Deree
School of Graduate
and Professional
Education

THE AMERICAN COLLEGE OF GREECE
Pierce
Deree
Alba
1875

A Novel Framework for Maritime Freight Rate Prediction Using Textual Data and Advanced AI Techniques

By

Konstantinos Tzoras

A thesis submitted in partial fulfillment of the
requirements for the degree of
MASTER OF SCIENCE
In
Data Science

Supervisor: Dr. Chatzimichali Elena

DEREE – The American College of Greece

July 2025

Abstract

The maritime shipping industry remains a backbone of global trade, responsible for transporting the majority of goods worldwide. However, the sector faces persistent challenges due to high volatility in freight rates driven by complex and often unpredictable interactions between economic conditions, geopolitical events, and market sentiment. Traditional forecasting models struggle to fully capture these dynamics, limiting their ability to provide accurate and timely predictions that are crucial for stakeholders navigating an increasingly uncertain environment.

This study introduces a novel and innovative approach that harnesses Large Language Models (LLMs) to extract meaningful features from maritime news articles, capturing qualitative signals such as sentiment and event characteristics relevant to freight rate fluctuations. By integrating these text-derived features with advanced machine learning and econometric models, the methodology enriches time-series forecasting beyond conventional quantitative data. A striking finding of this work is the clear causal relationship identified between news-derived features and freight rate movements on the key C5 route, validating the approach's ability to capture actionable market signals. Finaly, this work proposes a scalable, cost-effective framework for integrating textual data into maritime freight rate prediction, addressing existing gaps and supporting more informed decision-making in a volatile market environment.

Acknowledgments

I would like to sincerely thank my supervisor, Dr. Chatzimichali, for her continuous support and guidance throughout this thesis. I am also grateful to the library of ALBA for providing me with access to Clarkson's Network of Intelligence, which was invaluable for obtaining essential data. Finally, I deeply appreciate the unwavering support and encouragement from my family, which has been a constant source of strength.

Table of Contents

Lists of Tables

List of Figures

List of Abbreviations

Abbreviation	**Definition**
LLM	Large Language Model
NLP	Natural Language Processing
RAG	Retrieval-Augmented Generation
IDFE	Informed Decision Feature Extraction
IDRBFE	Informed Decision Reason-Based Feature Extraction
FE	Feature Extraction
RMSE	Root Mean Squared Error
R^2	Coefficient of Determination
PSO	Particle Swarm Optimization
XAI	Explainable Artificial Intelligence
VADER	Valence Aware Dictionary and Sentiment Reasoner
BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pretrained Transformer
RoBERTa	Robustly Optimized BERT Pretraining Approach
SVR	Support Vector Regression
LSTM	Long Short-Term Memory
TFPSO-DLSTM	Twofold Partial Swarm Optimization-Based Stacked Long Short-Term Memory

GARCH	Generalized Autoregressive Conditional Heteroskedasticity
VIF	Variance Inflation Factor
CCM	Convergent Cross Mapping
EDA	Exploratory Data Analysis

1. Introduction

By 2019 the shipping industry was attributed to transporting over 80% of goods traded (UNCTAD, 2019) and the total maritime traded volume shows an increase of growth since the disruption caused by the COVID-19 epidemic (UNCTAD, 2024). As a result of global trade, the world economy is heavily dependent on and significantly influenced by maritime shipping. Moreover, The Review of Maritime Transport 2024 by UNCTAD (2024) reports on external factors that disturb the correlation between maritime trade and macroeconomic metrics, highlighting a complex interrelation. These factors are events whose effects differ in terms of locality, duration and scale.

Maritime transportation is a significant factor in the economic growth of developing countries (Park et al., 2019), while the cost of transportation is a major determinant for the rate of a country's growth (Radelet & Sachs, 1998). Improvements in the efficiency of the shipping industry show positive effects on socioeconomic conditions across many regions (Yudhistira & Sofiyandi, 2017). Enhancing the operational efficiency of the shipping industry can be achieved through several key strategies (Hoffmann et al., 2017). These include the adoption of advanced technologies such as digital tracking systems and automated port operations, which streamline logistics and reduce turnaround times. Furthermore, investing in infrastructure improvements, such as modernizing port facilities and enhancing navigational systems, can significantly lower shipping costs and improve reliability.

Additionally, creating and maintaining stronger partnerships between public entities and private shipping companies encourages better coordination and resource sharing, which further optimizes transportation networks. Training and capacity building for personnel in maritime operations also play a crucial role in ensuring that advancements in technology and infrastructure are effectively utilized.

Overall, these enhancements contribute not only to reduced costs in maritime transportation but also to increased trade volumes, ultimately driving economic growth in developing nations (Park et al., 2019). By focusing on these areas, the shipping industry can effectively support the broader economic and social development goals of these countries.

As nations become increasingly interconnected, shipping networks support the flow of raw materials, intermediate products, and finished goods across borders, thereby bolstering industrial production and stimulating economic growth (Xu et al., 2020).

The demands of global production significantly shape maritime trade patterns (Changing demand for maritime trade, 2020). For instance, the growing need for specific commodities such as rare earth metals for technology or agricultural products for food security, directly influences shipping routes and volumes. When production levels rise in a particular sector, shipping companies often adjust their capacities to accommodate increased demand, creating a dynamic relationship between production needs and maritime logistics. Conversely, developments in the shipping industry, such as the emergence of larger vessels or more efficient routing practices can lead to changes in macroeconomic indicators, including trade balances, employment rates and inflation (Ferrari et al., 2023). The increase in shipping efficiency can reduce costs, making products cheaper and more accessible, thereby promoting consumption, economic activity and growth.

Maritime trade is pivotal in improving the quality of life globally by facilitating the distribution of essential resources such as energy, pharmaceuticals and agricultural products. For example, the transportation of crude oil and liquefied natural gas through shipping channels is crucial in meeting energy demands, particularly in regions that lack natural resources (Chatham House, 2017). Likewise, transporting agricultural products from regions with surplus to those experiencing shortages helps stabilize food availability and prices, thereby enhancing food security for millions across the globe (Food and Agriculture Organization of the United Nations, n.d.).

This supports the argument that efficient maritime trade systems directly contribute to global improvements in quality of life. By lowering transportation costs and ensuring timely delivery of goods, the maritime ecosystem facilitates better access to essential items and services. However, several deterrents exist that can hinder investment in the shipping industry. These include regulatory challenges, environmental concerns, and most important the volatility of global trade markets (UNCTAD, 2024). Additionally, external factors, such as political instability or natural disasters, can disrupt shipping routes, leading to further economic uncertainty in the industry of maritime trade (Notteboom & Rodrigue, 2021). Volatility in the shipping industry, often driven by fluctuating demand and global economic trends, can deter potential investment, which hinders the overall ecosystem of global trade (Stopford, 2009).

In conclusion, the shipping industry is integral to the global economy, connecting the production of goods and the fulfillment of needs, significantly impacting macroeconomic indicators, and therefore the development of countries. Maritime trade plays a crucial role in enhancing the quality of life globally by ensuring the efficient flow of essential resources. As emerging challenges and opportunities arise, the industry's adaptation will be vital in continuing to support global trade and therefore the well-being of people around the world.

1.1 Project Objective

To improve the accuracy of freight rate forecasts and support more effective decision-making within the maritime sector, the following methodology was developed and validated through systematic testing:

1.1.1 Data Collection

- Maritime news articles were systematically scraped from a prominent industry source using a custom web scraping algorithm.
- Structured freight rate data was sourced and transformed into weekly time series using interpolation for missing values.

1.1.2 Feature Extraction via Natural Language Processing (NLP)

A Large Language Model (LLM) was employed to derive event features from the textual data of the news articles, such as sentiment and duration of these events, the shipping sub-sector and routes affected, and the scale implied by them.

A Large Language Model (LLM) is a type of advanced artificial intelligence model trained on vast amounts of textual data to understand, generate, and analyze human language. These models leverage deep learning architectures, particularly transformer networks, to capture complex patterns and contextual relationships within text, enabling them to perform various natural language processing tasks such as sentiment analysis, summarization, translation, and feature extraction (Vaswani et al., 2017; Brown et al., 2020).

A transformer is a deep learning architecture designed to process sequential data, such as text, by using a mechanism called self-attention. Unlike traditional recurrent neural networks (RNNs), which process data sequentially, transformers analyze the entire input sequence simultaneously, allowing them to capture long-range dependencies and contextual relationships more efficiently (Vaswani et al., 2017). The self-attention mechanism assigns different weights to different parts of the input, enabling the model to focus on the most relevant words or tokens when generating predictions. This architecture forms the backbone of Large Language Models (LLMs) and has revolutionized natural language processing tasks by improving both accuracy and computational efficiency. By employing an LLM, the study was able to automatically extract meaningful event-related features from unstructured maritime news articles, facilitating a more nuanced input for the forecasting models.

These features were aligned with freight rate data to enrich the forecasting dataset with the respective market sentiment. Due to the scale of the data, it would be more efficient to accumulate a number of data points from an LLM (article-labels) and train classification models, resulting in low-cost and faster results compared to using exclusively an LLM.

This allowed the forecasting models to capture qualitative signals from maritime news that are not employed in traditional forecasting methods.

1.1.3 Predictive Modeling and Optimization

A variety of supervised machine learning models were employed to address regression tasks, where the objective is to predict continuous numerical outcomes based on input features. The models applied include Linear Regression, Ridge Regression, Lasso Regression, ElasticNet, Support Vector Regression (SVR), Random Forest Regression, XGBoost, and LightGBM. These algorithms learn patterns from labeled datasets, enabling them to estimate target variables effectively. Linear models such as Ridge, Lasso, and ElasticNet incorporate regularization techniques to prevent overfitting, while ensemble methods like Random Forest, XGBoost, and LightGBM leverage multiple decision trees to improve predictive accuracy and robustness. Support Vector Regression applies the principles of support vector machines to regression problems by fitting a function within a specified margin of tolerance. The use of this diverse set of models provides a comprehensive evaluation of different algorithmic approaches to the regression problem (James et al., 2013; Géron, 2019; Chen & Guestrin, 2016; Ke et al., 2017).

Grid search optimization was employed to systematically identify the optimal combination of hyperparameters, specifically the window size and forecasting interval, that yield the best predictive performance. Window size refers to the number of previous time steps or data points used as input features for the model to learn temporal patterns, while the forecasting interval denotes the time horizon into the future for which predictions are generated. By exhaustively searching through a predefined set of possible values for these parameters, grid search evaluates model performance across all combinations using cross-validation or a validation dataset. This approach ensures a comprehensive exploration of the hyperparameter space, enabling the selection of values that minimize prediction error and improve the robustness and accuracy of the forecasting models.

The performance of the regression models was assessed using two commonly employed evaluation metrics: Root Mean Squared Error (RMSE) and the coefficient of determination (R^2).

RMSE measures the average magnitude of the prediction errors, providing insight into how close the predicted values are to the actual values. It is calculated as:

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

where n is the number of observations, y_i is the actual value, and $\hat{y}_i$ is the predicted value. RMSE is expressed in the same units as the target variable, and lower values indicate better model performance.

R^2 represents the proportion of variance in the dependent variable that is predictable from the independent variables. It is defined as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \mu)^2}.$$

where μ is the mean of the observed data. The value of R^2 ranges between 0 and 1, with values closer to 1 indicating a better fit of the model to the data.

Together, RMSE and R^2 provide complementary perspectives on model accuracy: RMSE quantifies the average prediction error magnitude, while R^2 assesses the explanatory power of the model.

1.1.4 Model Validation and Causality Testing

Robustness of the predictive power was confirmed using advanced statistical techniques, including Toda-Yamamoto Granger causality, Transfer Entropy, and Convergent Cross Mapping (CCM).

- Toda-Yamamoto Granger causality is an extension of the traditional Granger causality test that allows for testing causal relationships in the presence of integrated or non-stationary time series without requiring pre-testing for cointegration (Toda & Yamamoto, 1995). It is used to assess whether one time series can predict another.
- Transfer Entropy is an information-theoretic measure that quantifies the directional transfer of information between two time series, capturing nonlinear and dynamic dependencies beyond linear correlations (Schreiber, 2000). It helps detect causality where traditional methods may fail.
- Convergent Cross Mapping (CCM) is a nonlinear causality detection method based on state-space reconstruction, designed to identify causal links even in complex, nonlinear dynamical systems

(Sugihara et al., 2012). CCM tests whether the historical states of one variable can reliably estimate the states of another, indicating causation.

These complementary methods provide robust evidence of correlation and causality between the derived features and freight rate movements, enhancing the credibility of the predictive models.

1.1.5 Visualization and Result Communication

Visualization tools such as heatmaps and performance plots were used to communicate model results.

By adopting this methodology stakeholders in the maritime industry can improve their forecasting capabilities. This framework enables a new approach to navigating volatility in shipping markets, which contributes to greater efficiency in decision making.

This study is rooted in the hypothesis that the effects of world events are not immediately reflected in freight rates but rather have short-term as well as long-term effects. In addition to the above, events can also have localized effects, commodity and vessel specific effects, but also affect the freight markets in many other ways.

It is important to note that there are direct effects from the events, such as port congestion due to workers strike, and secondary effects from market agents reacting to the events as they are reported in the news, i.e. market sentiment, for instance traders avoiding a specific port in the future due to frequent reports of strikes.

