statement Topics

Figure 16 is a scatterplot where the x-axis depicts semantic coherence and y-axis the exclusivity of topics across the cross-validation CTMs. The CTMs in Figure 16 were trained on subsets of the 2000-2021 FOMC statements. Each realization of K had its mean value represented by a dot, with lines of length one standard deviation extending out along the semantic coherence and exclusivity dimensions. Semantic coherence varied more per realization than exclusivity, especially after $K > 6$.

Based on the CV aggregate values, three candidates of K for the STM were selected to be further analyzed using human judgment. The first candidate value of K was chosen at the point at which by decreasing K, the standard deviation of semantic coherence last overlaps the mean value of the previous K, for example at $K = 16$ in Figure 16. The second point was selected where the mean lost semantic coherence was at least 10 times as high as the mean gained exclusivity as topics were added

[71] CTMs were selected for this initial step as the architecture is relatively similar to their successor the STM and crucially they are significantly less resource intensive to train.

for more than one added topic in row,[72] for example at $K = 5$ in Figure 16. The third point was based on previous research; 15 topics following Hansen and McMahone (2016, 116).

In the second and last step, the candidate values of K were evaluated by considering if the resultant topics, as represented by their terms, were cohesive, informative, interpretable, and distinct.[73] For FOMC statements, the best topics were formed at $K = 15$. This finding aligned with that of Hansen and McMahone (2016), who used another LDA topic model on FOMC statements.

Notably, two topics modeled by the STM directly related to international currency markets. Topic 3 contained terms relating to paragraphs that discussed swap line[74] arrangements and dollar FX, while another (Topic 15) contained terms relating to foreign pressures when taking a stance on monetary policy. The topics listed from most to least prevalent are listed in Figure 17 below. In Figure 17 the topics discovered are followed by the seven most relevant terms as measured by frequency.

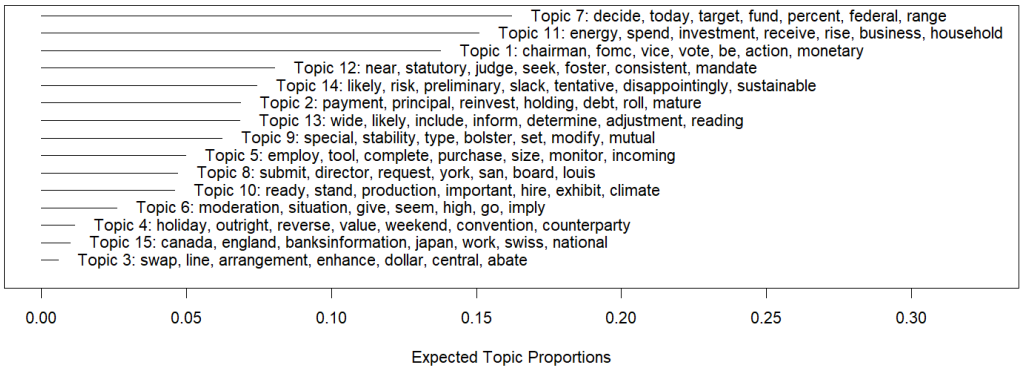

Figure 17: STM Topics from FOMC Statements

Based on further analysis, Figure 17 reveals, that the top five most common constituent topics of FOMC statements concerned: (Topic 7) setting the range for the federal funds rate, (Topic 11) elaborating on events in the real economy, (Topic 1) tallying votes for and against set actions of the meeting, (Topic 12) mentions of the dual mandate and objectives of the FOMC with reference to economic outlook, and (Topic 14) percived inflation risks and outlook. Word clouds for the top three most common topics and the international topics are shown in Figure 18 below. The size of the unigram lemmas are based on their relative topic-term parameter values $\beta_{d,k,v}$ (see 3.2.1).

[72] Essentially the 2nd point was where at least two consecutively added topics resulted in $\frac{d[Semantic\ Coherence]}{d[Exclusivity]} \leq -10$.

[73] Human judgment was facilitated by using the built-in-tools of the STM R-package such as *FindThoughts, Plot.STM, LabelTopics,* and *topicCorr* (see. Roberts et al. 2019 for descriptions of these functions).

[74] Fed Swap Lines, i.e. USD liquidity arrangements, are agreements where a foreign central bank can swap another currency for USD at set amounts and exchange rates with the New York Fed (New York Federal Reserve, 2026).

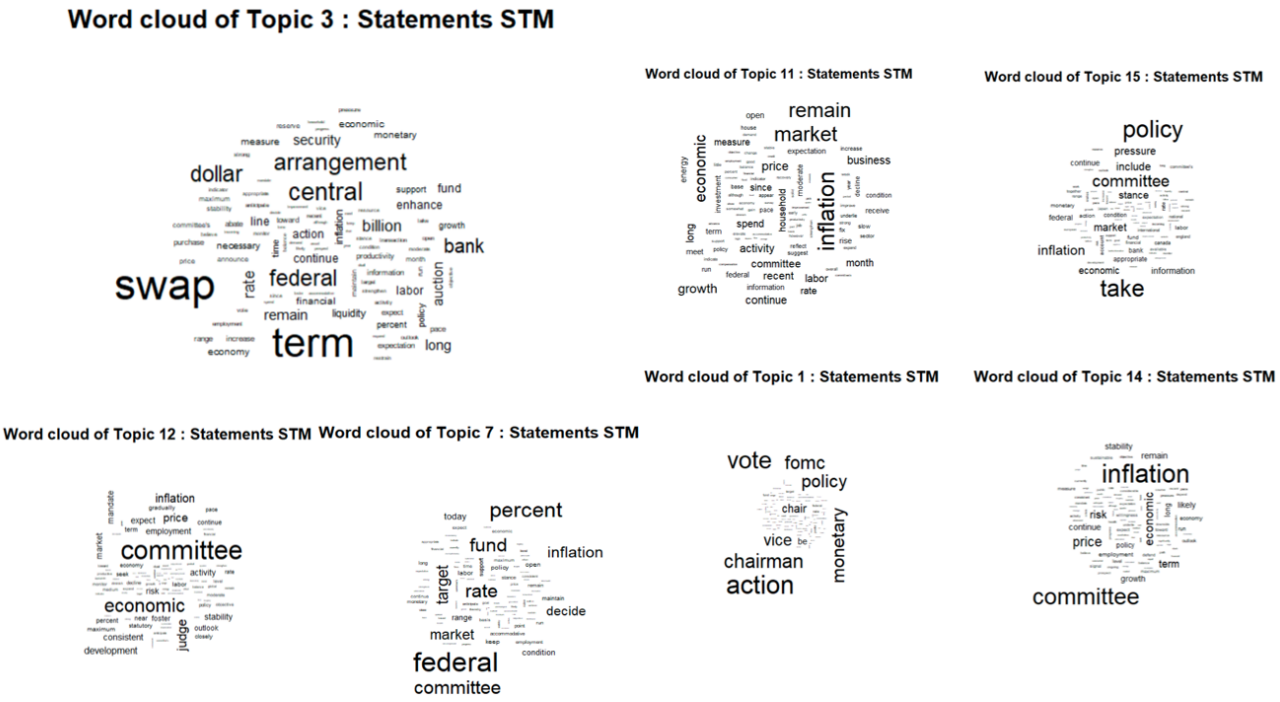

Figure 18: Topic Word Clouds from FOMC Statements STM

Three of the topics listed above and portraid in Figure 18 were deemed topics of interest, and their topic loadings $\boldsymbol{\theta}_d$ or topic distributions[75] were computed as input features for the hybrid EUR/USD predictors. The topics of interest derived from the FOMC statements for EUR/USD prediction were Topic 3 for its focus on international central bank issues, and Topics 11 and 14 for their strong emphasis on economic outlook and realities as communicated by the FOMC. Assuming the transmission of Fed's superior information (Jegadeesh & Wu 2017, 33.) to FX market participants via paragraphs of the FOMC statements which include such topics.

5.1.2 Discovering Latent Topics of the FOMC Minutes

A similar process was carried out for the FOMC minutes documents. Given the compute costs of the CTMs, only 3 to 25 topics were considered for K. Also, a single run with no document holdouts on the 2000-2021 FOMC minutes dataset was conducted. The results of these runs presented only two candidates of K to be evaluated with human judgement. The first point was selected as the point at which lost semantic coherence was at least 10 times as high as the exclusivity gained as topics were added, which indicated $K = 16$. The second point was $K = 8$ topics given the results of previous

[75] The topic distributions $\boldsymbol{\theta}_d$ in practice represent the probability of the topics in the document (see 3.2.1).

LDA-based topic model research on this document type (Jegadeesh & Wu 2017, 39; Aattouchi & Kerroum 2022, 129). The results of the preliminary CTMs are illustrated in Figure 19 below.

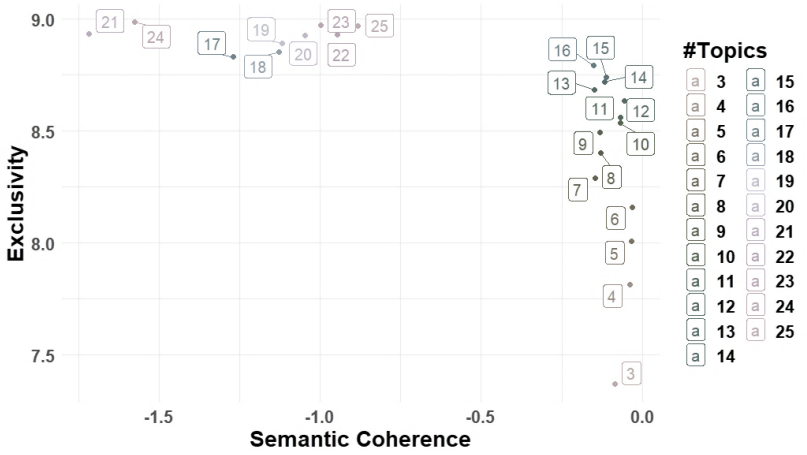

Figure 19: Preliminary CTM Results for Number of Minutes Topics

When examining the two options for K, better topics were achieved with $K = 16$. This allowed for more descriptive topics considering the broad scope of issues discussed in FOMC minutes. This perspective, but not the finding for K, aligned with previous research (Jegadeesh & Wu 2017, 39; Aattouchi & Kerroum 2022, 129). Most notably for the purposes of this thesis, two international topics were discovered. Also, the topics found by STM were more distinctive for minutes, likely due to the longer document lengths. The topics listed from most to least prevalent are listed in Figure 20.

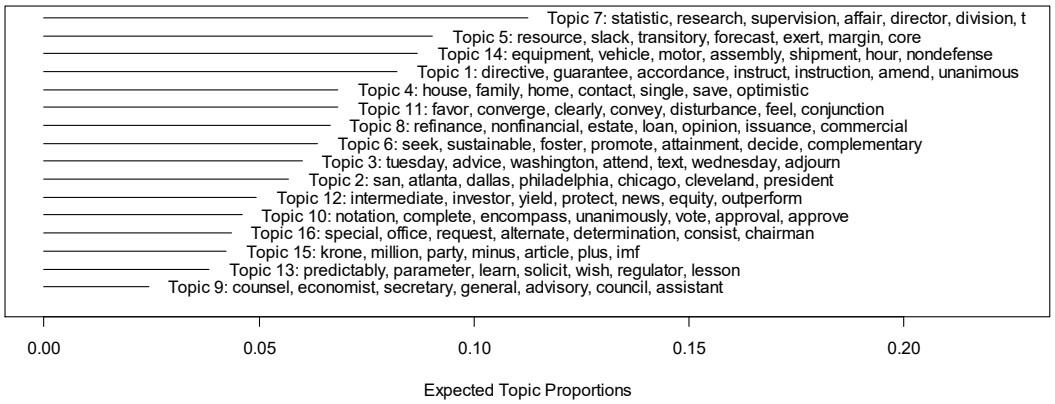

Figure 20: STM Topics from FOMC Minutes

Based on further analysis, four distinct groups of topics discussing: (1) swap lines and foreign markets, (2) U.S. inflation, (3) the domestic economy, and (4) financial markets were recognized. These groups were remarkably distinct, with the constituent topics of groups sharing more words together than with topics of other groups. At a more atomic level, strong cross-group similarities were

found between Topics 6 and 11, as both focused on policy decisions and guidance, but in light of the inflation and the economic environment, respectively. Also, many topics related to voting, discussion on economic research projects and federal matters (new FOMC members, discussions pertaining to the federal government) were identified. Figure 21 represents word clouds for some of the more cohesive topics discovered and their respective grouppings.

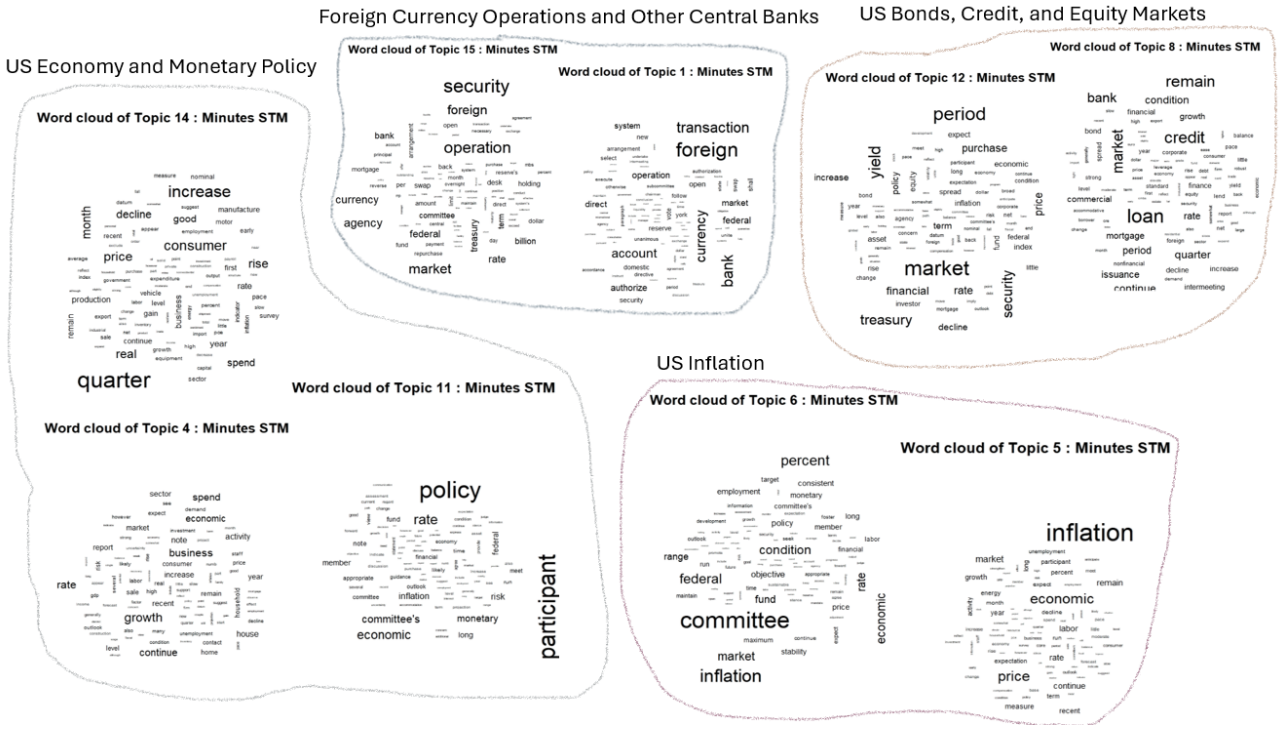

Figure 21: Topic Word Clouds from FOMC Minutes STM

Pertinently the topics of interest for predicting the EUR/USD were selected to be Topics 1, 4, 5, 8, 12, 14, and 15. These topics discussed matters related to the financial market and the economy generally and descriptively with less emphasis on policy implications. This made them appear particularly suitable for further analysis of their financial-economic sentiment by the FinBERT-FOMC model.

Notably, both the statement and minutes STMs were trained with the meeting year as a content covariate and with topic prevalence covariates constructed with natural splines parameterized by document and paragraph lengths. These time dependencies on terms via content covariates, lessened the influence of temporally local events on topic formation. To illustrate, 2001 releases had aspects related to terrorism that were associated with the year and not the topics. Also, terms related to hurricanes and lemma *Katrina* were recognized by the STM as relating strongly to the year 2005. Some temporally local events were neglected in preprocessing, yet the STMs managed to turn the

topic outcomes favorable with this configuration. All STM training and evaluation were conducted with R code and the STM package (Roberts et al. 2019). Additionally, all STMs were trained on 2000-2021 FOMC release data to ensure that topic loadings were generated fairly and out-of-sample for the test windows covering 2022, 2023, and 2024.