Following the advancements of LLMs, this study would help ascertain whether an LLM could extract features pertaining to the aforementioned effects, given adequate instructions and data.

The two hypotheses will be tested by incorporating the extracted features in forecasting models and comparing results with benchmark models.

2. Literature Review

2.1 Economic Cycles and Volatility in Maritime Markets

Economists have investigated the shipping industry and have identified three components of cyclicality in the respective market (Stopford, 2009). Similar to many industries, shipping displays:

- Long cycles/Secular trends

Long-term cycles that span up to 60 years peak-to-peak. Usually fueled by the emergence of new technologies (paddle-ship, specialized vessels, etc.) (Stopford, 2009; UNCTAD, 2018). Prime examples of that is the transition from sail to steamships in the 19th century, the introduction of containerization in the 1960s, and the rise of LNG carriers and eco-ships in the 21st century (Stopford, 2009; UNCTAD Review of Maritime Transport, 2018; Alderton, 2008).

- Short cycles

Average length of 5 years. Caused by market changes (and/or due to world events, wars, major accidents, etc.) and the over-corrections in the supply of shipping capacity (Stopford, 2009; Clarkson Research, 2021). Such cycles were caused by the shipping boom and bust of 2003–2008 driven by over-ordering of vessels; the post-COVID demand surge followed by capacity misalignment (Stopford, 2009; Clarkson Research Services, 2021; OECD-ITF Maritime Outlook, 2020).

- Seasonal cycles

Occur regularly each year. Seasonal patterns of demand for shipping services (harvest seasons, heating requirements, etc.) (UNCTAD, 2020; Drewry, 2019). Higher demand for grain shipping during Northern Hemisphere harvest seasons (Q3-Q4) and increased coal and LNG demand in winter months for heating in East Asia, are regular causes of seasonal cycles in the respective shipping sectors (UNCTAD, 2020; Drewry Maritime Research, 2019; Baltic Exchange Reports).

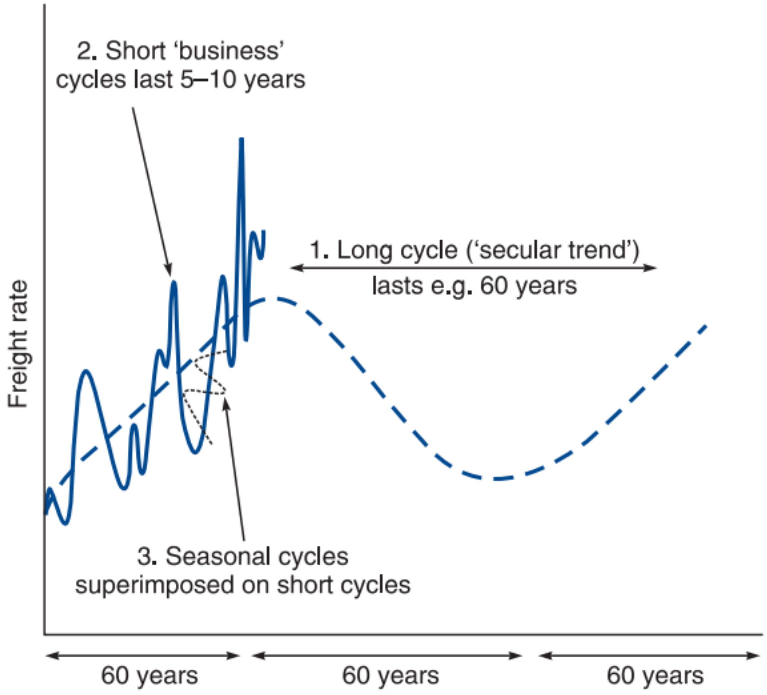

2. Short 'business' cycles last 5–10 years
1. Long cycle ('secular trend') lasts e.g. 60 years
Freight rate
3. Seasonal cycles superimposed on short cycles
60 years
60 years
60 years

Figure 1. Freight rate cyclicality illustrating long cycles (secular trends), short business cycles, and seasonal cycles (adapted from Stopford, 2009, 3rd ed., p. 95).

Amidst these cyclical phenomena, shipping rates display high volatility on a daily basis (Baltic Exchange, 2022). These characteristics are common across all vessel sizes and types, as well as most trading routes (Stopford, 2009; Clarkson Research, 2021; UNCTAD, 2020).

2.2 Characteristics of a Perfectly Competitive Market

The global shipping industry exhibits many hallmarks of a perfectly competitive market structure, including a large number of buyers and sellers, standardized (homogeneous) services, and minimal barriers to entry and exit. Shipping services, particularly in the dry bulk and tanker sectors, are largely interchangeable, with vessels of a given type (e.g., Panamax, Suezmax) performing identical transport functions. Market participants, both charterers and shipowners, generally have access to timely information on freight rates and market conditions, supported by platforms such as the Baltic Exchange and advanced vessel tracking systems, such as the AIS (Stopford, 2009; UNCTAD, 2020).

As a result of these features, individual operators in the shipping market act as price takers rather than price makers. Freight rates are determined by the intersection of aggregate supply and demand in the market. This pricing mechanism leads to a high degree of rate volatility, as shifts in global trade flows, fleet availability, and port logistics can cause immediate and significant price fluctuations (Clarkson Research, 2021).

2.3 Determinants of Supply in the Shipping Industry

The supply of shipping services is not static and is influenced by a variety of operational, infrastructural, and economic factors. The following are the primary determinants of shipping supply:

2.3.1 Construction of New Vessels

The construction of new vessels directly adds to the global fleet capacity and is one of the most significant long-term drivers of supply. The lead time for newbuild deliveries typically ranges from 18 to 36 months, making supply expansion relatively inelastic in the short run. New orders are usually influenced by market optimism, availability of shipyard slots, and evolving technological and environmental requirements, such as those mandated by the International Maritime Organization (IMO) (Stopford, 2009; BIMCO, 2022).

For instance, during the 2003–2008 super-cycle, a surge in demand for commodities led to a wave of newbuild orders. However, this oversupply became evident after the global financial crisis, when excess tonnage contributed to prolonged rate depression across segments. More recently, a spike in orders for LNG carriers and dual-fuel ships in 2021–2023 reflects the growing regulatory emphasis on decarbonization (Clarkson Research, 2022).

2.3.2 Sailing Speeds

Sailing speeds, often adjusted through a practice known as slow steaming, significantly affect the effective supply of vessels. A decrease in sailing speed lowers fuel consumption and operating costs but also reduces the number of voyages a vessel can make within a given time frame, thereby tightening available supply. Conversely, higher freight rates incentivize faster transits and increase effective supply (UNCTAD, 2020).

The use of slow steaming became widespread after 2008 as operators sought to cut costs amid falling freight rates and rising fuel prices. Moreover, new environmental standards such as the IMO's Energy Efficiency Existing Ship Index (EEXI) and Carbon Intensity Indicator (CII) are expected to lead to further speed reductions, constraining supply even when the available fleet remains unchanged (IMO, 2023; Maersk, 2022).

2.3.3 Port Efficiency

Port infrastructure and operational efficiency also play a critical role in determining the availability of shipping capacity. Delays at ports, whether due to congestion, labor disputes, or inadequate equipment, reduce fleet productivity by extending vessel turnaround times. Conversely, investments in automation and logistics technology can improve throughput and effectively increase available supply without changing fleet size (World Bank, 2021; Drewry, 2019).

For example, the congestion experienced at major U.S. West Coast ports during the COVID-19 pandemic in 2020–2021 resulted in significant supply-side constraints, as ships were held at anchor for extended periods. On the other hand, ports such as Singapore and Rotterdam have maintained high productivity levels due to advanced infrastructure and digitalized operations (UNCTAD, 2020).

2.3.4 Demolition of Vessels

The scrapping or demolition of vessels is another key supply-side adjustment mechanism, particularly during prolonged market downturns.

Older ships, which may be less fuel-efficient or non-compliant with new regulatory standards, are often sold for demolition when operating costs exceed projected earnings. Scrapping activity also correlates with steel prices, demolition yard capacity, and regulatory pressures (Stopford, 2009; BIMCO, 2023).

For instance, after 2016, a large number of older Capesize and Panamax bulk carriers were scrapped in response to low charter rates. Furthermore, the industry anticipates heightened scrapping activity in 2024–2026 as a result of environmental mandates such as CII thresholds, which may render older tonnage commercially inefficient (Clarkson Research, 2022).

2.4 Determinants of Demand in the Shipping Industry

The demand for shipping services depends on how much cargo needs to be moved across the world. This demand can change based on what goods are being traded and how the global economy is performing. Two of the main factors influencing demand are the dependence of commodities and macroeconomic factors.

2.4.1 Trade and Commodity Dependency

Shipping demand is strongly linked to international trade, especially in bulk commodities like coal, oil, iron ore, grain, and manufactured goods. When global trade increases, more cargo needs to be transported, which raises demand for ships. On the other hand, when trade slows down, fewer goods are moved, and demand for shipping services falls.

Many ships are designed for specific types of cargo. For example, dry bulk ships mostly carry commodities like coal, iron ore, and grain, while tankers carry oil and gas. So, demand for these ships depends directly on how much of these goods are being bought and sold around the world.

The rise of China's steel industry in the early 2000s led to a huge increase in demand for dry bulk shipping, especially for iron ore from Australia and Brazil (Stopford, 2009). When the COVID-19 pandemic disrupted trade in 2020, global shipping demand dropped sharply, especially in container shipping (UNCTAD, 2020). Grain exports from the U.S., Russia, and Ukraine create seasonal spikes in shipping demand each year (Drewry, 2019).

2.4.2 Macroeconomic Factors and Global Events

Big-picture economic trends also affect shipping demand. When the global economy grows, industries produce and consume more goods, which boosts trade and shipping. But during economic recessions, wars, or financial crises, global trade often slows down, which lowers the need for shipping.

Other events, such as natural disasters, pandemics, or geopolitical tensions, can also affect trade routes and cargo flows. These events may either increase demand (e.g., for emergency supplies or fuel) or reduce it (e.g., by slowing production or closing ports).

The 2008 global financial crisis caused a major drop in demand for all types of shipping, especially container ships. The COVID-19 pandemic disrupted supply chains, which first reduced and then sharply increased demand for container shipping in 2021 (UNCTAD, 2020; Clarkson Research, 2021). Russia's invasion of Ukraine in 2022 disrupted grain and oil exports from the Black Sea region, which affected demand and shifted trade routes (OECD, 2023).

In general terms, due to the nature of the demand for shipping services, freight rates are very volatile. On the other hand, supply of shipping capacity is slow to change, lags behind demand, and usually tends to over-correct; This mechanism creates the effects of short cycles as previously mentioned.

Further to the overall global market conditions, there are periods when supply and demand change very abruptly at a local level. In certain geographical clusters, specific market factors, such as port disruptions, weather events, or regional political instability, can shift suddenly and independently from the global trend. These local changes often cause the freight rate indices for specific shipping routes to diverge significantly from the broader market indices.

For example, a strike at a major port or a regional conflict can immediately reduce port capacity or limit cargo flows, causing freight rates in that area to spike, even if the global market remains stable (UNCTAD, 2020). Similarly, natural disasters such as hurricanes or floods may temporarily block access to key ports, reducing the availability of ships in the region and increasing short-term demand for nearby vessels (OECD, 2023).

These effects are typically a response to local or global events that disrupt normal trade patterns, and they result in sudden and localized volatility in the freight market. This can be seen in sharp divergences between route-specific indices such as the Baltic Dry Index (BDI) subcomponents and the broader cumulative indices (Clarkson Research, 2021; Baltic Exchange, 2022).

This study investigates the hypothesis that events reported in news articles impact freight rates not immediately, but through short-term and long-term effects. Additionally, events may induce localized impacts, specific effects related to commodities and vessels, and broader influences on freight markets. Notably, these effects encompass direct consequences, such as port congestion from workers' strikes, and secondary effects driven by market sentiment, like traders avoiding ports due to frequent strike reports. Leveraging advancements in Large Language Models (LLMs), the study explores whether an LLM can effectively extract features related to these effects given appropriate instructions and data. The two hypotheses will be tested by integrating the extracted features into forecasting models and evaluating performance against benchmark models.

2.5 Feature Extraction from News Article

This research utilizes Large Language Model (LLM) Prompt Engineering, emphasizing the importance of specificity and simplicity when constructing prompts. Effective prompts should include detailed instructions and examples, guiding LLMs to accurately assign numerical or categorical values to quantify the effects of events described in news articles on determinants of supply and demand (Liu et al., 2023; Brown et al., 2020).

Evaluating the accuracy of LLM-generated results initially requires manually labeling a subset of articles. Manual labeling is performed based on both objective criteria—such as explicit facts reported in the articles—and subjective historical data, interpreting historical implications of similar past events (Eisenstein, 2019).

However, examining historical data presents challenges, primarily due to the complexity involved in disentangling cumulative effects of numerous concurrent global events influencing freight rates. Isolating and quantifying the precise impact of individual articles or events becomes particularly challenging (Angrist & Pischke, 2009). One potential solution is to create a feedback loop through regression models, allowing continuous refinement and updating of the estimations based on model performance. Such an approach could enhance the identification of individual event impacts, particularly concerning market sentiment and secondary effects driven by news narratives (Shiller, 2017).