5.1.3 Encoding Sentiments of the FOMC with Sentiment Analyses

Subsequently, with the STMs selected and trained, topic loadings $\boldsymbol{\theta}_d$ per paragraph were generated using the STMs on all documents. The next step for sentiment analysis was to classify the paragraph-level sentiments with both FOMC-RoBERTa and FinBERT-FOMC. Two sentiment measures, hawkishness and topic-sentiment as specified in 4.2.2, were computed to the document-level.

The topic-sentiment scores for Topics 3, 11, and 14 were computed using the FinBERT-FOMC and STM outputs on statements as outlined in 4.2.2. These topics from FOMC statements represent (3) swap lines and FX, (11) real economic events, as well as (14) percived inflation risks and outlook, respectively. The raw topic-sentiment scores for Topic 3 were dominated by few extreme values. This was due to the occasional paragraphs on swap lines during crises like Covid19 and the global financial crisis. To normalize such shocks, Topic 3's topic-sentiment scores were clipped such that $Score^{*}_{\mathrm{Doc},k} = \min(\max(Score_{\mathrm{Doc},k}, -0.015), 0.015)$. The topic-sentiment scores for Topics 11 and 14 were left as is. The document-level topic-sentiment scores are presented in Figure 22.

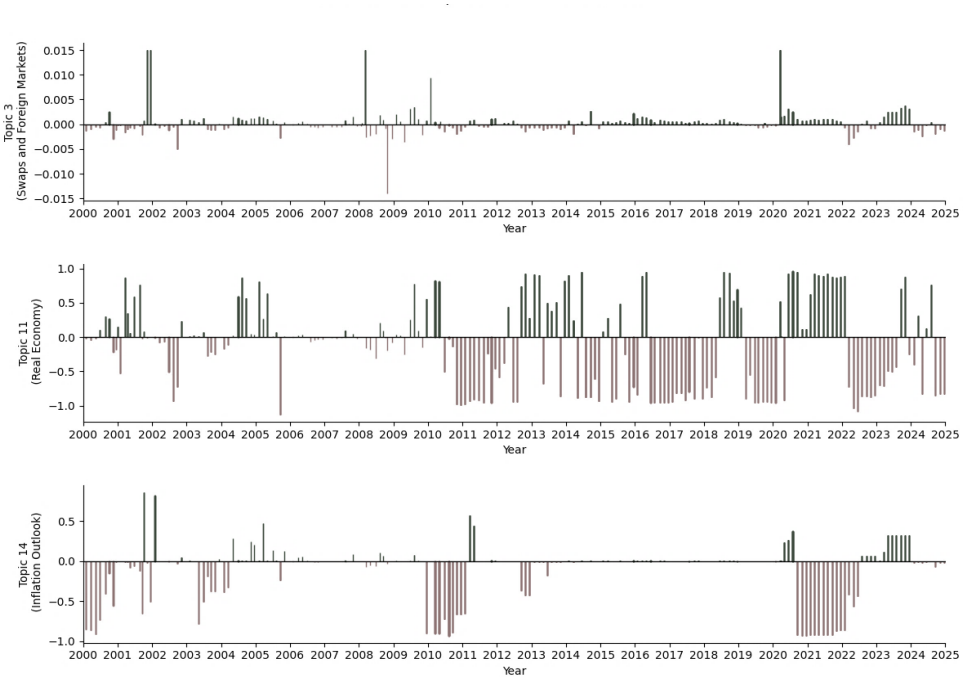

Figure 22: Statement Document-level Topic-Sentiment Scores

For minutes, the corresponding analysis yielded more stable topic-sentiment features. No clipping normalization was required. Interestingly, topic-sentiment measures appeared more correlated for minutes than statements. These document-level topic-sentiment scores are presented in Figure 23.

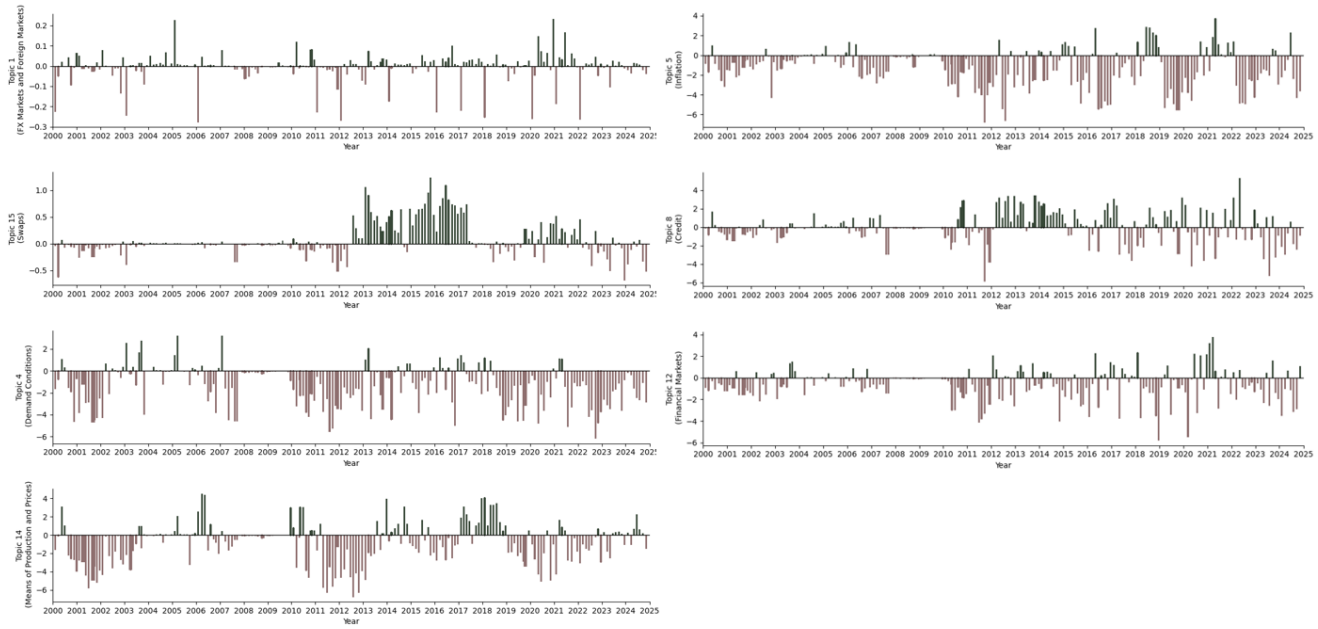

Figure 23: Minutes Document-level Topic-Sentiment Scores

As can be seen on topic-sentiment figures, the FinBERT-FOMC scoring method was more prone to assigning negative sentiment to the topics analyzed. Indeed, the mean sentiment score was -0.0015 and -0.0773 for statements and minutes respectively. This could be due to the FOMC focusing on economic health by addressing risks to its dual mandate that concern employment and inflation. Thus, their caution expressed in these topics might highlight the negative perceived scores.

The *hawkishness* scores for documents were measured based on the paragraph counts as described in 4.2.2. As with the topic-sentiment scores, the FOMC-RoBERTa seemed slightly cynical, by assigning a higher portion of the documents as more dovish than hawkish. The mean hawkishness was -0.1769 and -0.0539 for statements and minutes respectively. Hawkishness scores are shown in Figure 24.

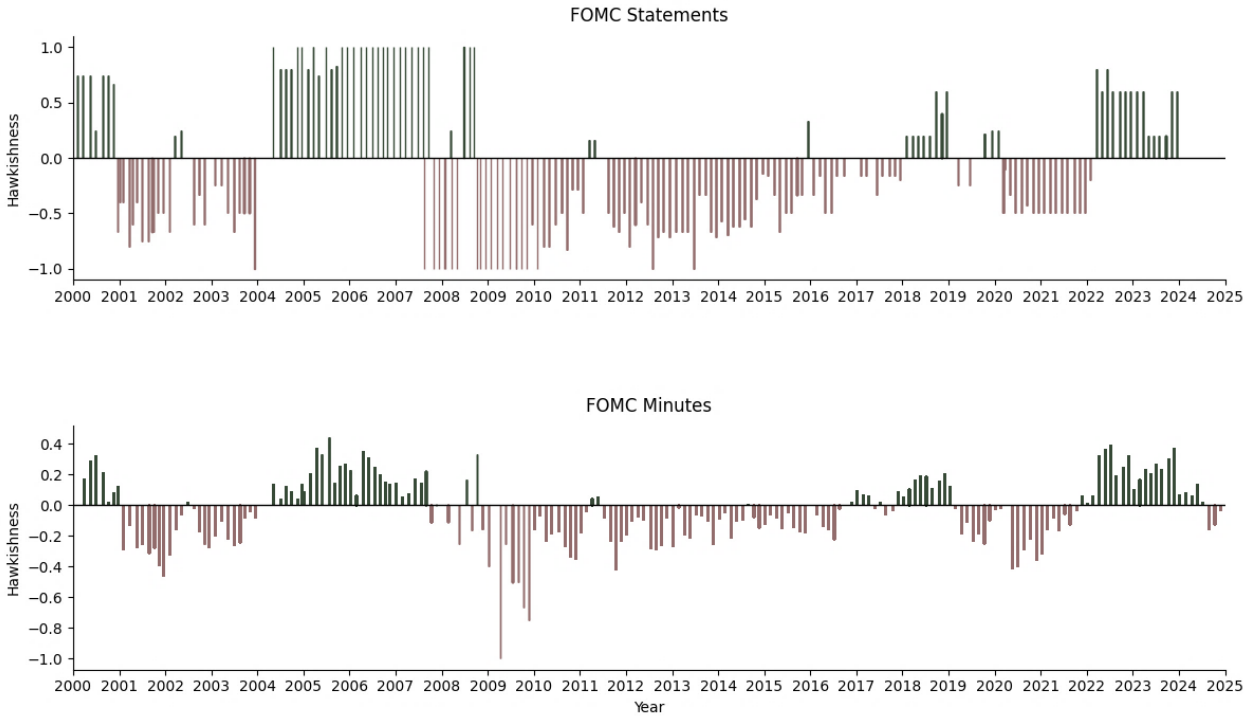

Figure 24: Hawkishness Scores

The *hawkishness scores* focused on the Feds monetary policy stance, such that a hawkish sentiment reflects the Feds orientation towards raising the federal funds rate and a dovish sentiment to lower the federal funds rate (Shah et al. 2023, 6664) as detected by FOMC-RoBERTa in the documents. In contrast, the topic-sentiment measures were used to proxy the FOMC's sentiment on the financial-economic (Gössi et al. 2023, 358) conditions associated with certain topics as proxied by the combined measure employing the STM and FinBERT-FOMC models.

Consequently, these fine-tuned BERT models inherently measure different tonal dimensions of the FOMC's text. This is illustrated by the BERT sentiment classifications for the following paragraph:

> *"Recent indicators suggest that economic activity has continued to expand at a solid pace. Job gains have moderated, and the unemployment rate has moved up but remains low. Inflation has eased over the past year but remains somewhat elevated. In recent months, there has been some further progress toward the Committee's 2 percent inflation objective."*
> (FOMC Statement, released 31st of July 2024.)

The FOMC-RoBERTa model assigns this paragraph with hawkish sentiment with certainty (0.9973), while the FinBERT-FOMC model assigns this paragraph positive financial-economic sentiment (0.9702). It is the authors interpretation that the models are fairly accurate in classifying this paragraph. Provided the description of inflation as *elevated* and the mention of only *some further*

progress toward the objective, sentiment along hawkish-dovish dimension is hawkish here. Also, low unemployment and solid economic growth suggest room for restrictive policy, but importantly, these signal positive financial-economic sentiment, which appears to be identified by FinBERT-FOMC.

In contrast, a paragraph which seems to simultaneously portray slightly dovish, yet to some degree positive but close to neutral financial-economic sentiment is also picked up by the models.[76]

> *"In the equity markets, the high perceived likelihood of a September cut in the target range for the policy rate induced a notable appreciation in the stocks of firms with small and medium capitalization, which tend to be more sensitive to interest rates. Stocks of larger companies, especially those in the technology sector, underperformed. Second-quarter earnings reports received before the meeting had been slightly above analysts' expectations, although some companies noted a softening in consumer spending."*
> (FOMC Minutes, released 21st of August 2024.)

The FOMC-RoBERTa model assigns this paragraph with slightly dovish sentiment (0.4944), while the FinBERT-FOMC model assigns this paragraph positive financial-economic sentiment with higher certainty (0.7614). Overall, aligning strongly with human judgement-based analysis, the models appeared to capture different sentiments provided by the FOMC releases quite well, but not perfectly. The following paragraph appeared wrongly classified as dovish (0.9499) and positive (0.7219),[77]

> *"In the second quarter, total nonfarm payroll employment posted its slowest average monthly increase since the recovery began in mid-2020, though payroll gains remained robust compared with those seen before the pandemic. Similarly, the private-sector job openings rate, as measured by the Job Openings and Labor Turnover Survey, fell in May to its lowest level since March 2021 but remained well above pre-pandemic levels. The unemployment rate edged down to 3.6 percent in June, while the labor force participation rate and the employment-to-population ratio were both unchanged. The unemployment rates for African Americans and Hispanics, however, both rose and were well above the national average. Average hourly earnings rose 4.4 percent over the 12 months ending in June, compared with a year-earlier increase of 5.4 percent."*
> (FOMC Minutes, released 16th of August 2023.)

In this paragraph, the first and last sentences imply some wage-push inflationary pressure, while increasing unemployment and slow job openings portray less optimistic financial-economic outlook.

---

[76] This is the Author's view provided the noted *softening in consumer spending* and stronger than expected earnings.
[77] This might be caused by the sliding window technique used for longer texts (discussed in 4.2.2).

5.2 Constructing the Hybrid Model Components

This section reports on the two components and benchmarks of the hybrid models: the structured data intraday regression ANNs, and the sentiment-driven logistic regression and RF classifiers. Subsection 5.2.1 delves into the performance and training of ANNs. Subsection 5.2.2 reports on the training and CV hyperparameter tuning process for sentiment models.

5.2.1 Structured Data Predictors: *Performance and Training of Intraday Regressors*

Overall, when measured by their regression performance, the MAPE-SB-MM2023 model fared better by MAPE than most models in the sampled previous literature. Yet, MSE-LB-MM2023, Macro and Tech exhibited similar RMSEs to the observed standard deviations for log-returns. All models performed similarly when measured by the classification metric DA. Indeed, the ANNs' classification performance was weak in light of literature discussed in 3.1.4. The aggregate means of key metrics observed in test data of the twelve test windows for ANNs are presented in Table 10.

Table 10: Aggregate Test Window Performance of ANNs

	MAPE-SB-MM2023	***MSE-LB-MM2023***	***Macro***	***Tech***
RMSE	0.0033583 (0.0052271)	0.0003196 (0.0000833)	0.0003320 (0.0000859)	0.0003190 (0.0000806)
MAPE	2.66% (1.22%)	129.27% (138.50%)	296.97% (183.48%)	424.90% (138.55%)
DA	49.92% (0.36%)	49.83% (0.29%)	50.06% (0.47%)	50.14% (0.46%)
Epochs	36.42 (14.89)	203.08 (34.70)	35.33 (25.41)	13.08 (5.73)

Note: *Numbers inside brackets are standard deviations. Data used encompasses the entire out-of-sample test window, not FOMC releases specifically.*

Importantly, these intraday models were implemented to FOMC release events. To test effectiveness and validity of implementing the ANN models to predict the EUR/USD during FOMC release events, univariate OLS-regressions for each of the models was conducted. The OLS-regressions were fit as,

$$r_{t_{type},5min} = \alpha_{type,model} + \beta_{model} * \hat{r}^{(Model)}_{t_{type},5min},$$

where, t_{type} is a document release event for document of $type \in \{Statments, Minutes\}$, subscript $model \in$ (MAPE-SB-MM2023,MSE-LB-MM2023, Macro, Tech) denotes the ANN evaluated. The data used to analyze the models' predictive power contained out-of-sample predictions and log-

returns observed during the FOMC releases from years 2022, 2023, and 2024. The adjusted R^2 is reported for each model with the p-values for t-test's statistical significance indicated by $***$, $**$, and $*$ for confidence levels 99, 95, and 90 percent, respectively. All regressions use Newey and West (1987) HAC covariance matrix estimators with lags chosen using automatic bandwidth selection by Andrews (1991).[78]

Table 11: Regression Analyses on Predictive Power of ANNs During FOMC Releases

	MAPE-SB-MM2023	***MSE-LB-MM2023***	***Macro***	***Tech***
Statements	-0.0147	0.01215	0.0882 (**)	-0.0323
Minutes	0.0148	0.2061 (**)	0.1145 (*)	0.0170

Results indicate that intraday ANNs could explain some variance observed in the 5-minute log-returns of the test data. Only MSE-LB-MM2023 achieved a α close to 0 with $\beta = 1.008$ under minutes releases. However, most relationships were not consistently significant. This shows that some learning might transfer over to high-volatility FOMC release events (Rosa 2013, 70; IMF 2023, 779).

The rest of this subsection focuses on the training and performance of individual ANN models as general intraday EUR/USD predictors. As summarized in Table 12, the two variations of the MM2023 regression model performed expectedly within test windows. MAPE-SB-MM2023 performed well by MAPE and MSE-LB-MM2023 by RMSE. Notably, DA for the predicted log-returns with both models appear to change in time, with MAPE-LB-MM2023 slightly outperforming its competitor.

Table 12: Performance of the MM2023 ANN Models in Test Windows

Test Window		**Q1 2022**	**Q2 2022**	**Q3 2022**	**Q4 2022**	**Q1 2023**	**Q2 2023**
MAPE-SB-MM2023							
Training Epochs		22	47	50	50	50	25
	MAPE	1.26%	1.34%	1.66%	1.93%	5.04%	2.06%
	RMSE	0.0194789	0.0035727	0.004929	0.0023943	0.0014721	0.0019202
	DA	0.5056	0.5027	0.5007	0.5022	0.4982	0.4974
MSE-LB-MM2023							
Training Epochs		232	109	240	174	187	192
	MAPE	41.85%	386.77%	37.97%	37.27%	50.64%	240.37%
	RMSE	0.0003503	0.0003727	0.0004602	0.0004698	0.0003566	0.0002712
	DA	0.4943	0.4978	0.4927	0.5016	0.5003	0.5009

[78] These regressions were implemented in R with the sandwich package (Zeileis 2004).

Table 12 continued

Test Window		Q3 2023	Q4 2023	Q1 2024	Q2 2024	Q3 2024	Q4 2024
MAPE-SB-MM2023							
Training Epochs		45	11	23	50	13	38
	MAPE	3.78%	2.84%	4.39%	3.17%	2.04%	2.38%
	RMSE	0.0006156	0.0013467	0.0009922	0.0007484	0.0015942	0.0012347
	DA	0.4950	0.5004	0.4960	0.4943	0.5019	0.4955
MSE-LB-MM2023							
Training Epochs		225	161	234	190	136	235
	MAPE	61.51%	141.41%	69.59%	23.02%	54.77%	406.04%
	RMSE	0.0002753	0.0002801	0.0002336	0.0002325	0.0002317	0.0003012
	DA	0.4952	0.5000	0.4989	0.4976	0.4983	0.5015

Note: *MAPE, RMSE, and DA show performance measured by: Mean Average Percentage Error, Root Mean Square Error and Directional Accuracy, respectively.*

Furthermore, while MAPE-SB-MM2023 outperformed MSE-LB-MM2023 on most metrics, their error plots revealed that the short-budget models' errors exhibit more heteroskedastic and autocorrelated behavior. This behavior is illustrated in Figure 25.

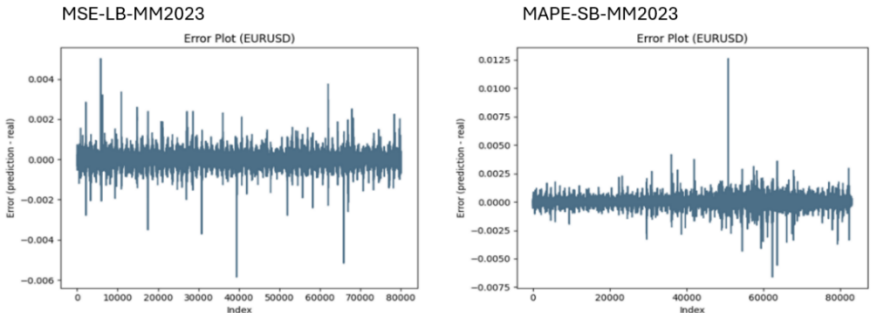

Figure 25: Regression Model Heteroskedasticity and Training Budget

During training, the metrics of MM2023 models appeared to follow a similar pattern for all test windows. As epochs increased MSE loss decayed, MAPE followed a humped path for the first 20-60 epochs, and DA appeared to stabilize around 0.5 with very high variance for the first 200 epochs. Due to the humped path MAPE-SB-MM2023 stopped training right before or after the hump appeared. A MM2023 training session for the first MSE-LB-MM2023 is illustrated in Figure 26 below.

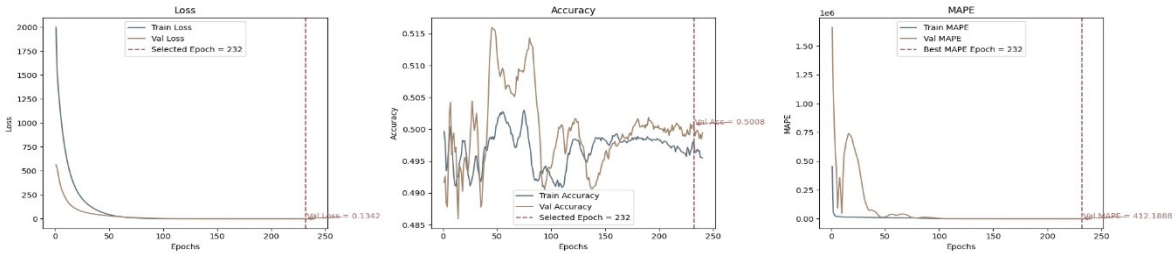

Figure 26: Regression Performance Metrics in Training (MSE-LB-MM2023 in 2021Q1, Scaled Label)

Interestingly, in training the diagnostic metrics of the Macro and Tech models behaved quite similarly to each other. The out-of-sample performance of these two models is reported in Table 13.

Table 13: Performance of the Macro and Tech ANN Models in Test Windows

Test Window		**Q1 2022**	**Q2 2022**	**Q3 2022**	**Q4 2022**	**Q1 2023**	**Q2 2023**
Macro							
Training Epochs		18	14	16	66	94	13
	MAPE	712.21%	522.43%	112.45%	386.3%	310.59%	310.96%
	RMSE	0.0003523	0.0003953	0.0004842	0.0004945	0.0003652	0.0002530
	DA	0.5029	0.5057	0.5069	0.5003	0.5026	0.5021
Tech							
Training Epochs		10	6	23	19	11	5
	MAPE	712.21%	460.29%	558.78%	397.18%	282.55%	656.91%
	RMSE	0.0003524	0.0003727	0.0004597	0.0004668	0.0003563	0.0002710
	DA	0.5029	0.5058	0.4915	0.5021	0.5010	0.5086

Test Window		**Q3 2023**	**Q4 2023**	**Q1 2024**	**Q2 2024**	**Q3 2024**	**Q4 2024**
Macro							
Training Epochs		31	29	63	25	32	23
	MAPE	422.02%	227%	63.79%	120.98%	190.66%	183.93%
	RMSE	0.0002879	0.0002995	0.0002523	0.0002473	0.0002397	0.0003127
	DA	0.4903	0.4974	0.5031	0.5020	0.4949	0.5007
Tech							
Training Epochs		14	6	13	16	15	19
	MAPE	352.32%	343.72%	306.04%	384.78%	311.88%	332.16%
	RMSE	0.0002753	0.0002801	0.0002337	0.0002325	0.0002265	0.0003013
	DA	0.4990	0.5049	0.4991	0.5002	0.4972	0.5068

Note: *MAPE, RMSE, and DA show performance measured by: Mean Average Percentage Error, Root Mean Square Error and Directional Accuracy, respectively.*

Both models performed better by regression metrics in the last six test windows and classifications metrics in the first six test windows. By MAPE and RMSE both models appeared to perform worse than models reported in previous literature, yet by classification accuracy the Tech model appeared most promising out of all four models. The computation time required to train these models was far less than that of the MM2023 models.

During training both the Macro and Tech models' metrics appeared to follow similar paths. MAPE followed a U-shape, DA was volatile but improved at first, the R2-scores grew until the validation and training data metrics began to diverge, and the loss values for training and validation data fell in tandem. Therefore, early stopping for these models was selected to attempt to stop at the point where R2-scores diverged as it usually coincided with increases in MAPE. This is illustrated in Figure 27.

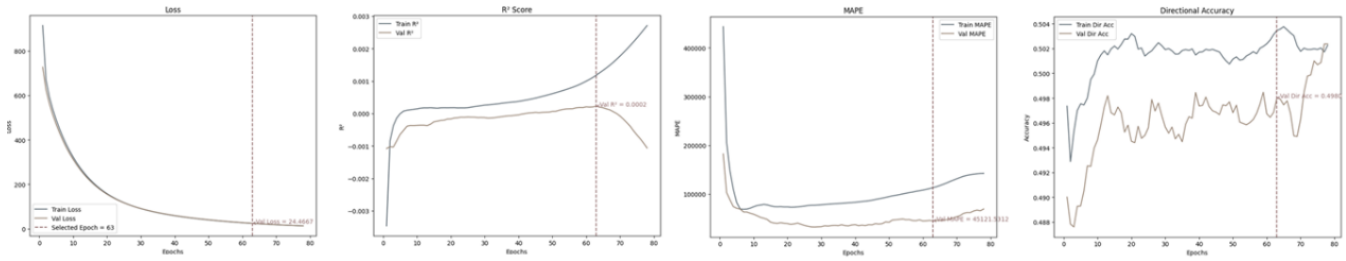

Figure 27: Regression Performance Metrics in Training (Macro in 2024Q4, Scaled Label)

5.2.2 Sentiment Feature Predictors: *Parametrization of Release Day Classifiers*

To gauge the relationship between the computed sentiment scores (see 5.1) and 5-minute log-returns observed during FOMC statements and minutes releases, two multivariate OLS-regressions were run by document type. Formally these validating regressions were fit as:

$$r_{t_{type},5min} = \alpha_{type} + \beta_{type,hawk} * hawkishness_{t_{type}} + \sum_{k \in type} \beta_{type,k} * Score_{t_{type},k} ,$$

where, t_{type} is the document release event index for document of $type \in \{Statments, Minutes\}$. Also, k denotes topics identified and scored for economic-financial sentiment. Following Shah et al (2023, 6671), regressions were carried out as standard OLS estimations based on document release timestamps. The t-test significances were computed for coefficients based on HAC covariance matrix estimators with the lag setting following automatic bandwidth selection outlined in Andrews (1991). Results of these regressions are reported Table 14. The signs of β-coefficients are shown for each feature, with t-test statistical significance indicated by ***, **, and * for confidence levels 99, 95, and 90 percent, respectively. Additionally, adjusted $\bar{R}^2$ scores are indicated below. The regression spanned the data consisting of 298 FOMC release events between 1.1.2004 and 31.12.2022.

Table 14: Regression Coefficient Analyses on Sentiment Features

	Hawkishness	***Topic 3***	***Topic 11***	***Topic 14***	***Topic 1***	***Topic 4***	***Topic 5***	***Topic 8***	***Topic 12***	***Topic 15***
Statements	+	+ (***)	+	−						
Minutes	+ (*)			+	−	+ (*)	−	+	+	+

Statements $\bar{R}^2 = -0.005483$ Minutes $\bar{R}^2 = 0.000965$

Statistically significant relationships between the observed cross-section of 5-minute log-returns and topic-sentiment on topics, related to foreign markets for statements and economic growth for minutes release events. Also, the hawkishness score computed for this thesis showed only a significant

relationship for future 5-minute log-returns observed during minutes releases. Notably, the coefficient significances and adjusted $\bar{R}^2$ do not signify strong potential for using the selected scoring techniques to predict future 5-minute log-returns during the two types of FOMC document release events. Also, in light of results of such those of Osowska and Wójcik (2024, 164-165), and Aattouchi and Kerroum 2022, 125-129), the reported results in Table 14 are not sufficient to posit that the selected approach to sentiment scoring was optimal for capturing sentiment effects on FX markets.

With no hyperparameter tuning the sentiment models appeared to overfit the training data. Especially, RF models trained with the default configurations provided by the Scikit-Learn Python package, usually fit the training data perfectly with AUC and accuracy close to 100%. However, results of such models were relatively weak in test data. Thus, AUC-based CV hyperparameter tuning process as outlined in 4.3.2 was carried out. Five-fold CV was used to select hyperparameters of the final sentiment models. The results by AUC for tuned and untuned models are reported in Table 15.

Table 15: Sentiment Model Performance by AUC

	2022		***2023***		***2024***	
	Tuned & Weighted	Default	Tuned & Weighted	Default	Tuned & Weighted	Default
Statements						
LR						
Test Data	0.2667	0.4000	0.1875	0.1875	0.1875	0.1875
Train Data	0.5240	0.5256	0.5339	0.5526	0.5254	0.5247
RF						
Test Data	0.7667	0.2667	0.8750	0.1250	0.4688	0.3125
Train Data	0.6207	1.0000	0.6777	1.0000	0.7071	1.0000
Minutes						
LR						
Test Data	0.4286	0.1429	0.8667	0.7333	0.6000	0.4000
Train Data	0.5846	0.5968	0.5830	0.5920	0.5909	0.5941
RF						
Test Data	0.7143	0.2857	0.6000	0.4667	0.5333	0.3333
Train Data	0.6735	0.9998	0.5920	0.6624	0.5941	0.8482

Note: *The figures reported are ROC AUC scores. All test year data include eight instances of statements and minutes releases.*

CV hyperparameter tuning combined with the weighted loss functions appeared to yield superior results. Though the datasets used in training were relatively small, with less than 200 observations (see Table 8), these procedures seemingly minimized overfitting as reflected in Table 15. Overall, without tuning models performed worse than random predictions with AUC scores of less than 0.5 on the statements test data. After tuning, especially RF models appeared to be reliable predictors.

During the CV procedure, AUC and accuracy were visually gauged to be somewhat correlated, with respect to changes in hyperparameter values. Thus, when selecting the best hyperparameters by AUC, the accuracy of the model in the test data often improved in tandem. This was apparent through mean AUC and accuracy values computed with CV validation. Examples of this behavior are illustrated in Figure 28, which was plotted on the training data for test sets of 2022 and 2023.