Lexical dictionary creation and sentiment quantification involve developing specialized dictionaries or lexicons containing words and phrases that have predefined sentiment values or semantic associations, used to measure sentiment in text data (Bai, Lam, & Jakher, 2021). Specifically, this approach includes identifying and categorizing key terms related to particular themes or sentiments, assigning each term a numerical or categorical sentiment score, i.e. positive, neutral, or negative, and subsequently applying these lexicons to quantify sentiment or emotional tone in textual content.

In the context described by Bai, Lam, and Jakher (2021), the process of lexical dictionary creation and sentiment quantification involves two primary steps:

1. Lexical Dictionary Creation

The first step focuses on extracting and compiling a set of relevant words or terms from textual sources such as news articles, social media content, or other text-based data. These terms are selected based on specific criteria, including their frequency of occurrence, semantic significance, or thematic relevance to the area under investigation. Once extracted, these terms are systematically organized into structured lexical dictionaries or lexicons. The assignment of sentiment scores or categorical labels (positive, neutral, negative) to each term typically relies upon expert judgment, statistical methodologies, or context-specific guidelines. The resulting lexical dictionaries thus become structured reference tools designed to support subsequent analytical tasks.

2. Sentiment Quantification

In the second stage, the lexical dictionary developed previously is applied systematically to textual data to quantify sentiment. This process involves automatically assigning and aggregating the pre-defined sentiment scores for each word or phrase present within the analyzed texts. By aggregating these scores, it is possible to produce comprehensive sentiment measurements for individual texts or collections of texts. Subsequently, these quantified sentiment scores can be summarized and interpreted to identify general trends, detect changes or shifts in sentiment over specified periods, and potentially link these sentiment dynamics to tangible real-world outcomes. Examples of such outcomes include market reactions, fluctuations in commodity prices, or alterations in demand and supply conditions, thereby establishing connections between textual sentiment and economic determinants or behaviors.

This methodological approach, as outlined by Bai et al. (2021), supports the structured and reproducible quantification of sentiment from unstructured textual data, facilitating deeper analysis of how narratives, sentiment, and news events may influence economic determinants or market behaviors.

2.6 Economic Drivers and Forecasting Techniques

Given the pivotal role of the shipping industry, comprehending and accurately forecasting freight rates is crucial, as variations in these rates have substantial impacts on global economic stability, international trade logistics, and strategies for resource allocation (UNCTAD, 2023). Freight rates exhibit cyclical behaviors, driven by long-term trends, short-term market corrections, and seasonal patterns. The volatility of freight rates arises mostly from differences between sudden fluctuations in shipping demand and the slow adjustments and over-correction of vessel supply (Stopford, 2009).

Economic factors significantly influence shipping markets. Global economic growth, commodity demand variations, fuel price dynamics, geopolitical developments, and international trade policies profoundly impact freight rates (Stopford, 2009; Hoffman et al., 2017). For instance, Hoffman et al. (2017) demonstrated that macroeconomic indicators such as GDP growth and industrial production are strongly correlated with maritime trade volumes and shipping costs, reinforcing the importance of these variables in forecasting models.

Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models are statistical tools used to analyze and forecast time series data with changing volatility over time, a phenomenon known as volatility clustering. Volatility clustering refers to the tendency of high-volatility events to be followed by more high-volatility events, and low-volatility periods to cluster together (Bollerslev, 1986). Unlike traditional models that assume constant variance, GARCH models explicitly model the variance as a function of past errors and past variances, allowing for dynamic changes in volatility. This makes GARCH particularly useful for financial and economic data where volatility is not stable, such as freight rates during disruptive events like the COVID-19 pandemic.

Michail and Melas (2020) used GARCH models to analyze volatility clustering during the pandemic, highlighting shipping market sensitivity to global disruptions. Their findings illustrate how sudden shifts in global demand and supply chains directly influence freight rates, thereby validating the need for robust predictive methodologies capable of accommodating such sudden changes in the maritime environment.

The Twofold Partial Swarm Optimization-Based Stacked Long Short-Term Memory (TFPSO-DLSTM) model is a hybrid machine learning framework designed to enhance time series forecasting accuracy by combining optimization algorithms with deep learning. The core of the model consists of Stacked Long Short-Term Memory (LSTM) networks, a type of recurrent neural network specialized in capturing long-term dependencies and sequential patterns in data (Hochreiter & Schmidhuber, 1997). To optimize the LSTM network's hyperparameters—such as learning rates, number of layers, and neuron counts—the model employs Twofold Partial Swarm Optimization (TFPSO), an advanced variant of particle swarm optimization (PSO). PSO is a population-based stochastic optimization technique inspired by social behavior in flocks of birds or schools of fish, which iteratively searches for optimal solutions by updating candidate solutions based on individual and group experiences (Kennedy & Eberhart, 1995). The twofold partial aspect refers to a refined mechanism that improves convergence speed and avoids premature stagnation during optimization. By integrating TFPSO with stacked LSTM, the model effectively balances exploration and exploitation, resulting in improved forecasting performance, especially in complex and volatile environments such as maritime freight rate prediction (Xiao, 2024).

The role of psychological factors, particularly investor sentiment, in influencing shipping market dynamics has recently received academic attention. Xiao (2024) implemented a Twofold Partial Swarm Optimization-Based Stacked Long Short-Term Memory (TFPSO-DLSTM) model, incorporating emotional sentiment derived from investor reactions into freight rate predictions. The study found that integrating emotional indices substantially enhanced forecasting accuracy, particularly under volatile market conditions.

Furthermore, Wu et al. (2020) investigated the asymmetric impacts of shipping fear indices, constructed from investor sentiment analysis, on the dry bulk shipping market. They found that negative sentiment or increased fear levels significantly raised market volatility, thereby providing valuable predictive signals. These findings suggest that sentiment indices serve as beneficial indicators, capable of capturing investor behaviors, thus adding a valuable dimension to freight rate forecasting models.

Traditional econometric and statistical methods have historically supported freight rate forecasting, offering insights into market behaviors. These models typically include univariate and multivariate time series approaches, regression analyses, and volatility models such as GARCH. Xu et al. (2021) applied panel regression models to examine the impact of the COVID-19 pandemic on global port operations, emphasizing the interconnected nature of shipping logistics and economic disruptions. Their research underlined the need for adaptable models capable of accounting for global shocks and their cascading effects across shipping networks and thereby the global economy.

Koyuncu et al. (2021) utilized Seasonal Autoregressive Integrated Moving Average (SARIMA) models to predict freight rates, demonstrating their effectiveness in capturing seasonal variations and short-term cyclical fluctuations inherent to shipping markets. Similarly, Jeon et al. (2020) and Yang et al. (2008) explored the application of Support Vector Machines (SVMs) for freight rate forecasting. Their research highlighted SVMs' capability in handling nonlinear and high-dimensional data, enhancing predictive accuracy compared to traditional linear models.

Despite the effectiveness of traditional models, their limitations, particularly in handling the complex and nonlinear relationships that characterize shipping data, pushed the exploration of advanced machine learning and deep learning techniques. Long Short-Term Memory (LSTM) networks have emerged prominently due to their superior ability to model temporal dependencies and complex sequential patterns. Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to effectively capture long-range dependencies and temporal patterns in sequential data. Unlike traditional RNNs, LSTMs use special memory cells and gating mechanisms to regulate information flow, which helps overcome issues like the vanishing gradient problem during training. This makes LSTMs particularly well-suited for time series forecasting and natural language processing tasks (Hochreiter & Schmidhuber, 1997). Wu et al. (2024) demonstrated that LSTM networks significantly outperform traditional models in forecasting shipping market trends, particularly when handling intricate temporal dynamics inherent to freight rate data.

To further enhance forecasting precision, hybrid models combining LSTM with optimization algorithms have been developed. Du et al. (2022) and Tian and Shi (2018) introduced hybrid LSTM models integrated with Particle Swarm Optimization (PSO) algorithms, which optimize hyperparameters and address premature convergence issues common in traditional neural networks. These hybrid approaches demonstrated greater predictive performance, highlighting their utility in real-world shipping market scenarios.

2.7 Advances in Textual Data Analysis for Freight Rate Forecasting

An emerging and under-explored area in shipping freight rate forecasting involves leveraging unstructured data sources such as maritime news articles. Natural Language Processing (NLP) methodologies have recently provided powerful tools for extracting sentiment and event-specific indicators from textual sources, improving traditional forecasting methods' predictive capabilities.

Sentiment analysis quantifies the emotional context of text, significantly improving predictive accuracy in financial and logistics forecasting. Widely used sentiment analysis tools include NLTK VADER, TextBlob, SentiWordNet, and AFINN, which offer distinct capabilities.

NLTK VADER (Valence Aware Dictionary and Sentiment Reasoner) is a rule-based model specifically designed to analyze sentiment in short, informal texts like social media posts and news snippets. It uses a lexicon of words rated for sentiment intensity, combined with rules that consider context, punctuation, and capitalization to capture nuanced emotional expressions effectively (Hutto & Gilbert, 2014).

TextBlob is a Python library built on top of NLTK that provides simple APIs for common natural language processing tasks, including sentiment analysis. It uses a lexicon-based approach and part-of-speech tagging to classify text polarity and subjectivity, making it user-friendly and suitable for general sentiment classification (Loria, 2018).

SentiWordNet is a lexical resource that assigns sentiment scores to WordNet synsets (sets of synonyms). It provides fine-grained sentiment annotations for words, enabling more precise sentiment classification based on the meanings of words in context (Baccianella, Esuli, & Sebastiani, 2010).

AFINN is a lexicon-based sentiment analysis tool where words are scored on a scale from negative to positive integers. It is designed for simplicity and speed, making it efficient for analyzing large volumes of text where quick sentiment scoring is needed (Nielsen, 2011).

Recent studies have underscored NLP-driven methodologies' potential in shipping market predictions. Bai et al. (2021) demonstrated the significant impact of emotional sentiment on shipping investment decisions, illustrating that sentiment data effectively captures investor behaviors which influence the state of markets. Jeon et al. (2020) successfully utilized NLP-driven system dynamics models to forecast shifts in major shipping indices such as the China Coastal Bulk Freight Index and the Baltic Freight Index, reinforcing the efficacy of sentiment-based forecasting approaches.

Advanced transformer-based NLP models, including Bidirectional Encoder Representations from Transformers (BERT), Generative Pretrained Transformer (GPT), and Robustly Optimized BERT Pretraining Approach (RoBERTa), have revolutionized sentiment analysis and feature extraction tasks (sources)*. These models excel at capturing contextual meanings within data in the form of text, enabling the extraction of accurate sentiment and event-specific predictive features. This research integrates these advanced language models, systematically evaluating their predictive capabilities in freight rate forecasting, significantly addressing existing literature gaps.

In conclusion, by explicitly integrating LLM-derived sentiment and event features into freight rate prediction models, this thesis greatly contributes to existing research. It demonstrates the considerable potential of combining advanced NLP techniques with econometric and machine learning forecasting methods, enhancing prediction accuracy, especially in the volatile maritime shipping sector.

3. Research Design & Methodology

3.1 Overview

A large number of articles and relevant meta-data were scraped off a well-established publisher. With the use of Large Language Models and prompt engineering, features relevant to shipping rates were extracted. For feature extraction, four distinct approaches were implemented. Features were extracted for ten percent of the total number of articles. Classification models were trained on the extracted features and were used to make predictions for the remaining. Then the features were implemented in regression models and the resulting accuracy was compared to benchmark results. Several approaches were used to create the final features used in the regression models. For example, one approach was to filter features for the C5 route and dry-bulk Capesize vessels, and calculate the average impact for each day.

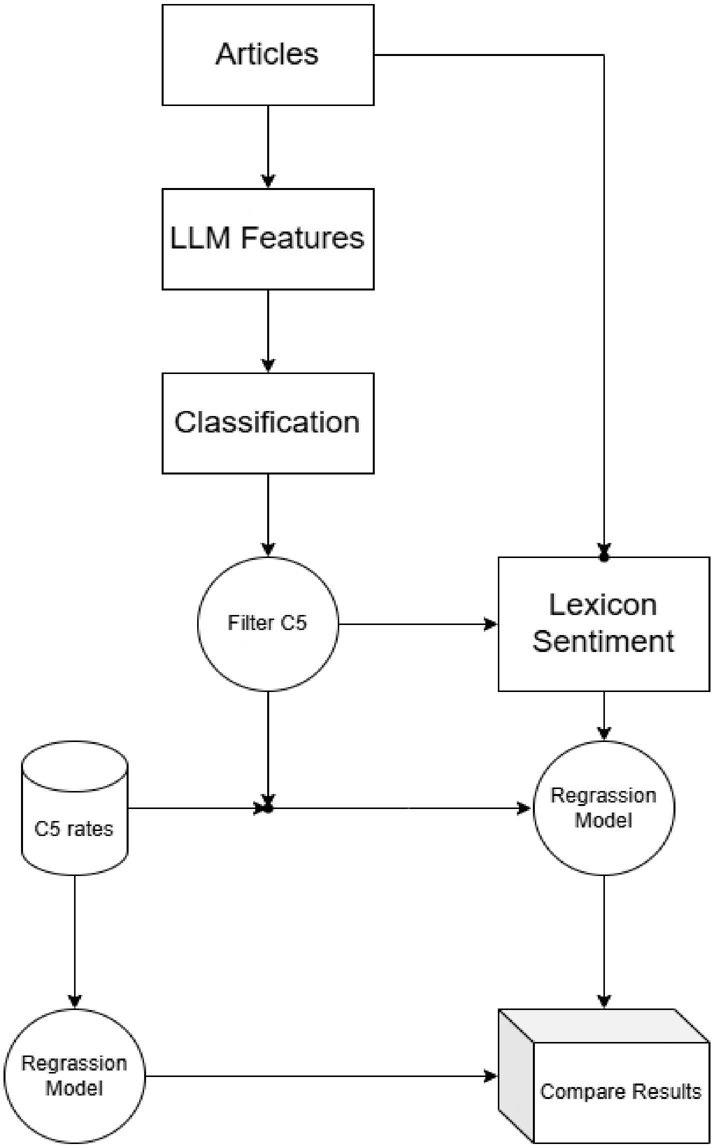


Figure 2 – Overview of the methodology from sentiment extraction to freight rate forecasting

3.1.1 Hardware and Software Specifications

All experiments were conducted on a local machine. The computer had a Windows 11 Pro 64-bit operating system. The python version was 3.10.11, to accommodate versions of libraries used. The hardware specifications of the local machine are as follows:

Hardware	Description
CPU	Intel(R) Core(TM) Ultra 7 155H 3.80 GHz
RAM	DDR5 32.0 GB
GPU	NVIDIA GeForce RTX 4060 Laptop GPU 8GB Memory

Table 1 – Hardware specifications used on the experiments

3.2 Data Sourcing

The news articles used for this study were retrieved from the web page of Hellenic Shipping News. This publisher circulates daily numerous articles collected from various sources, such as Reuters and Financial Times. These articles are catalogued under diverse categories, allowing for a better selection of the textual data set. Along these categories are: “Dry Bulk Market”, “Port News”, “Shipping Law News”, etc. However, only articles under the “International Shipping News” category were collected for this study.