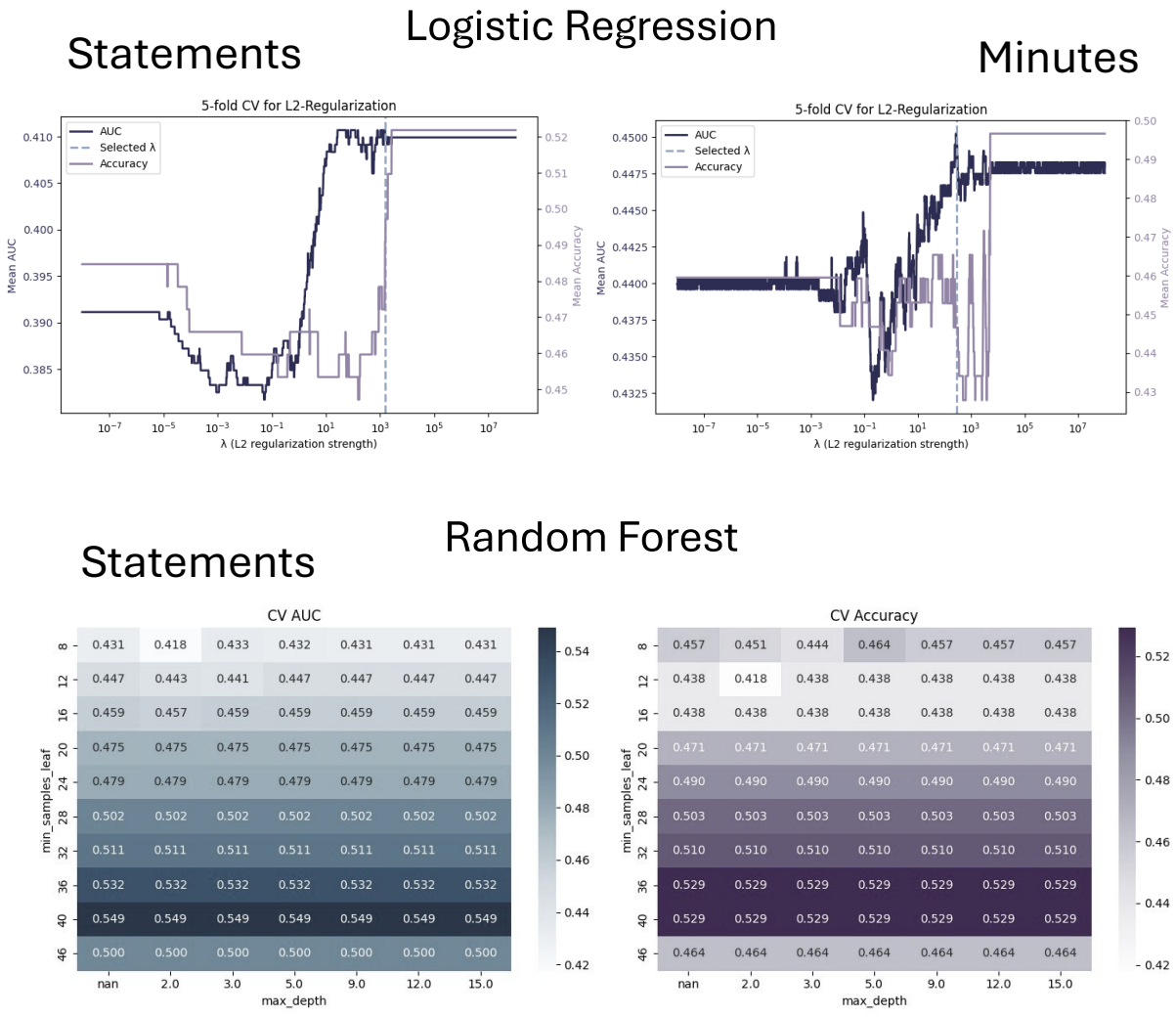

Figure 28: Sentiment Model Hyperparameter Tuning

5.3 Evaluating Performance During FOMC Statement and Minutes Releases

This section reports on the classification performance of intraday ANNs, sentiment RF and LR, and two hybrid models EW and PERF for 5-minute predictions on direction of the EUR/USD. As expected, prediction accuracy was higher during FOMC minutes releases than statements releases, with hybrid producing most of the total correct predictions, closely followed by the sentiment models. Out-of-sample accuracy of the eight models are summarized in Table 16.

Table 16: Accuracy of EUR/USD Direction Predictions

Statements	***2022***	***2023***	***2024***	***2022-2024***
ANNs				45.83%
MAPE-SB-MM2023	50.00%	37.50%	50.00%	45.83%
MSE-LB-MM2023	37.50%	37.50%	25.00%	33.33%
Macro	50.00%	37.50%	37.50%	41.67%
Tech	62.50%	50.00%	75.00%	**62.50%**
Sentiment Models				47.92%
LR	37.50%	50.00%	37.50%	41.67%
RF	50.00%	75.00%	37.50%	54.17%
Hybrid Models				52.08%
EW	37.50%	62.50%	50.00%	50.00%
PERF	37.50%	62.50%	62.50%	54.17%
Minutes	**2022**	**2023**	**2024**	**2022-2024**
ANNs				51.04%
MAPE-SB-MM2023	62.50%	62.50%	37.50%	54.17%
MSE-LB-MM2023	87.50%	50.00%	37.50%	**58.33%**
Macro	12.50%	37.50%	62.50%	37.50%
Tech	50.00%	50.00%	62.50%	54.17%
Sentiment Models				56.25%
LR	62.50%	37.50%	62.50%	54.17%
RF	75.00%	50.00%	50.00%	**58.33%**
Hybrid Models				54.17%
EW	62.50%	50.00%	62.50%	**58.33%**
PERF	62.50%	37.50%	50.00%	50.00%

Note: *Accuracy was calculated out-of-sample for eight release events per test year. The last column displays accuracy over all test years. ANNs were re-trained each quarter and sentiment models each year. Highest accuracies per release type are bolded.*

Based on accuracy by model category, hybrid models performed the best with a total of 51/96 correct classifications, sentiment models were essentially on par with 50/96 correct classifications. The intraday ANNs classified 93/192 instances correctly. However, as a single model the technical indicator ANN, or Tech, performed best, achieving the highest accuracy score out of all models with consistently high performance during both FOMC release events. It is notable that this was the only model to include a plethora of variance-like features, which may have contributed to its performance in these highly volatility market environments (Kansoy 2022). Yet, the ANN category exhibited the highest volatility, with relatively average category-level performance during minutes releases.

The sentiment models performed surprisingly well given the weak OLS regression results. Also, RFs quite consistently outperformed LR models. Similar to Osowska and Wójcik (2024, 166), the RFs exhibited the highest accuracy. This arguably positions RF as a viable method for predicting the EUR/USD using sentiment scores during FOMC release events for both document types.[79] In general, during minutes releases, sentiment models outperformed ANNs and were mostly more consistent.

While hybrid models rarely outperformed their best single component model in any given year, they appeared to smooth out misclassification risks, by showing stable performance around or mostly above 50% accuracy. Over the three-year time-horizon, individual hybrid models exhibited higher accuracy than the categorical averages of their component models, three out of four times. This illustrates viability of such hybrid ensembles as a mechanism to smooth-out variance. It is worth noting that, specifically one hybrid model, *PERF,* underperformed its components models' categorical three-year average accuracy during FOMC minutes release events. This was likely caused by volatile ANN performance in 2022 data influencing the dynamic weighting scheme.

When comparing quality of model predictions over the three-year test data precision metrics convey diversity in the reliability of models' predictions. Indeed, LR predictions were mostly unreliably.[80] EW had relatively balanced and comparably high precision scores across both datasets. Meaning that their predictions were correct relatively reliably when benchmarked against other models when predicting both depreciation and appreciation. PERF exceeded the precision of its best component model in the statements data, however no such performance is observed in minutes data. In contrast to EW, PERF showed imbalanced precision scores. Precision scores are reported in Table 17.

---

[79] A test of different proportions (Equation 35) was computed for a z-statistic 0.8230 and a p-value of 0.4105.
[80] The LR model predicted an EUR/USD decrease or a USD appreciation only three times, twice in the minutes data and once in the statements data. All three depreciations were predicted wrongly, as evidence by precision (see Table 17).

Table 17: Precision of EUR/USD Direction Predictions

	MAPE-SB-MM2023	MSE-LB-MM2023	MACRO	TECH	LR	RF	EW	PERF
Statements								
USD Depreciation	41.67%	35.29%	41.18%	57.14%	43.48%	50.00%	47.06%	50.00%
USD Appreciation	50.00%	28.57%	42.86%	70.00%	0.00%	60.00%	57.14%	75.00%
Frequency Weighted Average	46.18%	31.65%	42.09%	64.11%	19.93%	55.42%	52.52%	63.54%
Minutes								
USD Depreciation	66.67%	64.71%	50.00%	64.29%	59.09%	64.71%	64.71%	58.82%
USD Appreciation	41.67%	42.86%	28.57%	40.00%	0.00%	42.86%	42.86%	28.57%
Frequency Weighted Average	57.29%	56.51%	41.96%	55.18%	36.93%	56.51%	56.51%	47.48%

Note: *In this context, precision scores convey proportions of correct predictions made for the future 5-minute directions of the EUR/USD.*

To scientifically analyze relative model performance, GW and McNemar statistical tests were conducted as outlined (4.3.2). Notably, in this study, these two methods produced highly correlated p-values, with mid-*p* presenting a more conservative interpretation. In general, the statements release market environment lead to more statistically significant differences in model performance than the minutes release. The GW tests suggested evidence for five statistically significant relationships out of the total 56 inter-model loss-differentials, of which three are in favor of the hybrid models. Significant loss-differentials are summarized in Table 18 by number of misclassifications, with GW test significance indicated by $***$, $**$, and $*$ for confidence levels 99, 95, and 90 percent, respectively. The GW t-tests were evaluated with HAC using a lag of 1, allowing for autocorrelated outcomes for consecutive FOMC release events. It is important to note the small sample size. Thus, while each data point represents an economically meaningful event, the inference should be interpreted cautiously.

Table 18: Summary of Giacomini-White 2006 T-tests

	Tech	***RF***	***EW***	***PERF***
MSE-LB-MM2023	**+7**	**+5**	**+4**	**+5**
Statements data	$(**)$	$(**)$	$(**)$	$(**)$
Macro			**+5**	
Minutes data			$(*)$	

Note: *Each cell represents the number of additional misclassifications by the row model relative to the column model, formally,* $n \times \Delta\bar{L}_n$*, where* $n = 24$.

Furthermore, the GW and mid-*p* tests revealed that outperformance of Tech and RF against the MSE-LB-MM2023[81] model might have carried over to the hybrid models, in the statements data. It is also noteworthy that half of the significant GW and mid-*p* tests observed in statements data were in favor of hybrid models. Also, the only significant minutes loss-differential was identified between the EW hybrid model and the worst performing ANN, as evidenced by the GW test. The mid-*p* method found no significant differences between model misclassification under FOMC minutes releases. Indeed, in the minutes data, the only mid-*p* test close to significant was observed between the EW hybrid model and Macro ANN, supporting results of the GW test. This suggests that the idiosyncratic prediction errors of at least the weakest model were mitigated by the EW hybrid model. Results of these tests by their p-values are summarized in Figure 29 below.

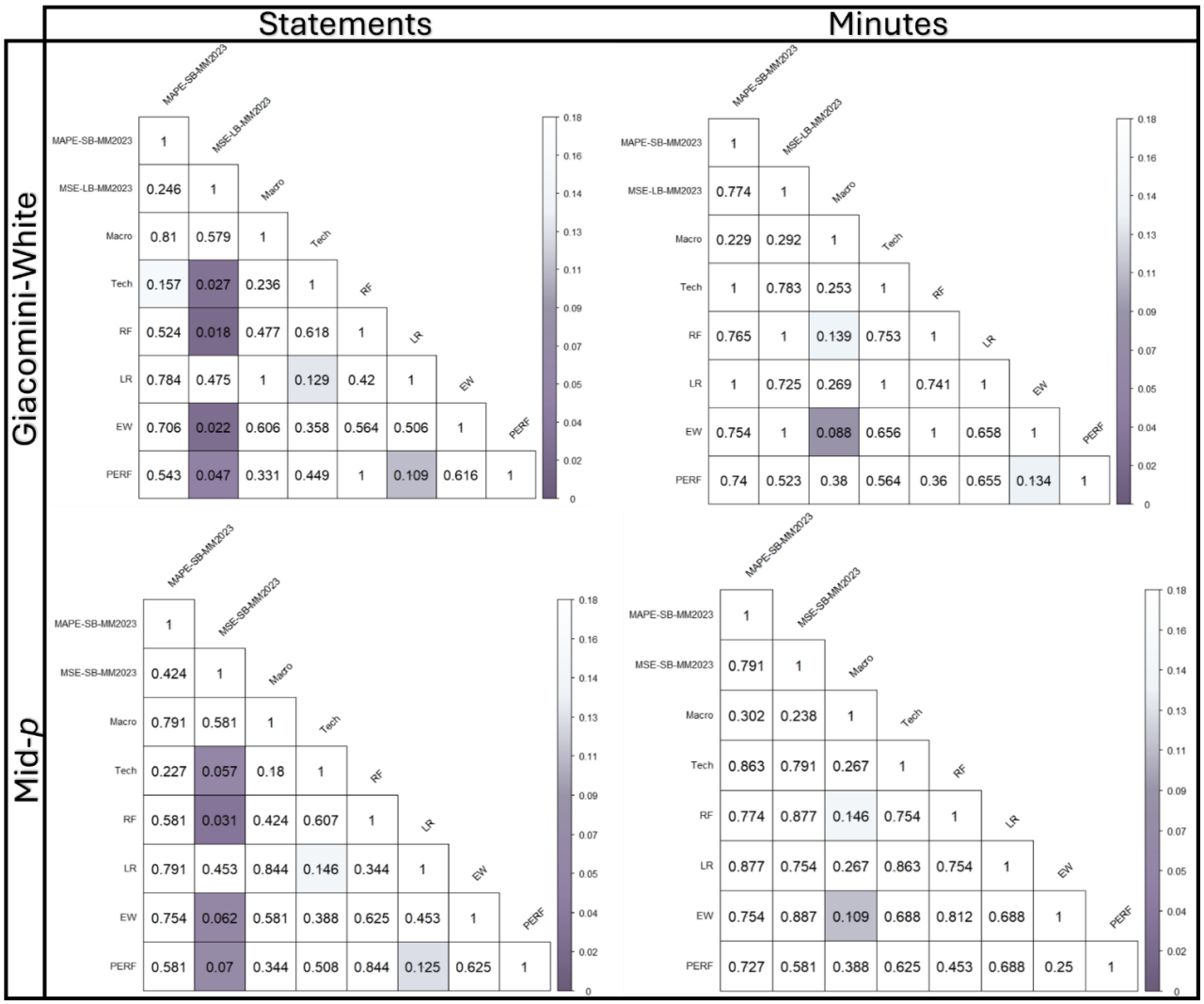

Figure 29: Giacomini-White 2006 and Mid-*p* test p-values

[81] Notably, this model had the highest adjusted $\bar{R}^2$ during minutes releases.

Moreover, conservative exact McNemar p-values were also computed. Exact p-values yielded statistically significant out-of-sample differences in misclassification between the MSE-LB-MM2023 model vs. the RF and Tech models in the statements test data set, with p-values of 6.25% and 9.22%, respectively. Simultaneously, hybrid model p-values were both 12.50%. Given literature on McNemar tests, with recommendations from Fagerland et al. (2013) and Sundjaja et al. (2025) in mind, exact p-values were not used. Yet, these results signal a need for caution when evaluating relative performance of models.

Overall, GW and McNemar mid-*p* tests results suggest that comparative performance of top performing models against weakest model is similar or occasionally improved in significance by implementing hybrid models. EW's p-values against the weakest model were either significant or marginally close to significant. Results provide tentative evidence that hybrid models could mitigate misclassification risks associated with the weakest component models and provide more consistency.

In general, the observed consistency gains from hybrid models align with results of Ling et al. (2021). Ling et al. (2021, 9, 18) find their hybrid model improved performance consistency and predictive power. Though as in Ling et al (2021, 9-10), some component models had higher directional accuracy than the hybrid. Additionally, Guyard and Deriaz (2024, 6-8) find that stacked meta estimators predicting the daily direction of the EUR/USD show improved accuracy over a multi-year period. While accuracy gains in this thesis were modest, vote-tallying hybrid models produced above average accuracies by category when all three years of test data were considered. Also, while statistical tests did not evidence hybrid models' outperformance over all models compared, they do suggest that idiosyncratic misclassification risks from component models during FOMC releases could be partially mitigated by combining sentiment-based models with intraday ANNs.

Therefore, In light of this thesis research question,

> To what extent do hybrid models that combine intraday ANN predictors and FOMC sentiment predictors enhance EUR/USD direction forecasting performance during FOMC statements and minutes releases compared to singular models?

Empirical findings of this thesis suggest that vote-tallying hybrid models could offer more consistent performance, rather than definitive accuracy gains over their component models, particularly in the volatile markets during FOMC statement and minutes releases. Also, balanced outcomes were achieved by employing an equal weighted hybrid model. Notably, most NLP architecture implemented was available, partly or completely, by 2024. This fact, together with strong prediction rates reported in Table 16, suggest possible pricing anomalies which can challenge local consistency of EMH in the most liquid market. Further exploration of these anomalies is left to future research.