The selected category allows for a data set with greater diversity. The purpose of a diverse collection of articles is to study the effectiveness of different feature-extraction methodologies.

3.3 Web Scraping

The web scraping algorithm used was customized to iteratively extract article content. The Selenium library was used for its WebDriver and the BeautifulSoup library for parsing the retrieved html. Initiating on the latest page listing articles, the web-scraping algorithm accesses each article in the list, before navigating to the next page, and extracts the following information:

- URL
- Publishing date
- Title
- Content

In each iteration of the algorithm, these features are concatenated in JSON format to a txt file named “articles.txt”. Due to networking issues – service timeout from the web site – and restarting of the scraping algorithm, multiple article files were generated and merged at the end.

The daily C5 route freight rates were acquired from the Clarksons Intelligence Network (Clarksons, n.d.).

3.4 Uploading to HuggingFace

HuggingFace (https://huggingface.co) allows for free storage and distribution of datasets and models. For the purposes of this study, replicability of the results, and free use of other studies, the complete dataset, together with backups and versions with features, were uploaded to that platform.

The different versions, with features and embeddings, were transformed to datasets, using the *datasets* library (Lhoest et al., 2021), and uploaded to HuggingFace using the appropriate token in the login function from the huggingface_hub library (Hugging Face, 2023).

3.5 LLM prompting

The LLM used for this study was the model DeepSeek-V3-0324 by DeepSeek, Inc (2024). The choice of this model was due to the ease of use, speed and low cost. Other models were examined and rejected for this study based on the above reasons. The OpenAI models provide ease of use, but the batching features, which lowers overall cost, does not allow JSON format answers and that could lead to not uniform features, despite setting the temperature to zero. The Llama models of Meta AI (2023) would have required setting up a virtual machine on the cloud, due to the size of the models, which would have increased the cost of this study and introduced unnecessary complexity. The Gemini model by Google DeepMind (2023) and Microsoft's Azure AI (2023) were much more complex and costly compared to DeepSeek's model and thus were rejected.

Although the use of a single LLM introduces a bias that cannot be explored, additional models would widen the scope of this study to a greater extent. Future work could explore the use of multiple models, while this study explores different approaches for prompting these models to extract features.

3.5.1 Feature Extraction – FE

Four different approaches were implemented for feature extraction using prompts on the DeepSeek LLM. The first approach was a single prompt (Appendix A), containing the article content and instructions on how to derive the features. This simple approach was used as a benchmark to gauge the merits of the following, more complex approaches.

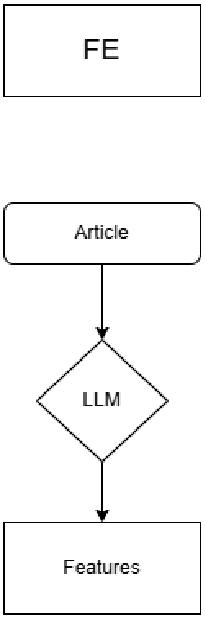


Figure 3 – Simple feature extraction approach using LLMs

The features extracted in every approach are the following:

1. Scale

The extent of the impact the event in the article has in terms of geographical scale. It could be regional, national or global.

2. Type of Vessel

The type of vessel affected by the event in the article. It could be Dry or Other, for ease of encoding and filtering in the classification and regression.

3. Vessel Size

The vessel size category affected by the events in the article. It could be Capesize, Panamax, Supramax, Handysize, Other, or All.

4. Sea Route

The affected sea route. These events could affect the C5, C3, All routes, or Other.

5. Duration of Positive Impact

The length of time the effect of each event is expected to last. It could be Short, Medium, or Long.

6. Impact

The direction of potential impact in hire rates caused by the event. It could be Negative, None, or Positive.

7. Hire rate impact

The magnitude of the event's impact. It could be None, Minor, Moderate, Significant, or Major.

3.5.2 Retrieval-Augmented Generation – RAG

For the purpose of improving prompt context, a database of relevant literature was created. The database consists of over four thousand chunks of text, sourced from maritime books and academic articles. The text chunks were of size 512 and were created with an overlap of 50 words.

Further study on the optimization of these two figures is of importance, but for simplicity and since the focus of the study is on prompting, no further examination was made. After chunking the literature, the text was transformed to embeddings using the pre-trained BERT model "all-MiniLM-L6-v2" (Reimers & Gurevych, 2019; Hugging Face, n.d.). The resulting dataset was uploaded to HuggingFace for ease of use and future work.

In this prompting methodology, each article is transformed using the same sentence transformer. The cosine similarity is calculated for the article embedding and the entire Literature database. The similarity scores are sorted, and the top thirty chunks are merged creating a string named context. The created context is then included in the prompt (Appendix B), along with the article content and relevant instructions for the LLM.

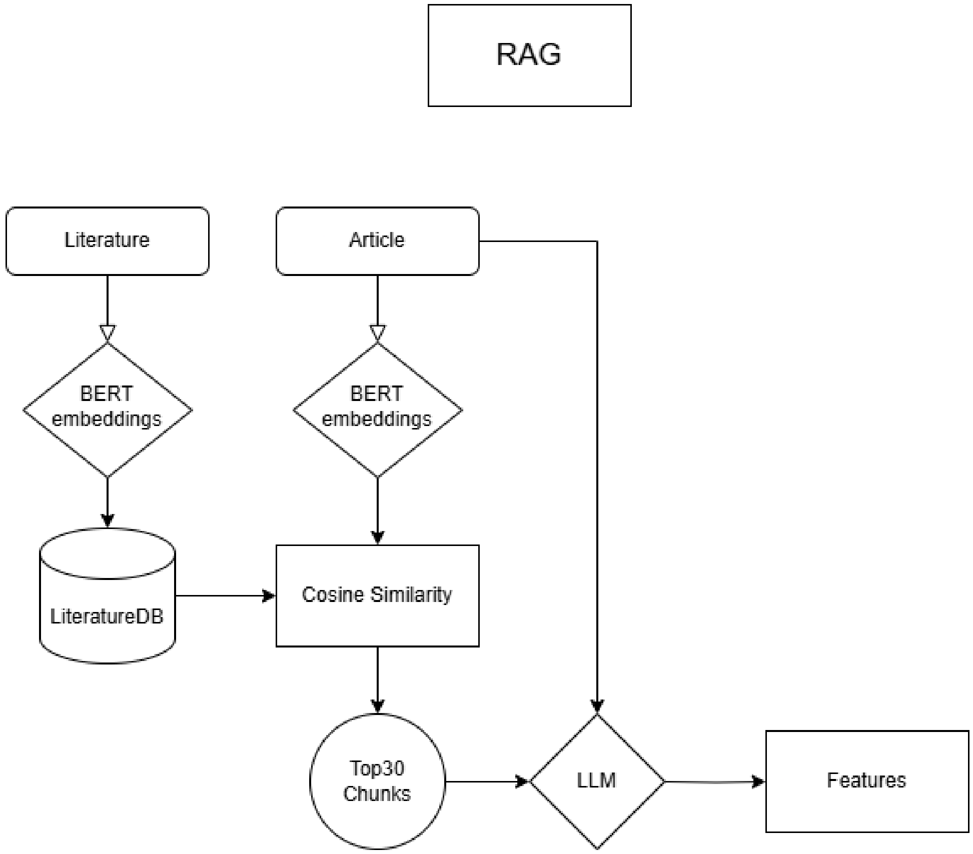


Figure 4 – Retrieval augmented generation and feature extraction approach

3.5.3 Informed Decision Feature Extraction – IDFE

In the IDFE approach the LLM is first prompted to filter for relevancy 10 from the resulting top 30 similar chunks. After that, the selected 10 chunks create the context that is used in the prompt along with the article content and the relevant instructions for the extraction of features. The purpose of this approach is to fine-tune the context that will accompany each article on the prompt for feature extraction (Appendix E) and thus creating a more "informed" decision by the LLM (Appendix C).

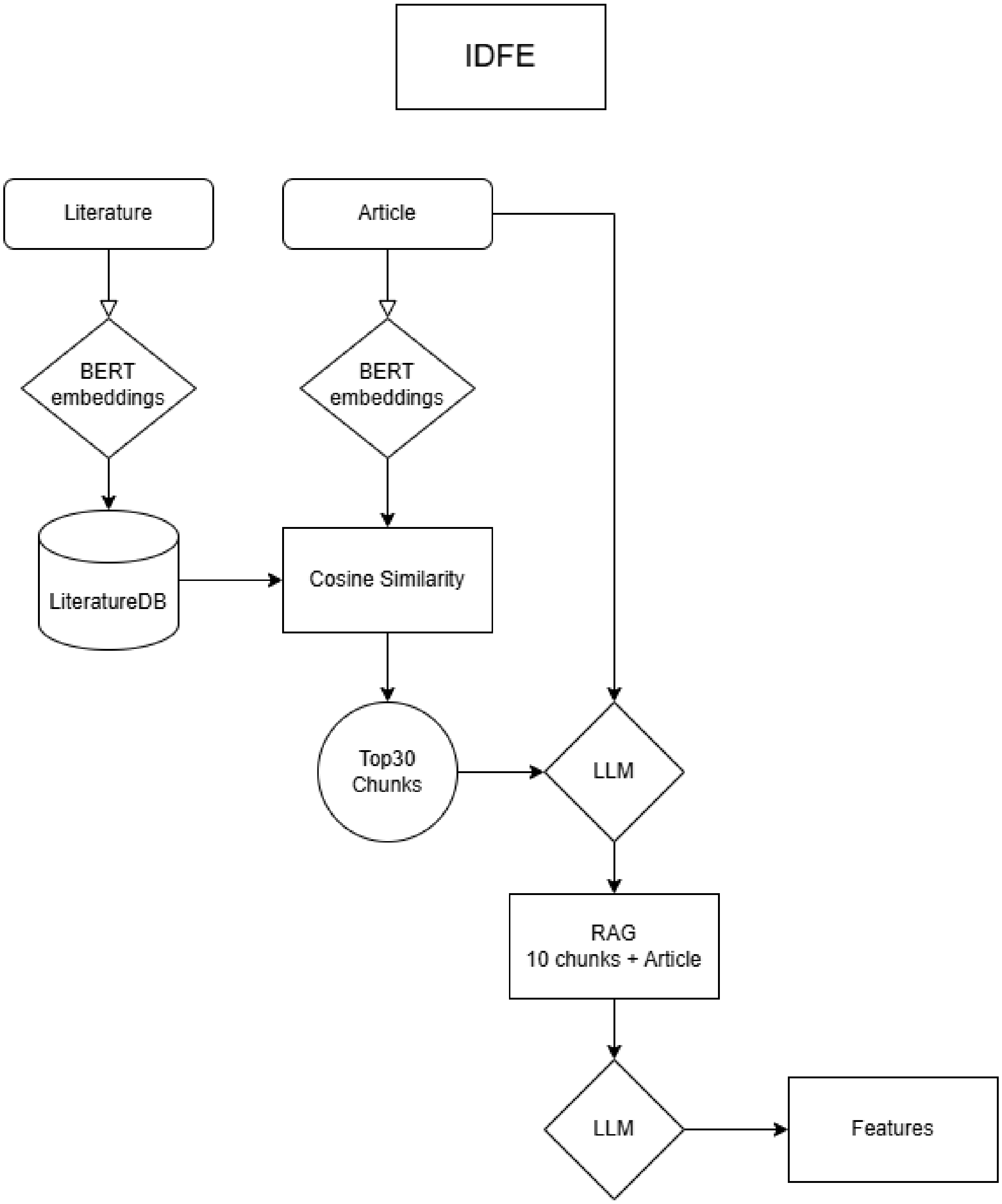


Figure 5 – Informed decision feature extraction approach. Hedging RAG methodologies in combination with LLM prompt engineering

3.5.4 Informed Decision Reason Based Feature Extraction – IDRBFE

In order to find the most relevant chunks, in this approach the LLM is prompted (Appendix F) to extract the key points pertaining to the feature that will be extracted. It is also tasked with generating search points. This text is then used for retrieving similar context chunks from the literature database. The chunks are filtered again by prompting the LLM (Appendix G) to select the most relevant chunks. The resulting chunks create the context which is used along with the article content and relevant instructions to create the final prompt (Appendix D) for feature extraction.

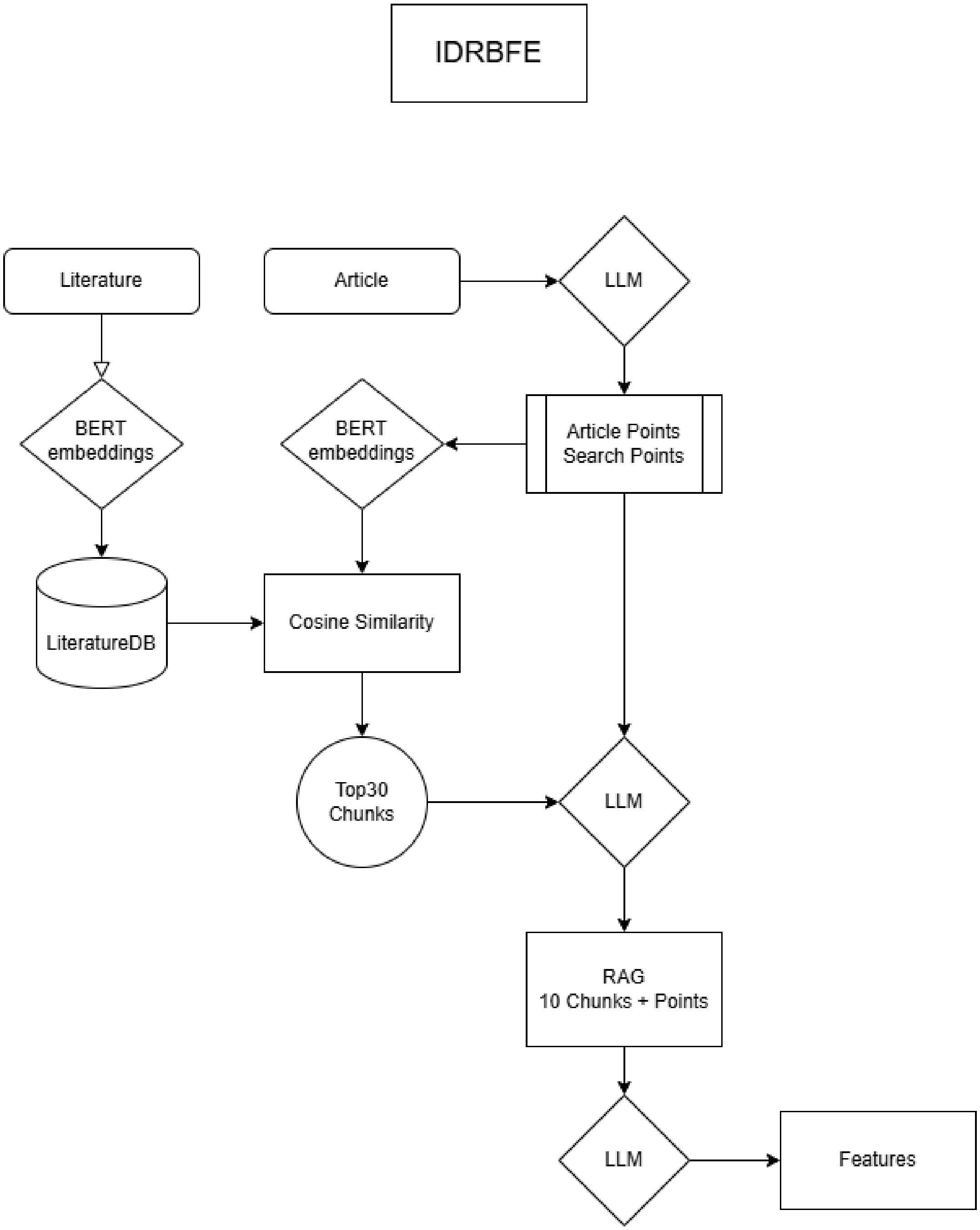


Figure 6 – Informed decision reason-based feature extraction approach. A more complex combination of RAG and LLM prompt engineering methods

All resulting features are saved in corresponding columns. The resulting Pandas Dataframe is transformed into a dataset and uploaded to HuggingFace for ease of use and future work.

3.6 Encoding of features

The extracted features were encoded to simplify and accommodate the process of training machine learning models for classification. The values extracted for these features are not completely uniform and, in some cases, the LLM returned unexpected values. For this reason, in the encoding phase multiple values per feature needed to be accounted for.

The “Scale” feature was encoded as follows:

‘Global’ was set to 2.

‘Regional’ and ‘Local’ were set to 1.

‘None’ and other values were set to 0.

The “Type of vessel” feature was encoded as follows:

‘All’, ‘Bulk Carrier’, and ‘Dry’ were set to 1.

All other values were set to 0.

The “Size of vessel” feature was encoded as follows:

'All', 'Capesize', 'Kamsarmax', 'Newcastlemax', 'Capesize, Panamax' 'Capesize, Panamax, Supramax' were set to 1, while everything else was set to 0.

The “Sea route” feature was encoded as follows:

'All' and 'C5' were set to 1 and everything else was set to 0.

These above-mentioned features were necessary for filtering of data used in the regression models and subsequent forecasting.

The duration feature was essential for the creation of the impact features used with the regression models and was encoded accordingly as follows:

'Short-term', 'Short term', or 'Short' were set to 30, which corresponds to an effect of the respective event having a duration of 30 days.

'Medium-term', 'Medium term', or 'Medium' were set to 90, which corresponds to an effect of the respective event having a duration of 90 days.

'Long-term', 'Long term' or 'Long' were set to 365, which corresponds to an effect of the respective event having a duration of 365 days.

Every other value was set to 0, to avoid errors in the subsequent steps.

The “Impact” feature was encoded as follows:

The values 'Positive' and 'Positive Impact' were set to 1, and 'Negative' and 'Negative Impact' values were set to -1. Everything else was set to 0, to account for non-values, “neutral” values and other non-uniform values.

The “Hire Rate Impact” feature was encoded as follows:

'Major' values were set to 4, 'Significant' to 3, ‘Moderate’ to 2, ‘Minor’ to 1, and everything else to 0.

After completing the encoding of these features for all four LLM extracting methods, the resulting Pandas Dataframe was transformed to a dataset and uploaded to HuggingFace for ease of use and future work.

3.7 Lexical Sentiment Analysis Tools

3.7.1 VADER

The lexicon used for this study was the VADER lexicon from the NLTK python library. VADER, known for its effectiveness in social media sentiment analysis, was selected due to its nuanced handling of negative and positive sentiment polarity in short texts typically found in news snippets.

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a sentiment analysis tool specifically designed for analyzing text from social media, but it is effective in general text as well. It's commonly used due to its speed, simplicity, and accuracy, especially with short and informal texts.

VADER uses a lexicon, which is essentially a large dictionary of words and their sentiment scores, collected from human raters. Each word is assigned sentiment intensity scores ranging from strongly negative to strongly positive. For example, the word "happy" has a strongly positive sentiment, and the word "terrible" a strongly negative sentiment. The word "okay" would be a neutral or mildly positive sentiment word

When VADER analyzes a sentence, it:

- Breaks down the text into individual words.
- Looks up each word in its lexicon.
- Assigns a sentiment intensity to each word found in the lexicon.
- Computes an overall sentiment score for the sentence.

It produces four scores:

- Positive (pos): Proportion of text identified as positive.
- Negative (neg): Proportion of text identified as negative.
- Neutral (neu): Proportion of text identified as neutral.

- Compound (compound): Aggregated and normalized score, ranging from -1 (extremely negative) to +1 (extremely positive).

The compound score is computed by combining the scores of all words, adjusting for negations (not good), emphasis (REALLY good), punctuation (great!!!), emojis, slang (meh), and capitalization (GREAT vs. great). Then the algorithm applies rules and weights to reflect emotional intensity accurately. Finally, the compound score is normalized to a scale between -1 and 1.

VADER accounts for several language patterns and contextual modifiers, such as, negations, not bad reduces negativity, potentially becoming positive, and intensifiers, words like very, extremely, super amplify the sentiment. Vader also accounts for punctuation and capitalization. A text consisting of “AMAZING!!!” would score higher than “amazing”. Emoji and emoticons play a role in the sentiment analysis too.

Typical interpretation of the compound score is as follows:

Compound score	Sentiment Interpretation
≥ 0.05	Positive
> -0.05 and < 0.05	Neutral
≤ -0.05	Negative

Table 2 – VADER compound scores’ interpretation

VADER is fast and computationally lightweight. It is effective at analyzing informal, short texts like tweets and reviews and handles nuances of language well. However, it relies on a predefined lexicon, which means that it has no contextual learning beyond built-in rules. It is not as effective with highly complex or nuanced text and it is even less effective with highly domain-specific language, as is the case for the data set of this study.

A Python function, vader_score, was defined to calculate sentiment analysis scores for a given text input using the Vader sentiment analysis tool (from NLTK's Vader SentimentIntensityAnalyzer). The function vader_score takes two parameters:

text: the input text to be analyzed.

sia: an instance of SentimentIntensityAnalyzer.

The function calculates the sentiment polarity scores from the input text, which are stored in a dictionary containing the following key-value pairs:

- 'neg': Negative sentiment score (0 to 1).
- 'pos': Positive sentiment score (0 to 1).
- 'neu': Neutral sentiment score (0 to 1).
- 'compound': Overall sentiment score (-1 to 1), indicating overall emotional sentiment.

Each individual sentiment score is extracted from the dictionary and the function returns them as a Pandas Series, which is helpful for integration into Pandas DataFrames and further analysis.

This function is useful for the sentiment analysis task of this study. The resulting features are filtered based on the respective route and vessel information gathered from the IDRBFE LLM process, so that only the relevant sentiment to the C5 rates is included. The filtered sentiment was then used in predictive regression models and the accuracy of them compared to benchmarks.

3.8 Summarization and Embeddings

For the purpose of creating a more compact textual dataset, the articles were summarized using the pretrained “bart-large-cnn” model by Facebook from HuggingFace (Lewis et al., 2020; Hugging Face, n.d.). The input text was trimmed to 1024 words, to account for unusually long text data, and the maximum number of output tokens was set to 450. This step allowed for the creation of complete embedding vectors from text that is not being truncated by the transforming algorithm of the following step. The model used for this transformation was the pretrained BERT model “bert-base-uncased", retrieved from HuggingFace (Devlin et al., 2019; Hugging Face, n.d.). Several methods to transform the summarized articles to embeddings were employed in this step and the resulting values were saved in respective columns. The first column created stored the complete embeddings vectors. Each summarized article was tokenized using the NLTK sentence tokenizer, and the resulting sentences were then transformed to embeddings and appended to a list containing each sentence vector. The resulting list of lists was stored in the main Pandas Dataframe with the articles and summaries and then uploaded to HuggingFace for ease of use and future work.

This column was used to create aggregated vectors. The aggregation functions used were mean, max, mix, which was the sum of mean and max, sum, concatenation, which was a flatten version of the list of sentence embeddings, called ‘concat’, and finally a mix of sum, mean, max, and min, called ‘mix2’.

These aggregated vectors were stored in the same Pandas Dataframe, which was then transformed into a dataset and uploaded to HuggingFace.

3.9 Classification Models

3.9.1 Classic Machine Learning Models

Classification modeling involved a comprehensive evaluation of diverse algorithms from the Scikit-Learn Python library, each selected for their unique strengths in handling various data characteristics and classification challenges. Due to the need of predicting seven separate and very different categorical features, a diverse collection of classifiers was needed to be employed for this step.

- Logistic Regression (Hosmer, Lemeshow, & Sturdivant, 2013). A widely used statistical model for binary classification tasks. It estimates the probability that a given input belongs to a particular class by modeling the relationship between the input features and the log-odds of the outcome. Due to its simplicity and interpretability, Logistic Regression is especially effective for problems where the classes are linearly separable or approximately so. It provides coefficients that directly indicate the influence of each feature on the prediction, making it valuable for understanding model behavior.

- Support Vector Classifier (SVC) (Cortes & Vapnik, 1995). A powerful supervised learning algorithm used for classification tasks. It is effective in high-dimensional spaces and works well with nonlinear decision boundaries by employing kernel functions to transform input data into higher-dimensional feature spaces. SVC seeks to find the optimal hyperplane that maximizes the margin between classes, thereby enhancing generalization performance. Its robustness and flexibility make it widely used in various applications, including text classification and bioinformatics.
- Decision Trees (DT) (Quinlan, 1986). Supervised learning algorithms that split data recursively based on feature values to create a tree-like model of decisions, capturing non-linear relationships in data. They are intuitive and provide clear interpretability by showing feature-based decision paths.
- Random Forests (RF) (Breiman, 2001). Extend Decision Trees by constructing an ensemble of multiple trees, each trained on random subsets of data and features, which enhances predictive accuracy and reduces overfitting. Additionally, Random Forests offer measures of feature importance, aiding in understanding which variables most influence model predictions.
- K-Nearest Neighbors (KNN) (Cover & Hart, 1967). A simple, instance-based learning algorithm used for classification and regression. It classifies a data point based on the majority label of its closest

neighbors in the feature space. KNN is non-parametric and adaptable to varying class distributions, making it effective in scenarios where the decision boundary is irregular or unknown. Its simplicity and ease of implementation contribute to its widespread use in various machine learning applications.