6 Conclusions

This thesis investigated the extent of performance enhancing effects that hybrid models, which combine intraday ANN and FOMC sentiment models, provide over singular models in forecasting 5-minute direction of the EUR/USD during FOMC statements and minutes releases. Based on out-of-sample performance, between six component models and two vote-tallying hybrid models, results suggest that rather than definitive accuracy gains over component models, hybrid models offer consistency during volatile FOMC statement and minutes release market environments. Such improvement appeared particularly true for an equally weighted hybrid model. Results were assessed by accuracy, precision metrics, and Giacomini-White (2006) and mid-*p* statistical tests.

To summarize, akin to diversification in portfolio selection, the old adage of *don't put all your eggs in one basket* carries a useful principle also for model selection when producing directional signals for the EUR/USD during FOMC statements and minutes release events. While further research is required, results of this thesis suggest that using equally weighted hybrid models can be beneficial.

Building on nascent research concerning intraday prediction of the EUR/USD with sentiment scores from FOMC releases, this thesis incorporates FOMC-specific BERT models, (Gössi et al. 2023; Shah et al. 2023) as suggested by Osowska and Wójcik (2024, 167).[82] The sentiment scoring implementation was inspired by previous research (Jegadeesh & Wu 2017; Aattouchi & Kerroum 2022; Shah et al. 2023). Results indicated promising performance and could question local consistency of EMH in FX market environments following FOMC statements and minutes releases.

However, as stated by Osowska and Wójcik (2024, 166-167) the time horizon for predicting returns with BERT-based sentiment-scores appears to be brief, with reliable sentiment-score impacts marginal. Similar findings for sentiment-scores used in this thesis were found. To alleviate issues with unreliable predictions, this thesis turned towards ensemble models from ML research (Breiman 2001) and previous research on vote-tallying hybrid models for EUR/USD prediction (Ling et al. 2021; Guyard & Deriaz 2024). Thus, this study focused on the extent to which hybrid models that combine intraday models together with FOMC specific sentiment models could enhance direction forecast performance during statements and minutes releases compared to singular models.

This was achieved by training both general intraday ANN predictors and sentiment models as component models to help implement and benchmark two hybrid models. First, intraday ANNs were

[82] This addresses the first aspect to consider for future research as outlined by Osowska and Wójcik (2024).

trained based on previous research. Next, sentiment models were developed and trained with a novel approach to FOMC sentiment predictors, motivated by Gössi et al. (2023) and Shah et al. (2023) for FOMC-specific BERT models, and Jegadeesh and Wu (2017) for topic-level sentiment scoring. Finally, by implementing six component models and two vote-tallying hybrid models to predict the 5-minute direction of the EUR/USD, results of this thesis were obtained slightly in favor of hybrid models. Additionally, accuracy and precision of sentiment RFs and hybrid models in the statements and minutes release data suggests that temporary market inefficiency with potential for local arbitrage opportunities might have existed for actors using such sentiment based FX prediction models.

Three categories of limitations to this study were (1) a small and noisy dataset, (2) non-assessment of inter-model correlation, and (3) risk of look-ahead-bias. The first limitation impacts generalizability. Given the small test datasets of 24 observations with which experiments were conducted, results should grant only suggestive evidence. This limitation could be addressed in future research by simply expanding the test data, conducting multiple tests with CV, or running bootstrapped resampling. Second, effects of correlation between model predictions were not accounted for. Results of a hybrid model could have improved by lowering covariance of predictions to smooth-out idiosyncratic errors.

The third limitation poses some risk to the validity of this study. Some of the ANNs trained in this study were informed by previous research in which models were evaluated in the same test data. Specifically, Markova (2023) tested her model in 2022 EUR/USD data and based on her results, a similar model was selected for this study to model 2022 data, for which the selected ANN architecture had been shown to be suitable. Thus, decisions on ANN predictors' hyperparameters for 2022 test data were, to an extent, informed by out-of-sample information. Moreover, FOMC-RoBERTa (Shah et al 2023) and FinBERT-FOMC (Kim et al. 2023) implemented for sentiment analysis was fine-tuned with data including sentences from 2022 FOMC communications. Therefore, some sentiment scores in 2022 data might have been computed with *memorized* information from model parameters.

Future research in predicting the EUR/USD during FOMC releases may explore more ways to compute sentiment scores with FOMC-specific BERT models or perhaps utilize pronouncements of the ECB. In addition, STMs could be implemented in an arguably opaque manner with direction of the EUR/USD used as a covariate. Moreover, as studied in Kim et al. (2024), FOMC-specific prompt engineering with LLMs, or even Retrieval Augmented Generation, can be considered. However, such approaches should bear in mind inference time against the backdrop of the fast-paced FX market.

Finally, potential future shifts in Fed governance subsequent to the 47th executive administration may, among other things, alter language of the FOMC, affecting applicability of the BERT models used.

References

Abhishek, K. —Kawahara, J. —Hamarneh, G. (2021) Predicting the clinical management of skin lesions using deep learning. *Scientific Reports*, Vol. 11(1), 7769. <https://doi.org/10.1038/s41598-021-87064-7>

Aggarwal, C. (2023) *Neural Networks and Deep Learning: A Textbook*. 2. Springer International Publishing.

Andersen, T. —Bollerslev, T. —Diebold, F. —Vega, C. (2003) Micro Effects of Macro Announcements: Real-Time Price Discovery in Foreign Exchange. *American Economic Review*, Vol. 93(1), 38–62. <https://doi.org/10.1257/000282803321455151>

Andrews, D. (1991. Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation. *Econometrica*, Vol. 59(3), 817–858. <https://doi.org/10.2307/2938229>

Araci, D. (2019) FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. arXiv:1908.10063. *arXiv*. <https://doi.org/10.48550/arXiv.1908.10063>

Arlot, S.—Celisse, A. (2010) A survey of cross-validation procedures for model selection. *Statistics Surveys*, Vol. 4(1), 40–79. <https://doi.org/10.1214/09-SS054>

Aattouchi, I.—Kerroum, M. (2022) A New Framework for Analyzing News in the Financial Markets to Enhance the Investor's Perception. *Journal of Advances in Information Technology*, Vol. 13(2). <https://doi.org/10.12720/jait.13.2.125-131>

Ayitey Junior, M. —Appiahene, P. —Appiah, O.—Bombie, C. (2023) Forex market forecasting using machine learning: Systematic Literature Review and meta-analysis. *Journal of Big Data*, Vol. 10(1), 9. <https://doi.org/10.1186/s40537-022-00676-2>

Bektash, O.—la Cour-Harbo, A. (2023) First-order Layer in Artificial Pain Pathway. *Neural Processing Letters*, Vol. 55(1), 319–343. <https://doi.org/10.1007/s11063-022-10884-9>

Belkin, M. —Hsu, D. —Ma, S., —Mandal, S. (2019) Reconciling modern machine learning practice and the bias-variance trade-off. *Proceedings of the National Academy of Sciences*, Vol. *116*(32), 15849–15854. <https://doi.org/10.1073/pnas.1903070116>

Ben Omrane, W. —Savaser, T. —Welch, R. —Zhou, X. (2019) Time-varying effects of macroeconomic news on euro-dollar returns. *The North American Journal of Economics and Finance*, Vol. 50, 101001. <https://doi.org/10.1016/j.najef.2019.101001>

Ben Omrane, W. —Welch, R. —Zhou, X. (2020) The dynamic effect of macroeconomic news on the euro/US dollar exchange rate. *Journal of Forecasting*, Vol. 39(1), 84–103. <https://doi.org/10.1002/for.2615>

Bengio, Y. —Simard, P.—Frasconi, P. (1994) Learning long-term dependencies with gradient is difficult. *Neural Networks*, Vol. 5 (2), 157–166.

BIS (2022), OTC foreign exchange turnover in April 2022. <https://www.bis.org/statistics/rpfx22_fx.htm>, Retrieved February 10, 2025.

BIS (2025), OTC foreign exchange turnover in April 2025. <https://www.bis.org/statistics/rpfx25_fx.htm>, Retrieved September 14, 2025.

Bischof, J. —Airoldi, E. (2012) Summarizing topical content with word frequency and exclusivity In Proceedings of the International Conference on Machine Learning, 201-208. <https://icml.cc/2012/papers/113.pdf>

Bishop, C. (2006) *Pattern recognition and machine learning*. Springer, Singapore

Bjønnes, G. —Osler, C. —Rime, D. (2021) Price discovery in two-tier markets. *International Journal of Finance & Economics*, Vol. 26(2), 3109–3133.

Blei, D. — Ng, A. — Jordan, M. (2003) Latent Dirichlet Allocation. *Journal of Machine Learning Research*, Vol. 3(1), 993–1022. <https://jmlr.csail.mit.edu/papers/v3/blei03a.html>

Botsch, M. (2023) *Machine Learning Techniques for Time Series Classification*. 2. Cuvillier Verlag, Göttingen, Germany.

Brusa, F.—Savor, P.—Wilson, M. (2020) One Central Bank to Rule Them All. *Review of Finance*, Vol. 24(2), 263–304. <https://doi.org/10.1093/rof/rfz015>

Butler, M. —Kazakov, D. (2010) Particle Swarm Optimization of Bollinger Bands. Swarm Intelligence, 504–511. <https://doi.org/10.1007/978-3-642-15461-4_50>

Cerchiello, P. —Nicola, G. (2018) Assessing News Contagion in Finance. *Econometrics*, Vol. 6(1), 5–23. <https://doi.org/10.3390/econometrics6010005>

Chaboud, A.—Chiquoine, B. — Hjalmarsson, E. —Vega, C. (2014). Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market. *The Journal of Finance*, Vol. 69(5), 2045–2084. <https://doi.org/10.1111/jofi.12186>

Chaboud, A.—Dao, A. —Vega, C. —Zikes, F. (2024) What Makes HFTs Tick? Tick Size Changes and Information Advantage in a Market with Fast and Slow Traders. *Management Science*, Vol. 71(1), 553–583. <https://doi.org/10.1287/mnsc.2022.02935>

Chaboud, A.—Rime, D.—Sushko, V. (2023) Chapter 12: The foreign exchange market. In *Research Handbook of Financial Markets,* eds. Gürkaynak—Wright, 253–275. Economics 2023.

Chaudhary, J. (2025) *Algorithmic foundations for generalizable artificial intelligence models: A Multi-Domain study*. Turku, Finland.

Chen, S. —Chen, H. (2007) Oil prices and real exchange rates. *Energy Economics*, Vol. 29(3), 390–404. <https://doi.org/10.1016/j.eneco.2006.08.003>

Cheridito, P.—Jentzen, A. —Rossmannek, F. (2022) Landscape Analysis for Shallow Neural Networks: Complete Classification of Critical Points for Affine Target Functions. *Journal of Nonlinear Science*, Vol. 32(5), 63–107. < https://doi.org/10.1007/s00332-022-09823-8>

Chicago Federal Reserve Bank (2020). The Federal Reserve's Dual Mandate. Retrieved February 19, 2025, <https://www.chicagofed.org/research/dual-mandate/dual-mandate>

Colah (2014) Conv Nets: A Modular Perspective—Colah's blog. Chrisopher Olah. <https://colah.github.io/posts/2014-07-Conv-Nets-Modular/>, Retrieved November 20, 2025.

Colah (2015) Understanding LSTM Networks—Colah's blog. Chrisopher Olah. <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>, Retrieved November 5, 2025.

Costa, J. —Silva, C. —Antunes, M.—Ribeiro, B. (2018) Adaptive Learning Models Evaluation in Twitter's Timelines. 2018 International Joint Conference on Neural Networks (IJCNN), 1–8. <https://doi.org/10.1109/IJCNN.2018.8489275>

Cormun, V. —Ristolainen, K. (2024). Exchange Rate Narratives. Bank of Finland Research Discussion Paper No. 11/2024. <https://doi.org/10.2139/ssrn.5010399>, Retrieved September 28, 2025.

CRAN (2015) Quanteda: Quantitative Analysis of Textual Data. Benoit, K. —Watanabe, K. —Wang, H. —Nulty, P. —Obeng, A. —Müller, S. —Matsuo, A. —Lowe, W. Retrieved April 29, 2025, <https://doi.org/10.32614/CRAN.package.quanteda>

Dael, F. A. —Avvad, H. (2025) Optimizing Random Forest Hyperparameters for Enhanced Stock Price Prediction. Int. Conf. New Trends Computer Science, ICTCS, 460–465. <https://doi.org/10.1109/ICTCS65341.2025.10989295>

Dhamija, A —Bhalla, V. (2010) Financial time series forecasting: Comparison of neural networks and ARCH models. *International Research Journal of Finance and Economics*, Vol. 49, 194–212.

Dedola, L. — Georgiadis, G. — Gräb, J. — Mehl, A. (2021) Does a big bazooka matter? Quantitative easing policies and exchange rates. *Journal of Monetary Economics*, Vol. 117, 489–506. <https://doi.org/10.1016/j.jmoneco.2020.03.002>

Deisenroth, M. —Faisal, A. —Ong, C. (2020) *Mathematics for Machine Learning*. 2. Cambridge University Press. <https://doi.org/10.1017/9781108679930>

Dietterich, T. (1998) Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, Vol. 10(7) 1895–1923

Devlin, J.—Chang, M.W. —Lee, K. —Toutanova, K. (2019) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*. <https://doi.org/10.48550/arXiv.1810.04805>

Diebold, F. X.— Mariano, R. S. (1995) Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, Vol 13(3), 253–263. <https://doi.org/10.2307/1392185>

Ding, H.—Zhao, X., — Jiang, Z. — Abdullah, S. N.— Dewi, D. A. (2024) EUR/USD Exchange Rate Forecasting Based on Information Fusion with Large Language Models and Deep Learning Methods. *arXiv*. <https://doi.org/10.48550/arXiv.2408.13214> . Working paper.

Dukascopy Bank (2025) Historical data feed. Retrieved February 4, 2025, from <https://www.dukascopy.com/swiss/english/marketwatch/historical>

Edwards, A. (1948). Note on the "correction for continuity" in testing the significance of the difference between correlated proportions. *Psychometrika*, Vol. 13(3), 185–187.

Eichler, S.—Littke, H. (2018) Central bank transparency and the volatility of exchange rates. *Journal of International Money and Finance*, Vol. 89, 23–49. <https://doi.org/10.1016/j.jimonfin.2018.07.006>

El Zaar, A. —Mansouri, A. —Benaya, N. —Bakir, T. —El Allati, A. (2025) Hybrid Transformer-CNN architecture for multivariate time series forecasting: Integrating attention mechanisms with convolutional feature extraction. *Journal of Intelligent Information Systems*, Vol. 63(4), 1233–1264. <https://doi.org/10.1007/s10844-025-00937-5>

Escudero, P.—Alcocer, W.—Paredes, J. (2021) Recurrent Neural Networks and ARIMA Models for Euro/Dollar Exchange Rate Forecasting. *Applied Sciences*, Vol. 11(12), 5658. <https://doi.org/10.3390/app11125658>

Euromoney FX Survey (2022) FX: Triennial BIS survey has market players mulling settlement risk and attractions of listed products. *Euromoney* 1.11.2022.

European Central Bank (2009) Structural Breaks, Cointegration and the Fisher Effect. Working Papers Series No 1013. <https://doi.org/10.2139/ssrn.1333613>, Retrieved February 24, 2025.

European Central Bank (2018) Exchange rate forecasting on a napkin. Working Papers Series No 2151. <https://data.europa.eu/doi/10.2866/204062>, Retrieved February 27, 2025

European Central Bank (2020) The predictive power of equilibrium exchange rate models. Working Papers Series No 2358.

<https://www.ecb.europa.eu/pub/pdf/scpwps/ecb.wp2358~4382d88430.en.pdf>, Retrieved February 12, 2025.

European Central Bank (2021) Introduction. Retrieved February 9, 2025, from <https://www.ecb.europa.eu/mopo/intro/html/index.en.html>

European Central Bank (2025) Euro area yield curves. Retrieved February 19, 2025, from <https://www.ecb.europa.eu/stats/financial_markets_and_interest_rates/euro_area_yield_curves/html/index.en.html>

Fagerland, M. —Lydersen, S.—Laake, P. (2013) The McNemar test for binary matched-pairs data: Mid-p and asymptotic are better than exact conditional. BMC Medical Research Methodology, Vol. 13, 91.<https://doi.org/10.1186/1471-2288-13-91>

Fama, E. (1970) Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance*, Vol. *25*(2), 383–417.

Fassas, A. P.—Kenourgios, D. —Papadamou, S. (2021) U.S. unconventional monetary policy and risk tolerance in major currency markets. *The European Journal of Finance*, Vol 27(10), 994–1008. <https://doi.org/10.1080/1351847X.2020.1775105>

Fawagreh, K. —Gaber, M. —Elyan, E. (2014) Random forests: From early developments to recent advancements. *Systems Science & Control Engineering*, Vol. *2*(1), 602–609. <https://doi.org/10.1080/21642583.2014.956265>

Fawcett, T. (2006) An introduction to ROC analysis. *Pattern Recognition Letters*, Vol. 27(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>

Federal Reserve (2010) Tips from TIPS: the informational content of Treasury Inflation-Protected Security prices. Finance and Economics Discussion Series Divisions of Research & Statistics and Monetary Affairs, Federal Reserve Board, Washington, D.C. Retrieved April 24, 2025, <https://www.federalreserve.gov/pubs/feds/2010/201019/201019pap.pdf>

Federal Reserve (2021) *The Fed Explained: What the Central Bank Does*. 11. Board of Governors of the Federal Reserve System. Washington, DC. <https://doi.org/10.17016/0199-9729.11>

Federal Reserve (2024) Federal Open Market Committee, Materials. Retrieved and Web Scraped January 5, 2025, <https://www.federalreserve.gov/monetarypolicy/materials>

Federal Reserve (2025a) Federal Open Market Committee. Retrieved February 3, 2025, <https://www.federalreserve.gov/monetarypolicy/fomc.htm>

Federal Reserve (2025b) Federal Open Market Committee Releases. Retrieved March 3, 2025, <https://www.federalreserve.gov/monetarypolicy/fomc_historical.htm>

Federal Reserve (2025c) Minutes of the Federal Open Market Committee. Retrieved May 9, 2025, <https://www.federalreserve.gov/monetarypolicy/fomcminutes20250319.htm>

Federal Reserve (2025d) Federal Reserve issues FOMC statement. Retrieved May 9, 2025, <https://www.federalreserve.gov/newsevents/pressreleases/monetary20250319a.htm>

Federal Reserve (2026) Federal Reserve issues FOMC statement. Retrieved May 1, 2026, <https://www.federalreserve.gov/newsevents/pressreleases/monetary20260429a.htm>

Ferrara, F. M.—Masciandaro, D. — Moschella, M. — Romelli, D. (2022) Political voice on monetary policy: Evidence from the parliamentary hearings of the European Central Bank. *European Journal of Political Economy*, 74, 102143. <https://doi.org/10.1016/j.ejpoleco.2021.102143>

Fisichella, M. —Garolla, F. (2021) Can Deep Learning Improve Technical Analysis of Forex Data to Predict Future Price Movements? *IEEE Access*, Vol. 9, 153083–153101. <https://doi.org/10.1109/ACCESS.2021.3127570>

Fontanari, T. —Fróes, T. —Recamonde-Mendoza, M. (2022) Cross-validation Strategies for Balanced and Imbalanced Datasets. In *Intelligent Systems*, eds. Xavier-Junior—Rios. 626–640. Springer International Publishing. <https://doi.org/10.1007/978-3-031-21686-2_43>

Foster, D. —Greenberg, S. —Kale, S. —Luo, H. —Mohri, M. —Sridharan, K. (2020) Hypothesis Set Stability and Generalization. arXiv:1904.04755. *arXiv*. <https://doi.org/10.48550/arXiv.1904.04755>

Galeshchuk, S.—Mukherjee, S. (2017) Deep networks for predicting direction of change in foreign exchange rates. *Intelligent Systems in Accounting, Finance and Management*, Vol. 24(4), 100–110. <https://doi.org/10.1002/isaf.1404>

Gimeno, R.—Ibáñez, A. (2018) The eurozone (expected) inflation: An option's eyes view. *Journal of International Money and Finance*, Vol. 86, 70–92. <https://doi.org/10.1016/j.jimonfin.2018.03.018>

Giacomini, R. —White, H. (2006) Tests of Conditional Predictive Ability. *Econometrica*, Vol. 74(6), 1545–1578. <https://doi.org/10.1111/j.1468-0262.2006.00718>

Graves, A. (2014) Generating Sequences With Recurrent Neural Networks. *arXiv*. <https://doi.org/10.48550/arXiv.1308.0850>

Griewank, A. (2012) Who invented the reverse mode of differentiation? In *Documenta, ed.* Grötschel M., *Mathematica Series,* 389–400. Vol. 6(1), EMS Press. <https://doi.org/10.4171/dms/6/38>

Guyard, K.—Deriaz, M. (2024) Predicting Foreign Exchange EUR/USD Direction Using Machine Learning. Proceedings of the 2024 7th International Conference on Machine Learning and Machine Intelligence (MLMI), 1–9. <https://doi.org/10.1145/3696271.3696272>

Gössi, S. —Chen, Z. — Kim, W. — Bermeitinger, B. — Handschuh, S. (2023) FinBERT-FOMC: Fine-Tuned FinBERT Model with Sentiment Focus Method for Enhancing Sentiment Analysis of FOMC Minutes. 4th ACM International Conference on AI in Finance, 357–364. <https://doi.org/10.1145/3604237.3626843>

Handlan, A. (2020) Text shocks and monetary surprises: Text analysis of fomc statements with machine learning. Published Manuscript. Retrieved January 9, 2025, <https://handlanamy.github.io/MyFiles/Handlan_TextShocks_sub.pdf>

Hansen, S. —McMahon, M. (2016) Shocking language: Understanding the macroeconomic effects of central bank communication. *Journal of International Economics*, Vol. 99, 114-133.

Hansun, S. (2013) A new approach of moving average method in time series analysis. 2013 International Conference on New Media Studies, CoNMedia 2013. <https://doi.org/10.1109/conmedia.2013.6708545>

Hansun, S.—Kristanda, M. (2017) Performance analysis of conventional moving average methods in forex forecasting. 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems, 11–17. <https://doi.org/10.1109/ICON-SONICS.2017.8267814>

Hatemi, A. (2009) The International Fisher Effect: Theory and application. *Investment Management and Financial Innovations*, Vol. 6(1), 117–121.

Hendershott, T. —Jones, C. —Menkveld, A. (2011) Does Algorithmic Trading Improve Liquidity? *The Journal of Finance*, Vol. 66(1), 1–33. <https://doi.org/10.1111/j.1540-6261.2010.01624.x>

Hernández-Murillo, R. — Shell, H. (2014) The rising complexity of the FOMC statement. *Economic Synopses*, Vol. 23(2). <https://fraser.stlouisfed.org/title/economic-synopses-6715/rising-complexity-fomc-statement-624434>

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

Houcine, B. —Zouheyr, G. —Abdessalam, B. —Youcef, H.—Hanane, A. (2020) The Relationship Between Crude Oil Prices, EUR/USD Exchange Rate and Gold Prices. *International Journal of Energy Economics and Policy*, Vol. 10(5), 234-242.

Hsing, Y. —Sergi, B. (2009) The dollar/euro exchange rate and a comparison of major models. *Journal of Business Economics and Management*, Vol. 10(3). <https://doi.org/10.3846/1611-1699.2009.10.199-205>

Hull, J. (2017*) Options, Futures, and Other Derivatives*, *Global Edition*. 9. Pearson Education, Limited.

Hämäläinen, J. (2015) *Portfolio selection under directional return predictability*. Turku, Finland. <https://www.finna.fi/Record/utupub_diss.11111_27193>

IMF (2023) Global Spillovers of the Fed Information Effect, Retrieved March 23, 202, from <https://link.springer.com/content/pdf/10.1057/s41308-023-00210-1.pdf? >

IMF (2024) Annual Report on Exchange Arrangements and Exchange Restrictions 2023. Retrieved February 19, 2025, from <https://www.imf.org/en/Publications/Annual-Report-on-Exchange-Arrangements-and-Exchange-Restrictions/Issues/2024/12/19/Annual-Report-on-Exchange-Arrangements-and-Exchange-Restrictions-2023-541890>

Ireland, P. (2024) The devolution of federal reserve monetary policy strategy, 2012–24. *Southern Economic Journal*. <https://doi.org/10.1002/soej.12730>

Ito, T. —Takeda, F. (2022) Do sentiment indices always improve the prediction accuracy of exchange rates? *Journal of Forecasting*, Vol 41(4), 840–852. <https://doi.org/10.1002/for.2836>

Ito, T. —Masuda, M.— Naito, A. — Takeda, F. (2021) Application of Google Trends-based sentiment index in exchange rate prediction. *Journal of Forecasting*, Vol 40(7), 1154–1178. <https://doi.org/10.1002/for.2762>

Jegadeesh, N. —Wu, D. (2017) Deciphering Fedspeak: The information content of FOMC meetings. Retrieved March 26, 2025, <https://www.academia.edu/96739673/Deciphering_Fedspeak_The_Information_Content_of_FOMC_Meetings >[83]

Jitmaneeroj, B.—Lamla, M.—Wood, A. (2019) The implications of central bank transparency for uncertainty and disagreement. *Journal of International Money and Finance*, Vol. 90, 222–240. <https://doi.org/10.1016/j.jimonfin.2018.10.002>

Jubinski, D. — Tomljanovich, M. (2017) Central Bank Actions and Words: The Intraday Effects of FOMC Policy Communications on Individual Equity Volatility and Returns. *Financial Review*, 52(4), 701–724. <https://doi.org/10.1111/fire.12143>

Kampouridis, M., — Otero, F. (2017) Evolving trading strategies using directional changes. *Expert Systems with Applications*, Vol. 73, 145–160. <https://doi.org/10.1016/j.eswa.2016.12.032>

Kansas City Federal Reserve (2016) A Corollary of Accountability: A History of FOMC Policy Communication. Retrieved March 12, 2025, <https://www.kansascityfed.org/AboutUs/documents/6512/acorollaryofaccountability.pdf>

83 This was a SSRN paper that was removed from publication at the request of the author. I used only its descriptions of the four sections of FOMC Minutes, as I found them to depict the contents of these FOMC documents quite accurately.

Kansoy, F. (2020) FOMC Minutes: As a Source of Central Bank Communication Surprise. <https://warwick.ac.uk/fac/soc/economics/research/workingpapers/2022/twerp_1436_-_kansoy.pdf>

Kavkler, A.—Boršič, D. —Bekő, J. (2016) Is the PPP valid for the EA-11 countries? New evidence from nonlinear unit root tests. *Economic Research-Ekonomska Istraživanja*, Vol. 29(1), 612–622. <https://doi.org/10.1080/1331677X.2016.1189842>

Kelany, O.—Aly, S. —Ismail, M. (2020) Deep Learning Model for Financial Time Series Prediction. 2020 14th International Conference on Innovations in Information Technology (IIT), 120–125. <https://doi.org/10.1109/IIT50501.2020.9299063>

Kim, W. —Spörer, J. — Lee, C. L. — Handschuh, S. (2024) Is Small Really Beautiful for Central Bank Communication? Evaluating Language Models for Finance: Llama-3-70B, GPT-4, FinBERT-FOMC, FinBERT, and VADER. Proceedings of the 5th ACM International Conference on AI in Finance, 626–633. <https://doi.org/10.1145/3677052.3698675>

Koelbl, M. — Laschinger, R. — Steininger, B. I. — Schaefers, W. (2024) Revealing the risk perception of investors using machine learning. *The European Journal of Finance*. <https://www.tandfonline.com/doi/abs/10.1080/1351847X.2024.2364831>

Lang, N. (2022) From Vectors to Tensors: Exploring the Mathematics of Tensor Algebra. TDS Archive 20.12.2022. Retrieved October 12, 2025, from <https://medium.com/data-science/what-are-tensors-in-machine-learning-5671814646ff>

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

Leung, J. —Chong, T. (2003) An empirical comparison of moving average envelopes and Bollinger Bands. *Applied Economics Letters*, Vol. 10(6), 339–341. <https://doi.org/10.1080/1350485022000041032>

Liu, L.— Patton, A. —Sheppard, K. (2015) Does anything beat 5-minute RV? A comparison of realized measures across multiple asset classes. *Journal of Econometrics*, 187(1), 293–311. <https://doi.org/10.1016/j.jeconom.2015.02.008>

Lin, T.-Y. —Goyal, P. —Girshick, R. —He, K. —Dollár, P. (2018). Focal Loss for Dense Object Detection. arXiv:1708.02002. arXiv. <https://doi.org/10.48550/arXiv.1708.02002>

Ling, J.—Tsui, A.—Zhang, Z. (2021) Trading Macro-Cycles of Foreign Exchange Markets Using Hybrid Models. *Sustainability*, Vol. *13*(17), Article 17. <https://doi.org/10.3390/su13179820>

Loginov, A. —Wilson, G. —Heywood, M. (2015) Better trade exits for foreign exchange currency trading using FXGP. 2015 IEEE Congress on Evolutionary Computation (CEC), 2510–2517. <https://doi.org/10.1109/CEC.2015.7257197>

Lothian, J. —Taylor, M. (1996) Real Exchange Rate Behavior: The Recent Float from the Perspective of the Past Two Centuries. *Journal of Political Economy*, Vol. 104(3), 488–509

Louppe, G. (2015) Understanding Random Forests: From Theory to Practice. *arXiv*. <https://doi.org/10.48550/arXiv.1407.7502>

Luangluewut, W. —Thiennviboon, P. (2023) Forex Price Trend Prediction using Convolutional Neural Network. *2023* 20th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 1–4. <https://doi.org/10.1109/ECTI-CON58255.2023.10153142>

Lunsford, K. (2020) Policy Language and Information Effects in the Early Days of Federal Reserve Forward Guidance. *American Economic Review*, Vol. 110(9), 2899–2934. <https://doi.org/10.1257/aer.20181721>

Malkiel B. (1973) *A Random Walk Down Wall Street*. 1. W. W. Norton & Co., NY, United States of America

Manahov, V. —Hudson, R. (2014) The implications of high-frequency trading on market efficiency and price discovery. *Applied Economics Letters*, Vol. 21(16), 1148–1151. <https://doi.org/10.1080/13504851.2014.914135>

Markova, M. (2023) Forex Time Series Forecasting Using Hybrid Convolutional Neural Network/Long Short-Term Memory Network Model. Springer Proceedings in Mathematics and Statistics, 412, 295–305. <https://doi.org/10.1007/978-3-031-21484-4_26>

Mimno, D. —Wallach, H.—Talley, E.—Leenders, M.—McCallum, A. (2011) Optimizing semantic coherence in topic models. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 262–272. <https://dl.acm.org/doi/10.5555/2145432.2145462>

Mohammadi, M. —Hofman, W. —Tan, Y. (2019) A Comparative Study of Ontology Matching Systems via Inferential Statistics. IEEE Transactions on Knowledge and Data Engineering, Vol. 31(4), 615–628. <https://doi.org/10.1109/TKDE.2018.2842019>

Mohana, H.—Suriakala, M. (2018) Integrated Cosine And Tuned Cosine Similarity Measure To Alleviate Data Sparsity Issues For Personalized Recommendation. 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), 41–49. <https://doi.org/10.1109/CSITSS.2018.8768763>

Mohri, M. —Rostamizadeh, A. —Talwalkar, A. (2018) *Foundations of machine learning*. 2. MIT Press, Cambridge, MA, United States of America

Mueller, P. —Tahbaz-Salehi, A. —Vedolin, A. (2017) Exchange Rates and Monetary Policy Uncertainty. *The Journal of Finance*, Vol. 72(3), 1213–1252. <https://doi.org/10.1111/jofi.12499>

Nakkiran, P. —Kaplun, G—Bansal, Y. —Yang, T. —Barak, B. —Sutskever, I. (2019) Deep Double Descent: Where Bigger Models and More Data Hurt (No. arXiv:1912.02292). *arXiv*. <https://doi.org/10.48550/arXiv.1912.02292>

Newey W —West K (1987) A Simple, Positive Semi-Definite, Heteroskedasticity and Autocorrelation Consistent Covariance Matrix. *Econometrica*, Vol. 55, 703–708.