- Voting Classifier and Bagging (Breiman, 1996; Kuncheva, 2004). Ensemble learning techniques that combine multiple base learners to improve predictive performance and robustness. The Voting Classifier aggregates predictions from diverse models through majority voting for classification or averaging for regression, enhancing stability and accuracy. Bagging (Bootstrap Aggregating) builds multiple models on different bootstrap samples of the training data, reducing variance and preventing overfitting. Both methods leverage the strengths of individual learners while mitigating their weaknesses.
- Histogram-based Gradient Boosting Regression (Hist-GBR) (Ke et al., 2017). An efficient variant of gradient boosting that leverages histogram binning of continuous features to accelerate training. This approach significantly reduces computational complexity and memory usage, making it highly scalable and fast, especially on large datasets. It also handles categorical features effectively by grouping them into bins, enhancing performance in applications involving mixed data types.

- The Stacking Classifier (Wolpert, 1992). An ensemble method that combines several base models by training a meta-model to optimally integrate their predictions. This approach exploits the complementary strengths of diverse classifiers, often resulting in improved overall predictive accuracy and robustness. By learning how to best weight or combine base model outputs, stacking can reduce errors and enhance model generalization.
- AdaBoost (Adaptive Boosting) (Freund & Schapire, 1997). An ensemble learning algorithm that sequentially trains weak classifiers, typically decision stumps, by focusing more on previously misclassified instances. Through iterative reweighting of the training data, AdaBoost adaptively improves the model's performance by combining these weak learners into a strong classifier. Its ability to concentrate on difficult cases makes it effective in enhancing prediction accuracy and reducing bias.
- Extra Trees (Geurts, Ernst, & Wehenkel, 2006). An ensemble learning method that builds multiple decision trees by introducing additional randomness in the split selection process compared to Random Forests. Instead of searching for the optimal split, Extra Trees choose splits at random, which increases variance reduction and computational efficiency. This method is effective at capturing complex patterns while reducing overfitting, making it a robust choice for various predictive tasks.

These models were systematically compared to identify the most accurate classifier for each feature. Initially every model was trained, and its predicting accuracy was stored in a list. In this step the parameters of every model were set by default. In order to study the most effective size of a training set retrieved from the LLM, the classifiers were iteratively trained with data of size 1000, 2000, 3000, 4000, and 5000. This was also done to provide insight into the length of training sample required from the LLM feature extraction.

The same iterative training followed for each model, a training data set of size 5000, and only for the flattened sentence embeddings, this time optimizing the models' hyperparameters through a grid search. The parameter grids used are as follows:

- LogisticRegression:
  - C: Controls regularization strength. Lower values increase regularization, reducing overfitting, while higher values decrease regularization, potentially improving training accuracy but risking overfitting. Values selected for the grid-search: [0.001, 0.01, 0.1, 1, 10, 100]
- SVC (Support Vector Classifier):
  - C: Similar to Logistic Regression, C balances the trade-off between achieving a low training error and a low testing error. Values Selected: [0.01, 0.1, 1, 10, 100]

- kernel: Determines the kernel function used, transforming input space into higher-dimensional space for better separability. Selected values: [linear, rbf, poly, sigmoid].
- gamma: Defines how far the influence of a single training example reaches. Scale adapts to feature variance, while Auto adapts inversely proportional to the number of features. Both values selected.

- RandomForestClassifier:
  - n_estimators: Number of trees in the forest; more trees typically improve accuracy but increase computation time. Values selected: [50, 100, 200, 300].
  - max_depth: Maximum depth of each tree, limiting complexity and preventing overfitting. Values selected: [None, 10, 20, 30, 50].
  - min_samples_split: Minimum number of samples required to split an internal node, controlling the size of the trees. Values selected: [2, 5, 10].
  - min_samples_leaf: Minimum number of samples required to be at a leaf node, reducing overfitting. Values selected: [1, 2, 4].
- DecisionTreeClassifier:
  - max_depth, min_samples_split, min_samples_leaf: Same as in Random Forest, controlling tree complexity and reducing

potential overfitting. The values selected for the grid search were the same, with the exception of max_depth, for which these values were selected: [None, 5, 10, 20, 30].

- KNeighborsClassifier:
    - n_neighbors: Number of neighbors considered for classifying a new instance; more neighbors smooth out decision boundaries. Values selected: from 1 to 20.
    - weights: Determines weight function used. "Uniform" treats all neighbors equally and "distance" weights closer neighbors higher.
    - metric: Distance metric (euclidean, manhattan, minkowski) affects how neighbors are identified.
- GradientBoostingClassifier:
    - n_estimators: Number of boosting stages, more typically enhance performance but risk overfitting. Values selected: [50, 100, 200].
    - learning_rate: Shrinks the contribution of each tree, balancing learning speed and performance. Values selected: [0.01, 0.1, 0.2].
    - max_depth, min_samples_split, min_samples_leaf: Similar roles as in decision trees, controlling individual tree complexity.
- VotingClassifier:

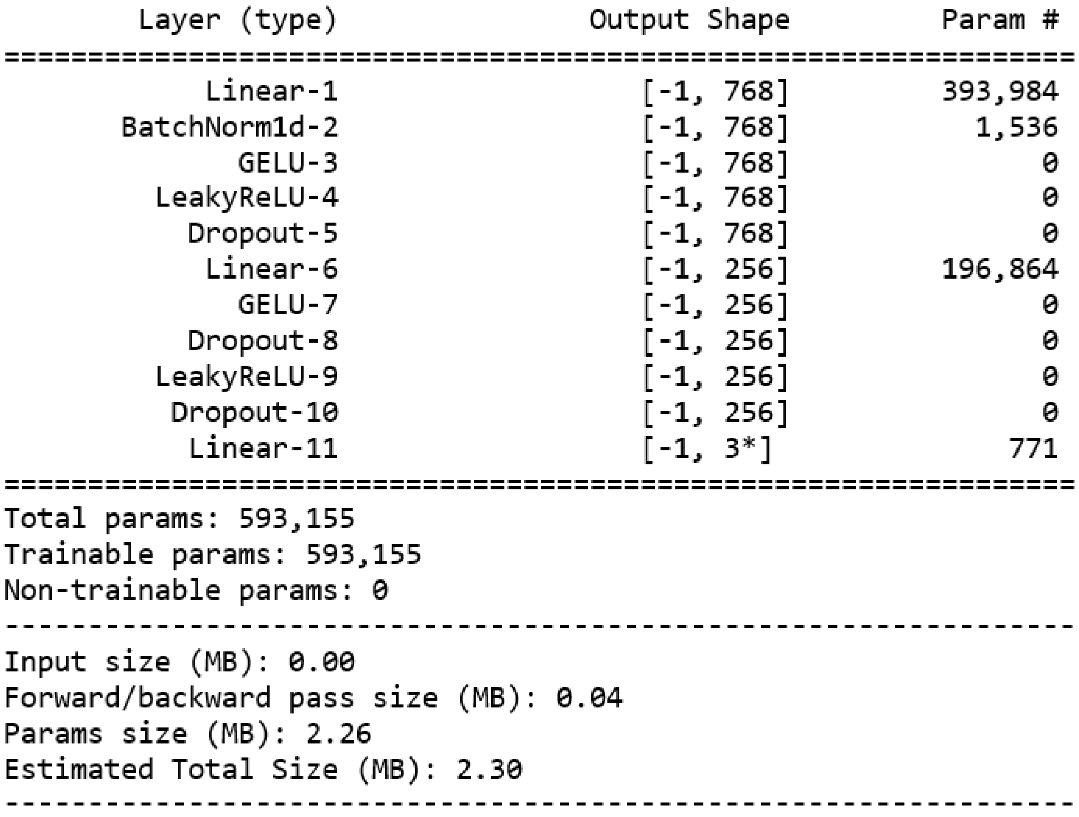
    - voting: Aggregation strategy. “Hard” uses majority voting, while “soft” uses weighted averages of predicted probabilities.
    - weights: Assigns relative importance to each estimator in voting. Values selected: [[1, 1, 1, 1, 1], [2, 1, 1, 1, 1], [1, 2, 1, 1, 1]].
- BaggingClassifier:
    - n_estimators: Number of base estimators (often decision trees) trained independently on subsets of data. Selected values: [10, 50, 100].
    - max_samples: Fraction of samples drawn to train each base estimator. Values selected: [0.5, 0.75, 1.0].
    - max_features: Fraction of features drawn for training each estimator, increasing diversity among estimators. Values selected: [0.5, 0.75, 1.0].
- HistGradientBoostingClassifier:
    - max_iter: Number of boosting iterations. Values selected: [50, 100, 200].
    - learning_rate: Controls the incremental contribution of each boosting iteration. Values selected: [0.01, 0.1, 0.2].
    - max_depth: Limits tree complexity. Values selected [3, 5, 7].
    - min_samples_leaf: Minimum number of samples at a leaf node, reducing complexity. Values selected: [1, 2, 4].
- StackingClassifier:

  - final_estimator__C: Regularization parameter for the final estimator (often Logistic Regression), controlling complexity. Values selected: [0.001, 0.01, 0.1, 1, 10, 100]
  - cv: Number of cross-validation folds for generating predictions used by the meta-classifier. Values selected: [3, 5].
- AdaBoostClassifier:
  - n_estimators: Number of weak classifiers iteratively trained. Values selected: [50, 100, 200].
  - learning_rate: Modifies the weight updates, influencing the impact of each weak classifier. Values selected: [0.01, 0.1, 0.2].
- ExtraTreesClassifier:
  - Similar parameters to Random Forest (n_estimators, max_depth, min_samples_split, min_samples_leaf), controlling complexity, diversity, and robustness of the ensemble.

After identifying the best models for each feature, the parameters of these models were optimized through a grid-search process and the predicted values of each feature were stored in the Pandas Dataframe with all articles. The Dataframe was transformed to a dataset and uploaded to HuggingFace for ease of use and future work.

3.9.2 Deep Neural Network for classification

The architecture of the deep neural network that was trained and made predictions for every feature is as follows:



```
        Layer (type)               Output Shape         Param #
================================================================
            Linear-1                 [-1, 768]         393,984
       BatchNorm1d-2                 [-1, 768]           1,536
              GELU-3                 [-1, 768]               0
         LeakyReLU-4                 [-1, 768]               0
           Dropout-5                 [-1, 768]               0
            Linear-6                 [-1, 256]         196,864
              GELU-7                 [-1, 256]               0
           Dropout-8                 [-1, 256]               0
         LeakyReLU-9                 [-1, 256]               0
          Dropout-10                 [-1, 256]               0
           Linear-11                 [-1, 3*]              771
================================================================
Total params: 593,155
Trainable params: 593,155
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.04
Params size (MB): 2.26
Estimated Total Size (MB): 2.30
----------------------------------------------------------------
```

Table 3 – Deep neural network architecture used for classification of features

The output of the neural network was not 3 nodes for every case, but the number of categorical values for each feature, which iteratively adjusted for every training and predicting process.

The training and validation sets were split 80-20, while the training process consisted of 10 warmup steps, 16 training steps and 5 epochs. The Adam optimizer (Kingma & Ba, 2015) was used, from the Pytorch library (Paszke et al., 2019), with a learning rate of 0.001 and 0.01 weight decay. By default the PyTorch library uses the cross-entropy loss function for classification problems, because it combines log-softmax and negative log-likelihood and it is suitable for classification tasks, especially with multiple labels.

3.9.3 Particle Swarm Optimization

This study employs Particle Swarm Optimization (PSO) to optimize hyperparameters of the classification neural network. Specifically, the PSO algorithm is utilized to find optimal values for four key hyperparameters: hidden layer dimension, dropout rate, learning rate, and batch size. The pyswarms library was used for this process (Miranda et al., 2020).

The neural network architecture consists of an input layer, one hidden layer with batch normalization, a GELU activation function, a dropout layer, and an output layer corresponding to the number of classification labels. The training utilizes the AdamW optimizer and Cross-Entropy Loss to handle classification tasks.

The PSO algorithm seeks to minimize an objective function defined as the negative weighted F1 score evaluated on a validation dataset. Minimization of the negative F1 score effectively maximizes classification performance. The F1 score balances precision and recall, making it ideal for evaluating models where both false positives and false negatives matter. Minimizing the negative weighted F1 score ensures the model accurately identifies relevant cases while reducing errors, improving overall prediction reliability. The PSO bounds are set as follows:

Hidden dimension: 64 to 412 neurons

Dropout rate: 0.1 to 0.5

Learning rate: 1e-4 to 1e-2

Batch size: 2 to 64

Due to hardware restrictions, these bounds as well as the number of active particles needed to be minimized.

The optimization proceeds by initializing a swarm of candidate solutions (particles), each representing a set of hyperparameters. The PSO algorithm iteratively updates particle positions based on both individual best positions and the global best position identified across all particles, guided by parameters c1 (cognitive coefficient), c2 (social coefficient), and inertia weight (w).

Each particle's performance is evaluated through training the neural network for a brief period of three epochs and calculating its F1 score. The particle's position leading to the best validation F1 score is retained as the global best hyperparameter set.