New York Federal Reserve (2006). The Information Content of FOMC Minutes. Research Paper. Retrieved March 12, 2025, from <https://doi.org/10.2139/ssrn.922312>

New York Federal Reserve (2011) The Pre-FOMC Announcement Drift. Staff Report. Retrieved March 12, 2025, from <https://www.newyorkfed.org/medialibrary/media/research/staff_reports/sr512.pdf>

New York Federal Reserve (2024) Volume Survey—Foreign Exchange Committee. Retrieved February 3, 2025, from <https://www.newyorkfed.org/fxc/fx-volume-survey>

New York Federal Reserve (2026) Central Bank Swap Arrangements. Retrieved January 8, 2026, from < https://www.newyorkfed.org/markets/international-market-operations/central-bank-swap-arrangements>

Nonejad, N. (2021) The price of crude oil and (conditional) out-of-sample predictability of world industrial production. *Journal of Commodity Markets*, Vol. 23. <https://doi.org/10.1016/j.jcomm.2021.100167>

Osler, C. (2003) Currency Orders and Exchange Rate Dynamics: An Explanation for the Predictive Success of Technical Analysis. *The Journal of Finance*, Vol. 58(5), 1791–1819. <https://doi.org/10.1111/1540-6261.00588>

Osowska, E. — Wójcik, P. (2024) Predicting the reaction of financial markets to Federal Open Market Committee post-meeting statements. *Digital Finance*, Vol 6(1), 145–175. <https://doi.org/10.1007/s42521-023-00096-8>

Oya, K. (2011) Bias-corrected realized variance under dependent microstructure noise, *Mathematics and Computers in Simulation*, Vol. 81(7), pp. 1290–1298. <https://doi.org/10.1016/j.matcom.2010.04.017>

Park, M. —Hastie, T. (2008) Penalized logistic regression for detecting gene interactions. *Biostatistics*, Vol. 9(1), 30–50. <https://doi.org/10.1093/biostatistics/kxm010>

Plíhal, T. (2021) Scheduled macroeconomic news announcements and Forex volatility forecasting. *Journal of Forecasting*, Vol. 40(8), 1379–1397. <https://doi.org/10.1002/for.2773>

Prince, S. (2023) *Understanding Deep Learning*. MIT Press, Cambridge, MA, the United States of America. <http://udlbook.com>

Rai, A. —Borah, S. (2021) Study of Various Methods for Tokenization. In *Applications of Internet of Things,* eds Mandal, J.—Mukhopadhyay, S.—Roy, A., 193–200. Springer. <https://doi.org/10.1007/978-981-15-6198-6_18>

Ristolainen, K.—Roukka, T.—Nyberg, H. (2024) A thousand words tell more than just numbers: Financial crises and historical headlines. *Journal of Financial Stability*, Vol. *70*, 101209. <https://doi.org/10.1016/j.jfs.2023.101209>

Roberts, M. —Stewart, B.—Airoldi, E. (2016) A Model of Text for Experimentation in the Social Sciences. *Journal of the American Statistical Association*, Vol. *111*(515), 988–1003. <https://doi.org/10.1080/01621459.2016.1141684>

Roberts, M.—Stewart, B.—Tingley, D. (2019) stm: An R Package for Structural Topic Models. *Journal of Statistical Software*, Vol. 91, 1–40. <https://doi.org/10.18637/jss.v091.i02>

Rokach, L. (2010) Ensemble-based classifiers. *Artificial Intelligence Review*, Vol. *33*(1), 1–39. <https://doi.org/10.1007/s10462-009-9124-7>

Rogoff, K. (1996). The Purchasing Power Parity Puzzle. *Journal of Economic Literature*, Vol. 34(2), 647–668.

Rosa, C. (2013) The Financial Market Effect of FOMC Minutes. Research paper. New York Federal Reserve.

Safi, S.—Aliyu, S.—Ibrahim, K.—Sanusi, O. (2022) Can Oil Price Predict Exchange Rate? Empirical Evidence from Deep Learning. *International Journal of Energy Economics and Policy*, Vol. 12(4), 482–493. <https://doi.org/10.32479/ijeep.13200>

Schlossberg, B. (2006) Technical analysis of the currency market. Wiley trading, Hoboken, NJ.

Schmidhuber, J. (2015) Deep Learning in Neural Networks: An Overview. *Neural Networks*, Vol 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>

Shah, A.—Paturi, S.—Chava, S. (2023) Trillion Dollar Words: A New Financial Dataset, Task & Market Analysis (SSRN Scholarly Paper No. 4447632). *Social Science Research Network*. <https://papers.ssrn.com/abstract=4447632>

Shapiro, A.—Hanouna, P. (2020) *Multinational Financial Management*. 11. Wiley, Hoboken, NJ, United States of America.

Shawi, R.—Bahman, M.—Sakr, S. (2025) To tune or not to tune? An approach for recommending important hyperparameters for classification and clustering algorithms. *Future Generation Computer Systems*, Vol. *163*, 107524. <https://doi.org/10.1016/j.future.2024.107524>

Sohangir, S.—Wang, D. (2017). Improved sqrt-cosine similarity measurement. *Journal of Big Data*, Vol. *4*(1), 25. <https://doi.org/10.1186/s40537-017-0083-6>

Sosvilla-rivero, S.—García, E. (2005) Forecasting the dollar/euro exchange rate: Are international parities useful? *Journal of Forecasting*, Vol. 24(5), 369–377. <https://doi.org/10.1002/for.955>

S&P Global (2025) S&P U.S. Indices Methodology. Retrieved March 23, 2025, from <https://www.spglobal.com/spdji/en/documents/methodologies/methodology-sp-us-indices.pdf>

St. Louis Federal Reserve (2019) Here's How to Read an FOMC Statement. Retrieved March 12, 2025, from <https://www.stlouisfed.org/open-vault/2019/may/how-read-fomc-statement>

St. Louis Fed (2025) 10-Year Breakeven Inflation Rate. Retrieved March 6, 2025, <https://fred.stlouisfed.org/series/T10YIE>

Sundjaja, J. —Shrestha, R. —Krishan, K. (2025) McNemar and Mann-Whitney U Tests. In *StatPearls.* StatPearls Publishing, FL, United States of America. <http://www.ncbi.nlm.nih.gov/books/NBK560699/>

Swift (2025) About us. Retrieved February 17, 2025, from <https://www.swift.com/about-us>

Tadle, R. C. (2022) FOMC minutes sentiments and their impact on financial markets. *Journal of Economics and Business*, Vol. 118. <https://doi.org/10.1016/j.jeconbus.2021.106021>

Taye, M. (2023) Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions. *Computation*, Vol. 11(3), 52. <https://doi.org/10.3390/computation11030052>

Tse, Y. (2019) The impact of FOMC announcements on currency futures markets. *Applied Economics Letters*, Vol. 26(19), 1590–1596. <https://doi.org/10.1080/13504851.2019.1588940>

United States Treasury (2025) Resource Center. Daily Treasury Par Yield Curve Rates. Retrieved February 20, 2025, from <https://home.treasury.gov/resource-center/data-chart-center/interest-rates/TextView>

Van Hoa, T.—Tuan Anh, D. —Ngoc Hieu, D. (2021) Foreign Exchange Rate Forecasting using Autoencoder and LSTM Networks. Proceedings of the 2021 6th International Conference on Intelligent Information Technology, 22–28. <https://doi.org/10.1145/3460179.3460184>

Vaswani, A. —Shazeer, N. —Parmar, N. —Uszkoreit, J. —Jones, L. —Gomez, A. —Kaiser, Ł. —Polosukhin, I. (2017) Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems, 6000–6010.

Vega, C. — Scotti, C. — Schmanski, B. (2024) Fed Communication, News, and Disagreement (SSRN Scholarly Paper No. 5072648). Social Science Research Network. <https://doi.org/10.2139/ssrn.5072648>

Vochozka, M. — Rowland, Z. — Suler, P. — Marousek, J. (2020) The influence of the international price of oil on the value of the EUR/USD exchange rate. *Journal of Competitiveness*, Vol. 12(2), 167–190. <https://doi.org/10.7441/joc.2020.02.10>

Vyshnevskyi, I.—Jombo, W.—Sohn, W. (2024) The clarity of monetary policy communication and financial market volatility in developing economies. *Emerging Markets Review*, Vol. 59, 101121. <https://doi.org/10.1016/j.ememar.2024.101121>

Wang, J. —Yang, M. (2011) Housewives of Tokyo versus the gnomes of Zurich: Measuring price discovery in sequential markets. *Journal of Financial Markets*, Vol. 14(1), 82–108. <https://doi.org/10.1016/j.finmar.2010.08.002>

White, H. (2000) A Reality Check for Data Snooping. *Econometrica*, Vol. 68(5), 1097–1126.

Wu, S. — Irsoy, O. — Lu, S. — Dabravolski, V. — Dredze, M. — Gehrmann, S. — Kambadur, P. — Rosenberg, D. — Mann, G. (2023) BloombergGPT: A Large Language Model for Finance. *arXiv*. <https://doi.org/10.48550/arXiv.2303.17564>

Xu, Z. (2003) Purchasing power parity, price indices, and exchange rate forcasts. *Journal of International Money and Finance*, Vol. 22(1), 105–130. <https://doi.org/10.1016/S0261-5606(02)00049-9>

Yae, J. (2024) Unintended look-ahead bias in out-of-sample forecasting. *Applied Economics Letters*, Vol. 31(10), 953–957. <https://doi.org/10.1080/13504851.2022.2159002>

Yıldırım, D.—Toroslu, I.— Fiore, U. (2021) Forecasting directional movement of Forex data using LSTM with technical and macroeconomic indicators. *Financial Innovation*, Vol. 7(1), 1. <https://doi.org/10.1186/s40854-020-00220-2>

Yono, K. — Izumi, K. — Sakaji, H. — Matsushima, H. — Shimada, T. (2019) Extraction of Focused Topic and Sentiment of Financial Market by using Supervised Topic Model for Price Movement Prediction. IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr), Vol. 1–7. <https://doi.org/10.1109/CIFEr.2019.8759119>

Zhang, L.—Mykland, P.A.—Aït-Sahalia, Y. (2005) A Tale of Two Time Scales, *Journal of the American Statistical Association*, Vol. 100(472), pp. 1394–1411. <https://doi.org/10.1198/016214505000000169>

Zhang, J.—Feng, L.—He, Y.—Wu, Y.—Dong, Y. (2023) Temporal Convolutional Explorer Helps Understand 1D-CNN's Learning Behavior in Time Series Classification from Frequency

Domain. Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, 3351–3360. <https://doi.org/10.1145/3583780.3615076>

Zhou, S.—Bahmani-Oskooee, M. —Kutan, A. (2008) Purchasing Power Parity before and after the Adoption of the Euro. *Review of World Economics*, Vol. 144(1), 134–150. <https://doi.org/10.1007/s10290-008-0140-5>

Zhu, X. (2023) Volume dynamics around FOMC announcements. BIS. Working Paper.

Zeileis, A. (2004) Econometric Computing with HC and HAC Covariance Matrix Estimators. *Journal of Statistical Software*, Vol. *11*, 1–17. <https://doi.org/10.18637/jss.v011.i10>

Appendices

Appendix 1: Brief Note on Notation and Data

In order to accurately discuss implementations of ML models in a compact, informative, and unambiguous manner, mathematical representations are employed. In the spirit of Deisenroth et al. (2020) and Alpaydin (2020) this thesis primarily uses vector-matrix notation to concisely represent the numeric data typical to ML. In this sense, the vectors presented can be interpreted as arrays of numbers containing stored data, or mathematical objects obeying rules of addition and scaling.

Notation in this thesis deviates slightly from the conventions of literature. Consequently, some key notational elements of this thesis are presented here to clarify the discussion on ML methods. The transpose operation is denoted with $'$. Vectors and matrices are represented with bold letters, with matrices capitalized. Expressions and formalizations are emphasized in italics. For example, matrix $\boldsymbol{I}_n \in \mathbb{R}^{n\times n}$ is the $n \times n$ identity matrix, and its columns, i.e. standard unit vectors, would be represented as $\boldsymbol{e}_1 = (1,0,\dots,0)' \in \mathbb{R}^n$. Also, the expected value, indicator functions, and number sets are represented with blackboard bold as $\mathbb{E}$, $\mathbb{1}$ and $\mathbb{R}$ respectively. Additionally, functions are denoted in italics e.g. $f(\cdot)$. The composition operator $\circ$, is used to simplify nested functions such that $(f \circ g)(x) = f\big(g(x)\big)$.

Tensors are multidimensional arrays where the dimensionality of a tensor is defined by its order. For example, an o-order tensor is single scalar value, a 1st order tensor can be regarded as vector, and 2nd order tensor as a matrix (Yuwang et al. 2019, 162952). For example, a tensor representing a red-green-blue color image of size 3x3 pixels could be seen as stack of three matrices (see Figure 30).

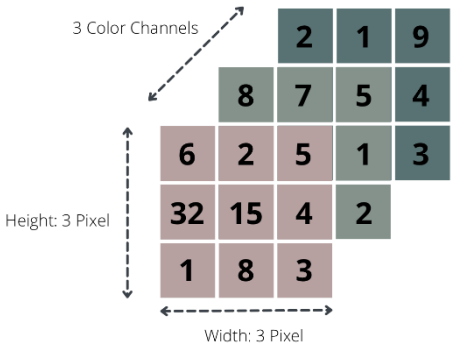

Figure 30: A 3rd Order Tensor Visualized (Lang 2022)

In this thesis, the tensor representation of the image would be denoted as $\boldsymbol{A}_{image} \in \mathbb{R}^{3\times3\times3}$. Each element in Figure 30 represents the intensity of the three colors in each pixel of the original image. In this thesis tensors refer typically to a 3rd order tensor.

Inputs for models are typically expressed as data matrix $\boldsymbol{X} \in \mathbb{R}^{N \times D}$, an array of *N*-rows of instances[84] or observed input vectors $\boldsymbol{x}_t$ such that $\boldsymbol{X} = (\boldsymbol{x}_1 | \boldsymbol{x}_2 | \ldots | \boldsymbol{x}_t | \ldots | \boldsymbol{x}_N)'$. Each input vector consists of real numbers, called features,[85] which describe the data at instance *t*. Deviating from typical notation, values of individual input variables are usually denoted with $x_{d,t}$ instead of $x_i^{(t)}$. Here subscript *d* is viewed as the column or feature index, and superscript *(t)* as the instance or row index. Each row of the data matrix has a *D* number of features. When superscripts are used, they are distinguishable from exponentiation by brackets. In supervised learning, the label[86] is the data vector the model aims to predict. The label is usually denoted as $\boldsymbol{y} \in \mathbb{R}^N$. In the case of discreate label, such as in classification, the label is usually denoted as $\boldsymbol{c} \in \mathcal{Y}^N$. Instances of the label vector are viewed as belonging to a common label space $c_t \in \mathcal{Y}$.[87] Furthermore, labels and input data are usually paired by instance *t* as datasets $\mathcal{D} = \{\{\boldsymbol{x}_1, y_1\}, \{\boldsymbol{x}_2, y_2\}, \ldots, \{\boldsymbol{x}_N, y_N\} \ldots, \{\boldsymbol{x}_T, y_T\}\}$. (Deisenroth et al. 2020, 225-230.) Predictions of models' are usually denoted by $\hat{y}_t$ for regression and $\hat{c}_t$ or $\hat{r}_t$ for classification.

Also, surrounding vertical lines $|\cdot|$ usually denote cardinality, and occasionally absolute value.

[84] Instances are single data points and analogous to observations in traditional statistics contexts.
[85] Features can be seen as synonymous to attributes or covariates (Deisenroth et al. 2020, 227).
[86] Labels are sometimes also called the target, dependent, or response variable (Deisenroth et al. 2020, 228).
[87] For a binary trade action variable, one could denote the label space as $\mathcal{Y} = \{buy, sell\}$.