This approach leverages the capability of PSO to efficiently explore and exploit the hyperparameter search space, potentially leading to better-performing neural network models compared to traditional hyperparameter optimization methods.

3.10 XAI for classification

For model interpretability, explainable AI technique LIME (Local Interpretable Model-agnostic Explanations) was employed (Ribeiro et al., 2016). Specifically, LIME was used to generate local explanations by perturbing the input text and observing changes in predicted probabilities, leveraging the BERT-based embeddings as features.

To extract meaningful representations from textual data, articles were first segmented into sentences, and each sentence was transformed into embeddings using a pretrained BERT model (“bert-base-uncased”). Sentence embeddings were aggregated via mean pooling to create comprehensive article-level embeddings. These embeddings served as inputs to machine learning classification models trained to predict target labels.

In this study, textual data was used directly in the explainability analysis to obtain LIME values at the word level rather than at the embedding level. Although the predictive models were trained on aggregated BERT embeddings, these embeddings are high-dimensional and do not map directly to individual words or phrases in the original text. By applying LIME on the raw text inputs, it becomes possible to assess the contribution of specific words or terms to the model’s predictions. This granular insight into word-level importance is crucial for interpretability, as it allows for more intuitive explanations and better understanding of which textual features drive model decisions. Therefore, leveraging the original textual data for LIME ensures that explanations remain interpretable and actionable, aligning with the goal of transparent and explainable AI.

This approach provided insight into which textual components most influenced model predictions. Both true positive and false positive cases were analyzed to assess the model’s behavior under correct and incorrect predictions.

This methodology facilitates understanding of complex NLP-based classification models by linking input text features with prediction outcomes, enhancing transparency and trustworthiness in the forecasting process.

3.11 Feature engineering

For each LLM prompting method (FE, RAG, IDFE, IDRBFE) two new features of sentiment were created. The features were filtered for vessel type and size, as well as route and scale of each event. The two features were created to represent the positive and the negative market sentiments. By iterating through the filtered dataset, the positive and negative magnitudes are added on the respective rows, offseted by the duration feature.

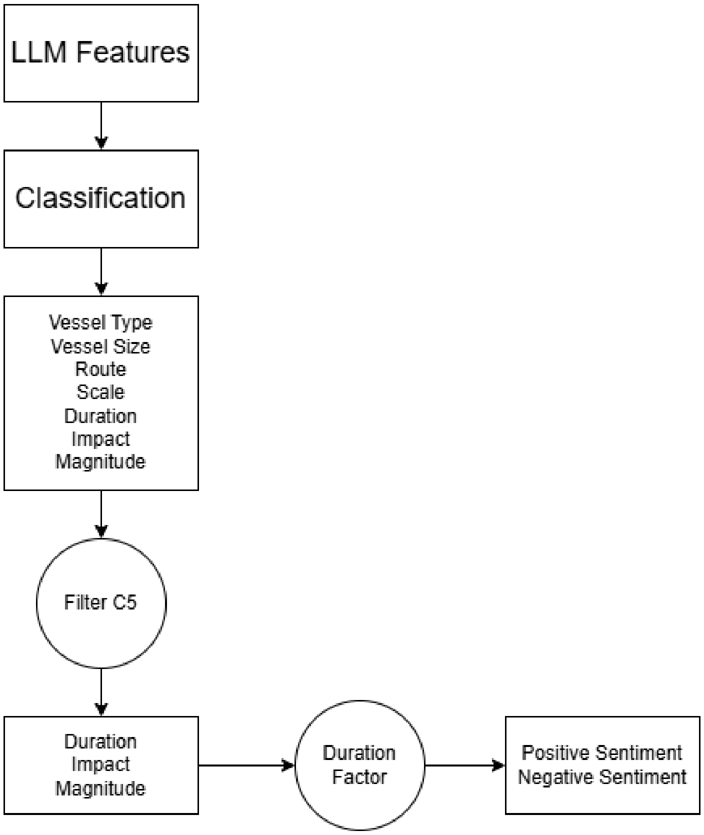


Figure 7 – Process of sentiment feature engineering

3.12 Pre-processing and Regression Models

The time-series data, sentiment and C5 rates, were daily values. These features were transformed into weekly data, to have a more practical predictive horizon, i.e. the rates of the following weeks, to present more valuable information compared to daily variations. The sentiment data was transformed into weekly data using mean and sum aggregation and both methods were tested for forecasting accuracy. Similarly, the sentiment derived from the VADER lexical analysis, was transformed into weekly data, and three methods were tested.

- The summation and average VADER features.
- The summation and average the polarity feature.

These four methods were included in the training dataset, first filtered for the C5 route and then unfiltered, to include all sentiments.

The features were structured using sliding-window methods, whereby each predictive sample comprised lagged historical values. The optimal length of historical data (window size) and forecasting horizon (how far ahead forecasts were made) were optimized through systematic grid search. This procedure involved iteratively testing combinations of window sizes ranging from 1 to 9 weeks and forecast horizons ranging from 1 to 5 weeks ahead.

Regression modeling involved extensive comparative analyses across several advanced statistical and machine learning algorithms:

- Linear Regression (Montgomery, Peck, & Vining, 2012). A fundamental statistical method that models the linear relationship between a dependent variable and one or more independent variables. It is widely used due to its simplicity, interpretability, and ability to provide baseline performance against which more complex models can be compared. The model estimates coefficients that quantify the influence of each predictor on the outcome, facilitating straightforward insights into data relationships.
- Ridge, Lasso, and ElasticNet (Tibshirani, 1996; Hoerl & Kennard, 1970; Zou & Hastie, 2005). Regularization techniques used in regression modeling to prevent overfitting and handle multicollinearity among predictors. Ridge regression adds an L2

penalty to shrink coefficient values, reducing model complexity. Lasso regression applies an L1 penalty, promoting sparsity by driving some coefficients to zero, thus performing feature selection. ElasticNet combines both L1 and L2 penalties, balancing the benefits of Ridge and Lasso, particularly useful when predictors are highly correlated.

- Support Vector Regression (SVR) (Drucker et al., 1997). An extension of Support Vector Machines (SVM) designed for regression tasks. SVR seeks to find a function that approximates the target values within a specified margin of tolerance, effectively handling nonlinear relationships through the use of kernel functions. Its ability to model complex patterns while controlling model complexity makes it a powerful tool for nonlinear predictive modeling.
- Random Forest (Breiman, 2001). Builds an ensemble of trees using bootstrap sampling and random feature selection, reducing variance and overfitting.
- Gradient Boosting (Friedman, 2001). Sequentially builds trees that correct errors of previous ones, improving accuracy.
- XGBoost and LightGBM (Chen & Guestrin, 2016; Ke et al., 2017). Efficient implementations of gradient boosting that optimize speed and scalability, handle large datasets effectively, and capture complex feature interactions. These methods are widely used due to

their strong predictive power and interpretability through feature importance.

Model training followed best practices in machine learning, involving data normalization via Min-Max scaling, cross-validation to ensure robust generalizability, and rigorous testing on chronological hold-out sets to avoid data leakage. Hyperparameter tuning for each model was automated through grid search and cross-validation to find the best-performing configuration. Performance metrics evaluated included Root Mean Squared Error (RMSE) and R-squared (R^2), providing complementary insights into model accuracy and explanatory power.

3.13 Diagnostics – Causality

To thoroughly validate the relationships identified by the models, several statistical tests were performed. Causality analysis methods such as the Toda-Yamamoto test (Toda & Yamamoto, 1995), Transfer Entropy (Schreiber, 2000), Convergent Cross Mapping (Sugihara et al., 2012), and PCMCI causal discovery (Runge et al., 2019) were used to confirm that sentiment-based features have a real influence on freight rates, and to rule out reverse effects or false connections. Transfer Entropy, in particular, helped measure how strongly and in which direction information flows between news features and freight rates. Additionally, multicollinearity was checked using the Variance Inflation Factor (VIF) to make sure the regression results were reliable and easy to interpret (O'Brien, 2007).

4. Experiments

4.1 Description of textual data

The total number of articles scraped was 40,013, covering the period 1 March 2020 through 18 April 2025. The average length of all articles was 509 words, while the standard deviation was 427 words. The longest article was 12017 words long and the shortest article 20 words long.

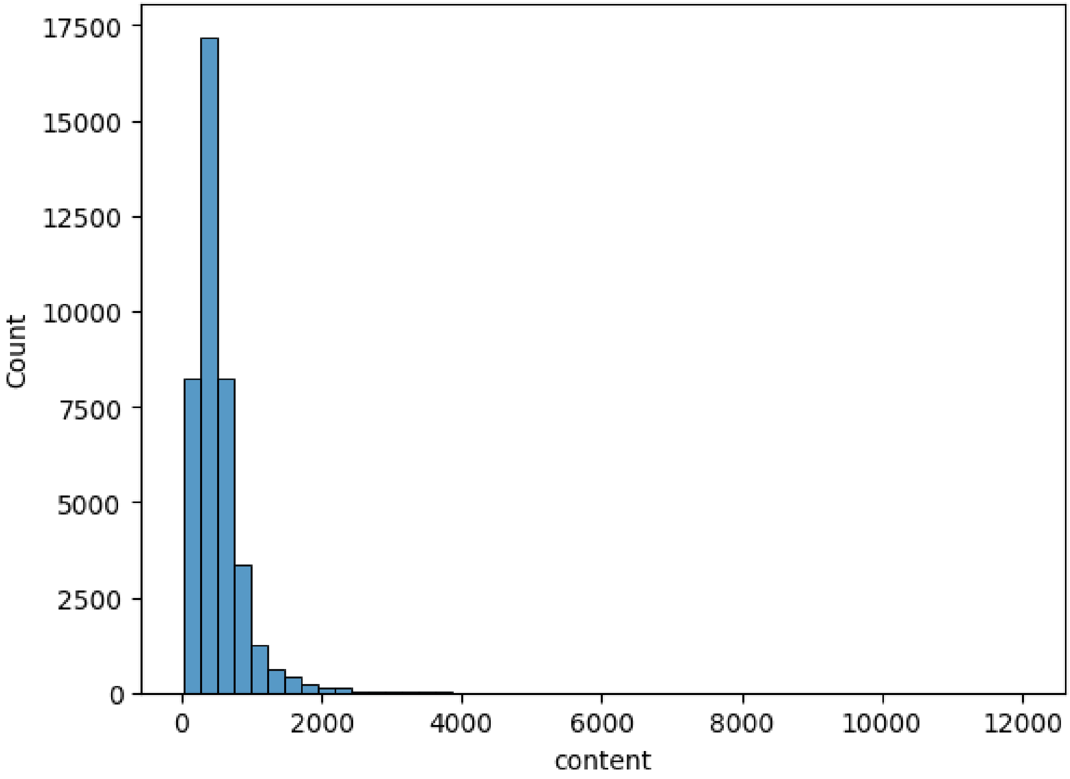


Figure 8 – Histogram showing the distribution of article lengths in words

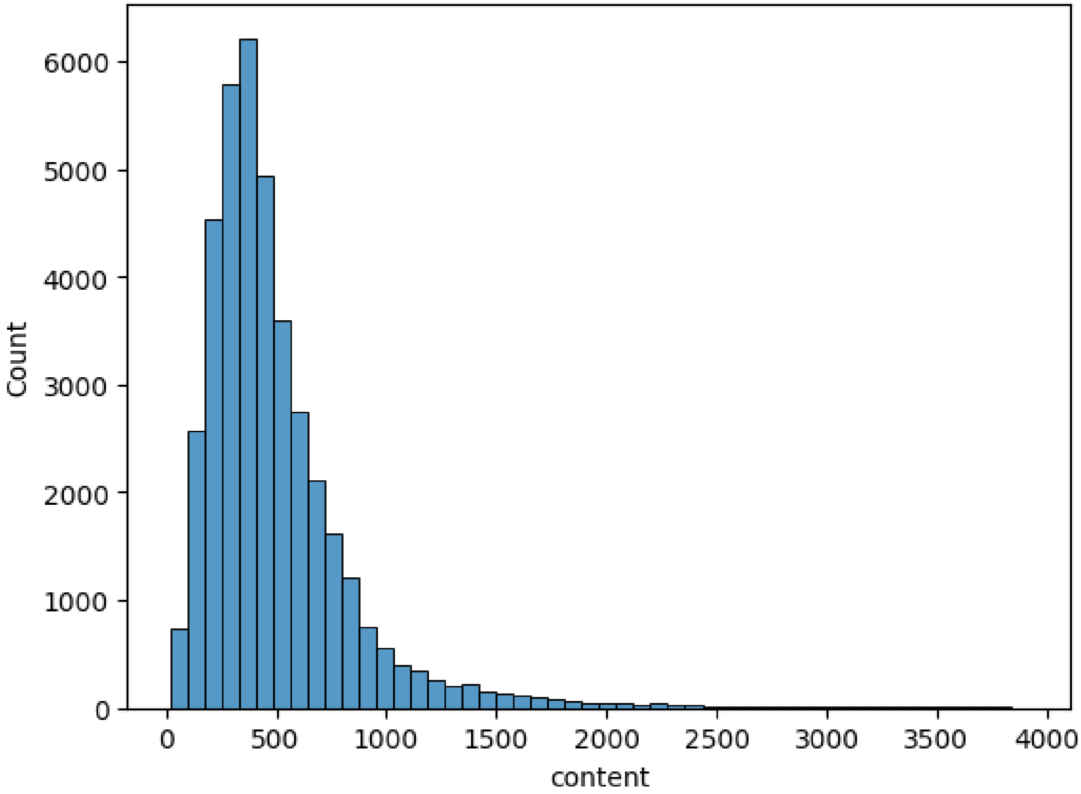


Figure 9 - Histogram showing the distribution of article lengths in words limited to maximum 4000 words