Appendix 2: Illustration of Backpropagation

Let's consider a simple regression neural network with one node output layer L, a hidden layer $L-1$ with two neurons h_1 and h_2, and a three-node input layer $L-2$. The activation functions in the hidden layer $\boldsymbol{h}$ are sigmoid functions $\sigma(\cdot)$, and the output layer uses an identity function i.e. the raw score. Additionally, the model uses bias variables b. We can consider the weight vector for the ANN,

$$\boldsymbol{w} = \left(\underbrace{b_{L-2}^{(h_1)}, w_{L-2,a}^{(h_1)}, w_{L-2,c}^{(h_1)}, w_{L-2,d}^{(h_1)}}_{components\ of\ \mathrm{h}_1}, \underbrace{b_{L-2}^{(h_2)}, w_{L-2,a}^{(h_2)}, w_{L-2,c}^{(h_2)}, w_{L-2,d}^{(h_2)}}_{components\ of\ \mathrm{h}_2}, \underbrace{b_{L-1}^{(L)}, w_{h_1}^{(L)}, w_{h_2}^{(L)}}_{components\ of\ \hat{y}} \right)'.$$

From the input layer $L-2$ to the hidden layer $L-1$ the interim products z_j and h_j, $j = 1,2$, are computed using the input data $\boldsymbol{x}_t = (a_t, c_t, d_t)'$,

$$h_1 = \sigma(z_1), \quad z_1 = b_{L-2}^{(h_1)} + \boldsymbol{w}_{L-2}^{(h_1)\prime}\boldsymbol{x}_t,$$

$$h_2 = \sigma(z_2), \quad z_2 = b_{L-2}^{(h_2)} + \boldsymbol{w}_{L-2}^{(h_2)\prime}\boldsymbol{x}_t.$$

Here, the sigmoid activation function and its partial derivates are,

$$h_j = \sigma(z_j) = \left(1 + exp(-z_j)\right)^{-1}, \qquad \frac{\partial h_j}{\partial z_j} = h_j(1 - h_j).$$

The interim products are computed and fed forward to the next and final layer L such that,

$$\hat{y} = b_{L-1}^{(L)} + w_{h_1}^{(L)} h_1 + w_{h_2}^{(L)} h_2.$$

To optimize the parameters $\boldsymbol{w}$, gradient descent is employed. The gradients for each parameter are calculated w.r.t a loss function, for simplicity, one could select a scaled MSE: $\ell = \frac{1}{2}(y - \hat{y})^2$. Assuming we know the value of inputs, weights, interim products, labels, and predictions, the only thing left to do is to figure out the gradients of the gradient vector to use in gradient descent.

Applying the chain rule, gradients are defined one local gradient δ at a time, for every t separately,

$$\frac{\partial \ell}{\partial \hat{y}_t} = \hat{y}_t - y_t = \delta^{(\mathrm{L})},$$

Now, for instance t, the first local gradient $\delta^{(L)}$ is calculated, then we proceed and define the gradients of the parameters on layer L,

$$\frac{\partial \ell}{\partial b_{L-1}^{(L)}} = \frac{\partial \ell}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial b_{L-1}^{(L)}} = 1\delta^{(L)},$$

$$\frac{\partial \ell}{\partial w_{h_j}^{(L)}} = \frac{\partial \ell}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial w_{h_j}^{(L)}} = \delta^{(L)} h_j \qquad \forall \mathrm{j} = 1,2,$$

Thus, we have received the gradients of the weights on layer L. Next, we go a layer down, and the second local gradients $\delta_{\boldsymbol{h_j}}^{(L-1)}$ are computed. The next step splits down two paths, one for h_1 and h_2,

$$\delta_{\boldsymbol{h_j}}^{(L-1)} = \frac{\partial \ell}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial h_j}\frac{\partial h_j}{\partial z_j} = \delta^{(L)}\frac{\partial \hat{y}_t}{\partial h_j}\frac{\partial h_j}{\partial z_j} = \delta^{(L)} w_{h_j}^{(L)}\frac{\partial h_j}{\partial z_j} = \delta^{(L)} w_{h_j}^{(L)} h_j(1-h_j), \qquad \forall\, j = 1,2.$$

All of the numeric values needed to compute the 2nd local gradients are available. Thus, we can proceed to define these local gradients. Next, gradients for parameters on input layer, are evaluated, $(b_{L-2}^{(h_j)}, \boldsymbol{w}_{L-2}^{(h_j)})$, where $\boldsymbol{w}_{L-2}^{(h_j)} = (w_{L-2,a}^{(h_j)}, w_{L-2,c}^{(h_j)}, w_{L-2,d}^{(h_j)})'$.

$$\frac{\partial \ell}{\partial b_{L-2}^{(h_j)}} = \frac{\partial \ell}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial h_j}\frac{\partial h_j}{\partial z_j}\frac{\partial z_j}{\partial b_{L-2}^{(h_j)}} = \delta_{h_j}^{(L-1)}\frac{\partial z_j}{\partial b_{L-2}^{(h_j)}} = \delta_{h_j}^{(L-1)} \qquad \forall\, j = 1,2,$$

$$\frac{\partial \ell}{\partial \boldsymbol{w}_{L-2}^{(h_j)}} = \frac{\partial \ell}{\partial \hat{y}_t}\frac{\partial \hat{y}_t}{\partial h_j}\frac{\partial h_j}{\partial z_j}\frac{\partial z_j}{\partial \boldsymbol{w}_{L-2}^{(h_j)}} = \delta_{h_j}^{(L-1)}\frac{\partial z_j}{\partial \boldsymbol{w}_{L-2}^{(h_j)}} = \delta_{h_j}^{(L-1)}\mathbf{x}_\mathrm{t} \qquad \forall j = 1,2$$

Thus, using backpropagation we can efficiently, and step-by-step define the gradient vector's value,

$$\nabla_\mathrm{w}\ell = \left(\delta_{h_1}^{(L-1)},\ \delta_{h_1}^{(L-1)}a_t,\ \delta_{h_1}^{(L-1)}\mathrm{c_t},\ \delta_{h_1}^{(L-1)}d_\mathrm{t},\ \ \delta_{h_2}^{(L-1)},\ \delta_{h_2}^{(L-1)}a_t,\ \delta_{h_2}^{(L-1)}\mathrm{c_t},\ \delta_{h_2}^{(L-1)}d_\mathrm{t},\ \ \delta^{(L)}, \delta^{(L)}h_1,\ \delta^{(L)}h_2\right)'.$$

This example was inspired by Deisenroth et al. (2020, 140-143) and Aggarwal (2023, 46-52).

Appendix 3: Sequential Nature of Global Currency Markets

Table 19: Trading hours of Global FX Market Sessions (Adapted from Wang & Yang 2011)

Markets	UTC	London	Frankfurt/ Zurich (CET)	Hong Kong/ Singapore	Tokyo	Sydney	San Francisco	New York (ET)
ASIA	0	1	2	8	**9**	**10**	17	20
	1	2	3	**9**	**10**	**11**	18	21
	2	3	4	**10**	**11**	**12**	19	22
	3	4	5	**11**	**12**	**13**	20	23
	4	5	6	**12**	**13**	**14**	21	0
	5	6	7	**13**	**14**	**15**	22	1
	6	7	8	**14**	**15**	16	23	2
EUROPE	7	8	**9**	**15**	16	17	0	3
	8	**9**	**10**	16	17	18	1	4
	9	**10**	**11**	17	18	19	2	5
	10	**11**	**12**	18	19	20	3	6
	11	**12**	**13**	19	20	21	4	7
	12	**13**	**14**	20	21	22	5	8
Witching Hours	13	**14**	**15**	21	22	23	6	**9**
	14	**15**	16	22	23	0	7	**10**
US	15	16	17	23	0	1	8	**11**
	16	17	18	0	1	2	**9**	**12**
	17	18	19	1	2	3	**10**	**13**
	18	19	20	2	3	4	**11**	**14**
	19	20	21	3	4	5	**12**	**15**
	20	21	22	4	5	6	**13**	16
	21	22	23	5	6	7	**14**	17
	22	23	0	6	7	8	**15**	18
ASIA	23	0	1	7	8	**9**	16	19

It is notable that effects of DST are unaccounted for. European nations tend to implement daylight savings later than the United States,[88] thus the time difference between Europe and the United States varies slightly. This has some minor effects on the features based on ECB par yield releases. Also, In the structured dataset, the above markets were encoded as one-hot variables. The original Dukascopy dataset's UTC timestamps were used to encode the four trading sessions.

[88] ET is switched to EST (DST) on the 2nd Sunday of March; CET is swapped for CEST on the last Sunday of March. Also, EST is switched to ET on the 1st Sunday of November, whilst CEST to CET occurs on the last Sunday of October. Thus, the 6h difference is 5h: between the 2nd and last Sunday of March; and the last Sunday of October and the 1st Sunday of November. This was disregarded.

Appendix 4: Pre-Processing FOMC Documents for STM

Below is the R-Code used to pre-process the FOMC documents for STM modelling.

```
#### Get FOMC Data and Set-up Packages ####

# Packages

library(tidyverse)

library(quanteda)

library(stm)

library(ggrepel)


# Define file location of FOMC releases (Minutes & Statements)

path <- "C:/Users/pette/OneDrive/Työpöytä/TSE/Gradu/Datasets/new_fomc.csv"


# Load in the FOMC release data

FOMC <- read.csv(path);rm(path)


# subset to by document type

FOMC_s <- filter(FOMC, Type == "Statement")

FOMC_m <- anti_join(FOMC, FOMC_s)




#

##

###

#### Tokens to Remove ####
```

```
## Find removable objects

library(spacyr)

spacy_initialize(model = "en_core_web_sm")


# Run NER on FOMC full texts

parsed <- spacy_parse(FOMC$Text, entity = TRUE)


# Extract only tokens marked as PERSON entities

name_tokens <- parsed %>%

  filter(entity == "PERSON_B"| entity =="PERSON_I") %>%

  pull(token) %>%

  unique()



spacy_finalize()


write.table(name_tokens, file = "names.txt", row.names = F)


## Vector of removable strings

# Get names recognized by spaCy "en_core_web_sm" and manually filtered

names <- read.delim("names.txt", header = F)$V1 %>%

  unique() %>%

  tolower()


# temporal variables and names out + and any redundants out

rm_tokens <- c(c("coronavirus","outbreak", "vaccination", "virus", "pandemic","covid", #Covid
```

"invasion", "war", #Russia Ukraine

"pdf","emailprotected", #technical metadata

"jerry", "jordan", "mcteer","lorie","logan","mcdonough", "christopher", #names

"goolsbee", "austan", "kugler","adriana", "minehan", "waller",# more names

"cathy", "santomero", "anthony", "guynn", "jack", "jeffrey", # more more names

"stein","jeremy","kocherlakota", "narayana", "frederic", "robert", #even more names

"mishkin", "poole", "barr","jefferson","philip", "greenspan",# names

"raskin", "bloom", "sarah", "gary", "dennis", "ferguson",# some more names

"raphael", "barkin", "bostic", "neel", "kashkari", "roger", # names

"lockhart", "kroszner", "randall", "edward", "hoenig", "kohn", # look more names

"plosser", "stanley", "richmond", "fischer", "gramlich", "donald", # Names

"alan", "george", "esther", "patrick", "harker", "randal", "lawrence", # NAMES

"eric", "rosengren", "bullard", "james", "duke", "elizabeth", # yes names

"michael", "geithner","timothy", "warsh", "kevin", "mester", "clark", #...

"loretta", "pianalto", "sandra","susan","thomas", "bowman", "olivei", #

"michelle", "evans", "richard", "daniel", "tarullo", "brainard", # names

"lael","williams", "john", "yellen", "janet", "dudley", "charles", # nameSss

"charles", "bernanke", "ben", "powell", "jerome", "william","david", #nam

"jamie", "beth","alfred", "broaddus", "clarida", "mary", "collins", # Names

"rebecca","gagnon", "schweitzer", "roberts", "wohl", "schulhofer", "samuel", # NAMES!

"paulson","julie", "stewart", "engstrom", "laura", "jeff", "cook", # NAMES names

"zobel", "giovanni","patricia","harvey","nellie","krane","covitz", "lisa", # name

"figura", "wright", "ellis", "weber", "kartik", "paul", "jane", "simpson",# names

"anna", "shaghil", "ahmed", "zickler", "bernard", "kelley", "liang", # NAMES

"leahy", "vincent", "rochelle", "jonathan", "jon", "christine","stephen", # names

"weide", "kiley", # names

```
        "january", "february", "march", "april", "may", "june", "july", "august", #months
        "september", "october", "november", "december", # months
        "j", "n", "e", "c", "l", "w", "g", "f", "r","p", # single Letters
        "_____" # common string
), names)
```

Appendix 5: Announcement on the Use of Generative AI

In writing this thesis multiple different Generative AI models were used. Generative AI was used to assist with both the research for and writing of this thesis, as well as to review and generate draft project code. The purpose of use and my actions to verify AI generated content are described in this Announcement, with models listed by my frequency of their use. At the same time, I confirm that I have used AI tools with appropriate care, reported their use in accordance with guidelines, and assume full responsibility and ownership for the contents of Thesis and the appendices included.

1. **LLM used:** Scopus AI
    - **Stage of use:** Finding relevant literature.
    - **Purpose of use:** Scanning the Scopus database for relevant academic source materials.
    - **Verifying generation:** As this AI model produces retrieval augmented generation on the abstracts of articles it finds in the database. Thus, there is a high likelihood of getting irrelevant sources. Thus, I ended up reading the conclusions and methodology of most articles presented by this model to verify their suitability for further reading. For example, when looking for the original papers on FinBERT, of which there are many different versions, the most widely sited paper by Araci (2019) was not among the literature listed by Scopus AI. This paper was found via reviewing other finance NLP research, such as Kim et al. (2024, 627, 629) who state that Araci (2019) was the first to propose and develop such a model.

2. **LLM used:** Perplexity
    - **Stage of use:** Finding relevant source materials.
    - **Purpose of use:** Scanning the internet for relevant materials.
    - **Verifying generation:** When introducing myself to topics I often used Perplexity to point myself self-pedagogically towards relevant simple to digest material, before reading scientific literature. In essence, this was an alternative to using search engines such as google search. Like with google some sources that were pointed out were less

reliable than others. Thus, before reporting anything based on internet sources, academic validity was assessed by reviewing scientific research. For example, figures 8 and 7 were found in Colah's Blog which I found originally via Perplexity. Before using these figures myself, I looked at peer reviewed research and found Colah's materials referenced by LeCun et al.'s (2015) paper published in Nature.

3. **LLM used:** Google Gemini

- **Stage of use:** Finding relevant source materials and reviewing thesis text.
- **Purpose of use:** Scanning the internet for relevant materials and reviewing thesis text.
- **Verifying generation:** For finding relevant materials online, similar issues and ways of validating the LLM inference were applied as with Perplexity. For reviewing grammatical issues and especially when prompted for alternative phrasing, Gemini's outputs sometimes tended towards an overly confident direction. For example, when reporting findings, Gemini tended to use quite confident phrasing with words such as "robust" or "clear indications" when one or more statistical test results were used to discuss empirical findings of my thesis. Yet, given the limited sample size, I usually used milder adjectives and hedges my phrasing more than in the suggested phrases.

4. **LLM used:** Chat-GPT

- **Stage of use:** Drafting project code and reviewing thesis text.
- **Purpose of use:** Creating specific code snippets and reviewing thesis text.
- **Verifying generation:** Chat-GPT was used throughout this thesis project to create and edit Python and R-code scripts. All code produced by Chat-GPT was validated by the author and all new unfamiliar approaches were left out of the final project code. I needed to understand all code that was implemented. Sometimes Chat-GPT tended to attempt to rewrite ANN structures in an unfamiliar style when using the Keras package. In such cases I prompted Chat-GPT for an explanation of the code before rewriting it myself in a more familiar way that I understood. Also, when making complex R-Plots with the, for example the corrplot package, the suggested code

occasionally printed an empty plot or a plot with labels wrongly order. In such cases I ended up manually testing and fixing the LLM generated code until the wanted outcome was achieved. Chat-GPT served to draft an initial idea of the final code.

Authors Comments on the Use of Generative AI and ML Models within this Thesis:

While no sentence of this thesis was written by AI, models such as Chat-GPT 4 and 5, and Gemini 3 were used extensively to assist in clarifying statements and reviewing grammatical errors. Scopus AI, Perplexity, and Gemini 3 were used to assist in finding suitable academic literature.

Additionally, project code, particularly as it relates to the figures presented in this thesis, Chat-GPT was used to review code and generate preliminary snippets which were reviewed, tested, and edited by the author before implementation. Finally, DeepL was used to assist in translating the abstract from English to Finnish.

From the perspective of replicability, the BERT models used in this thesis are available at Hugging Face repositories uploaded by their original authors. Also, the configurations of other ML models used in the empirical analyses are described within chapter 4. The seed numbers used for both STMs and Keras trained ANNs were set to 2048 whenever possible.