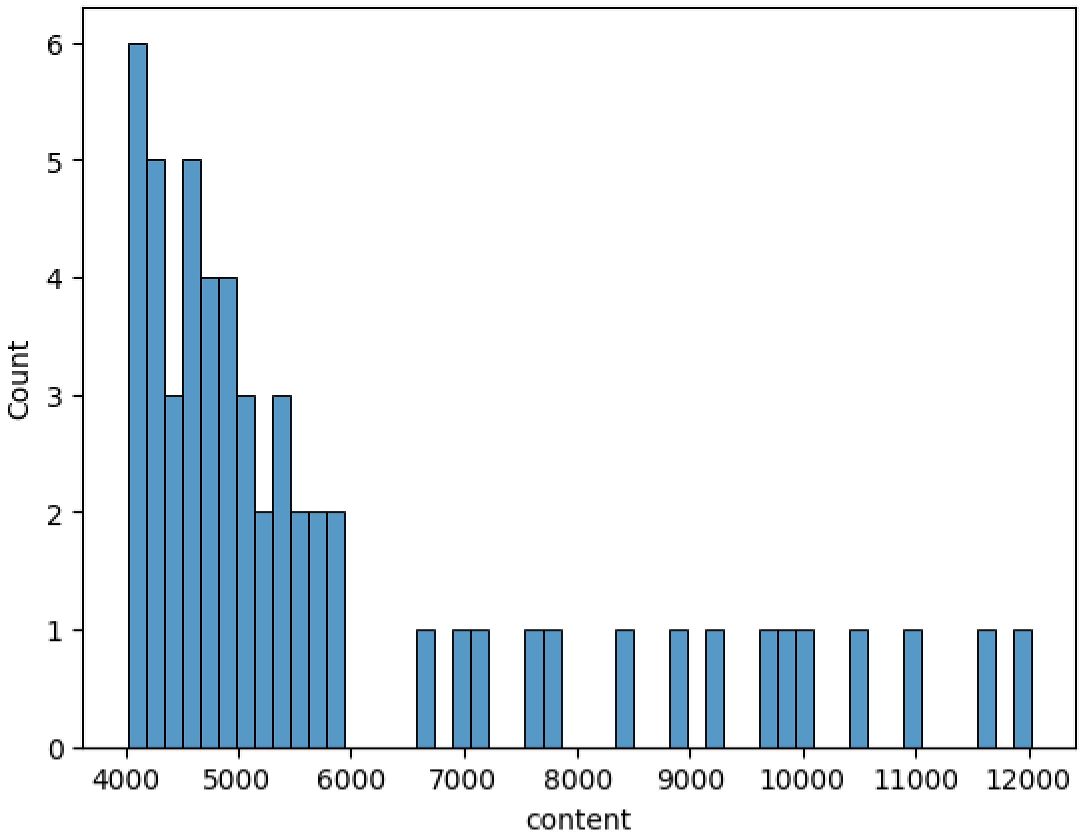


Figure 10 - Histogram showing the distribution of article lengths in words limited to minimum 4000 words

The most common words in the articles were “ship”, “vessel” and “year”. While the most common words in the titles of articles were “new” and “ship”.

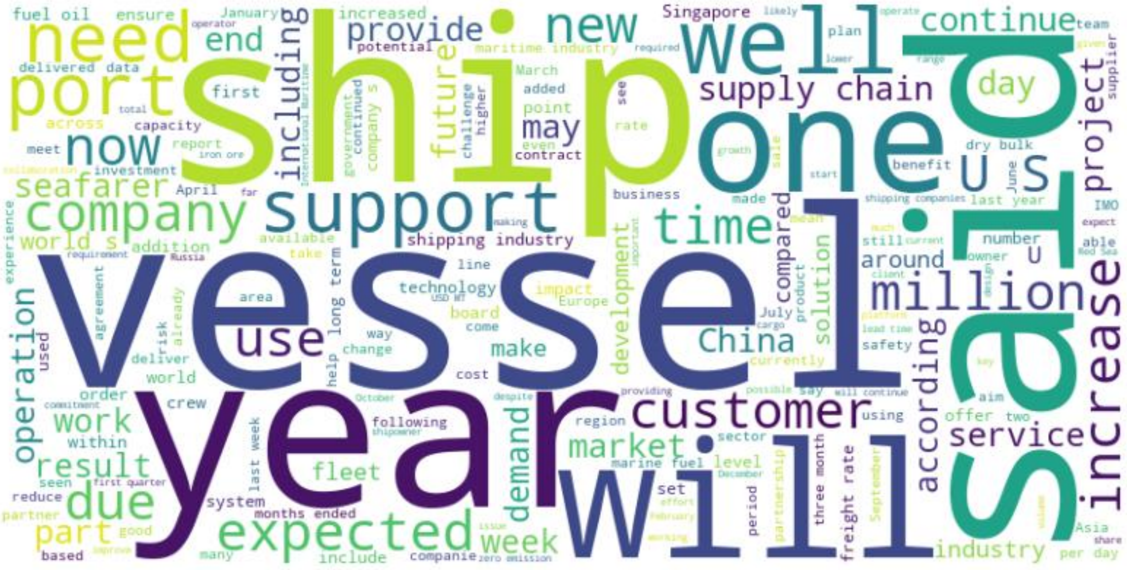


Figure 11 - Wordcloud of article contents

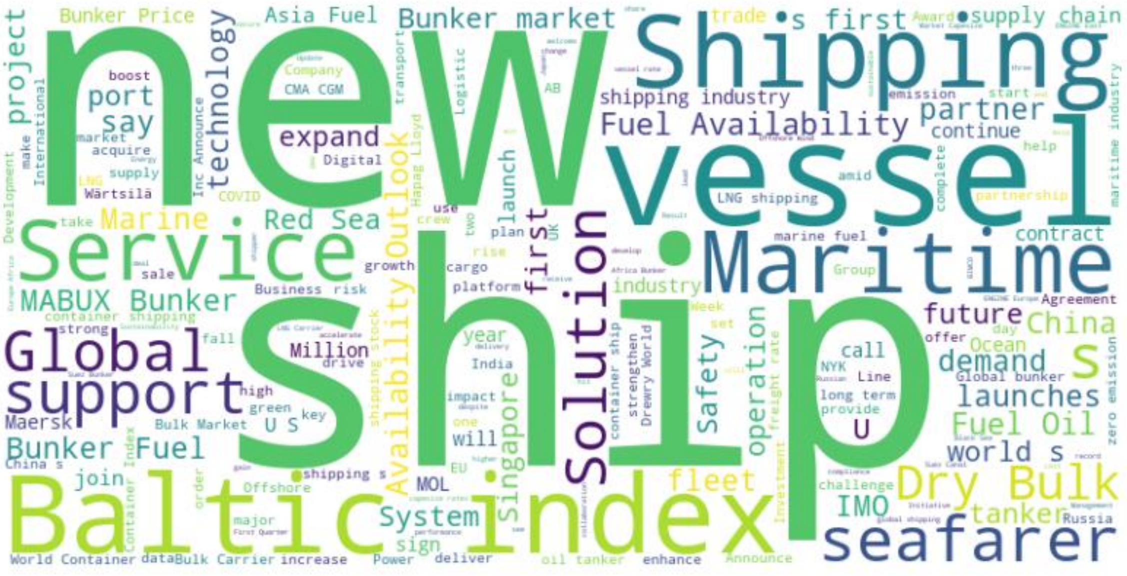


Figure 12 - Wordcloud of article titles

4.2 Classification Results

The results of the comprehensive training of classification models showed similar accuracy for the same features across all models, with a few exceptions. It is important to note that the results below are the product of training with all parameters being the defaults of the Scikit-learn library. Decision tree, Bagging, RandomForest, and ExtraTree classifiers performed slightly lower compared to the rest of the models.

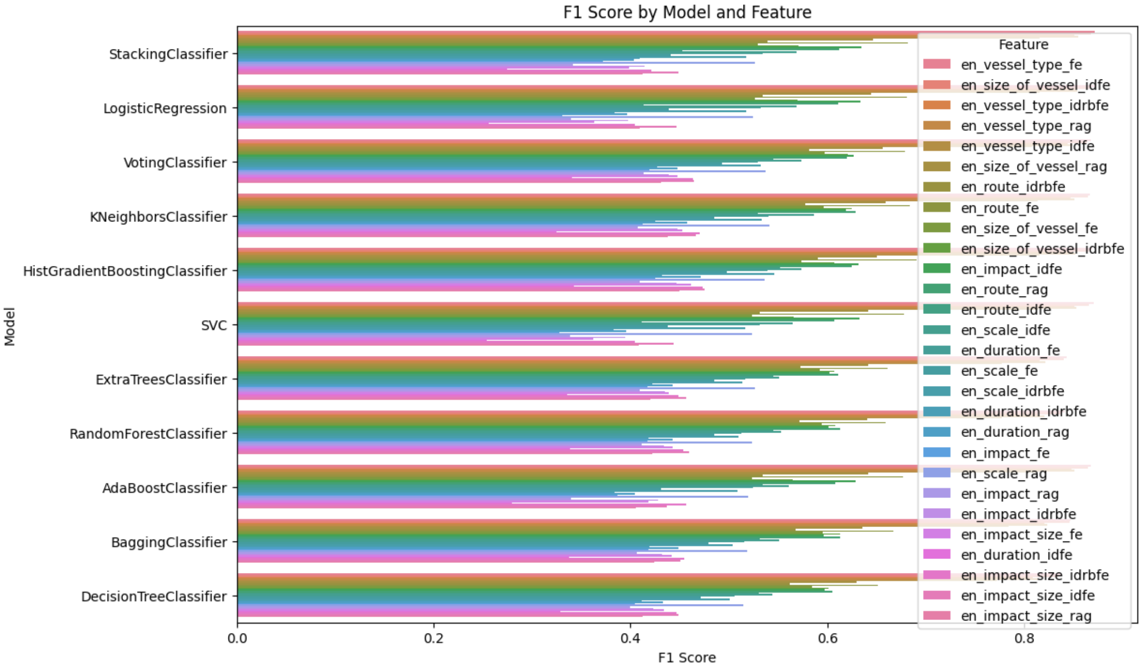


Figure 13 - F1 scores of various classification models across different feature sets

Notably, the relationship between classification accuracy and length of training data set is not linear, for all models, with only exception the Histogram Gradient Boosting Classifier. However, the results show that after a certain length of the training data-set the models' performance increases.

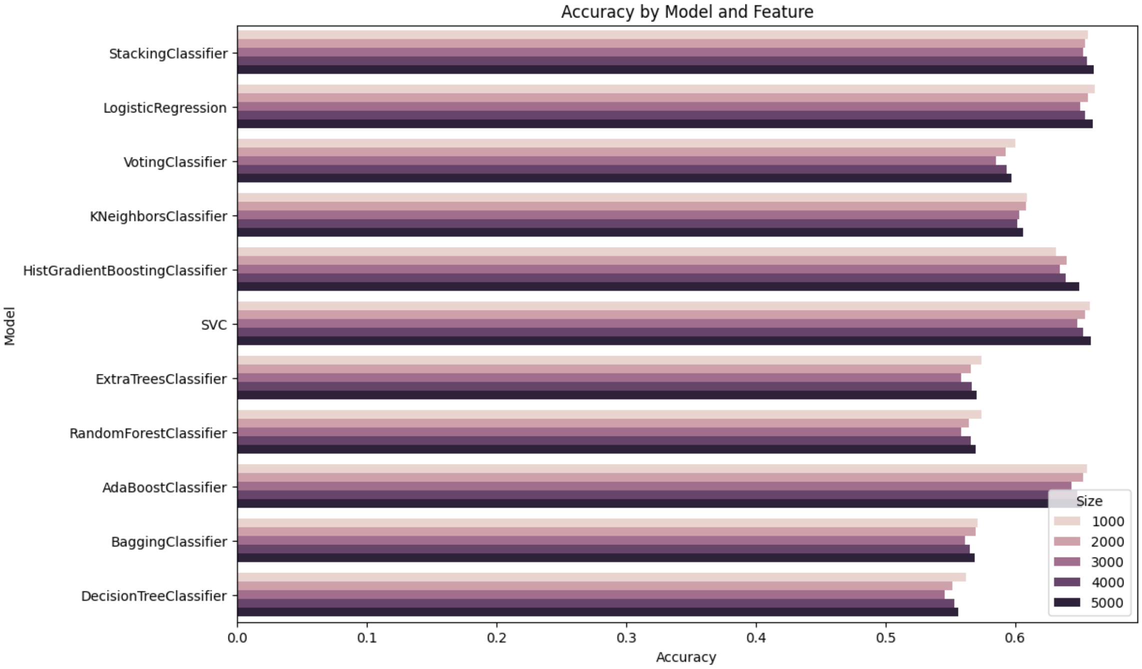


Figure 14 - Accuracy of different classification models across varying dataset sizes

Optimization results show a significant improvement in accuracy across all models and features.

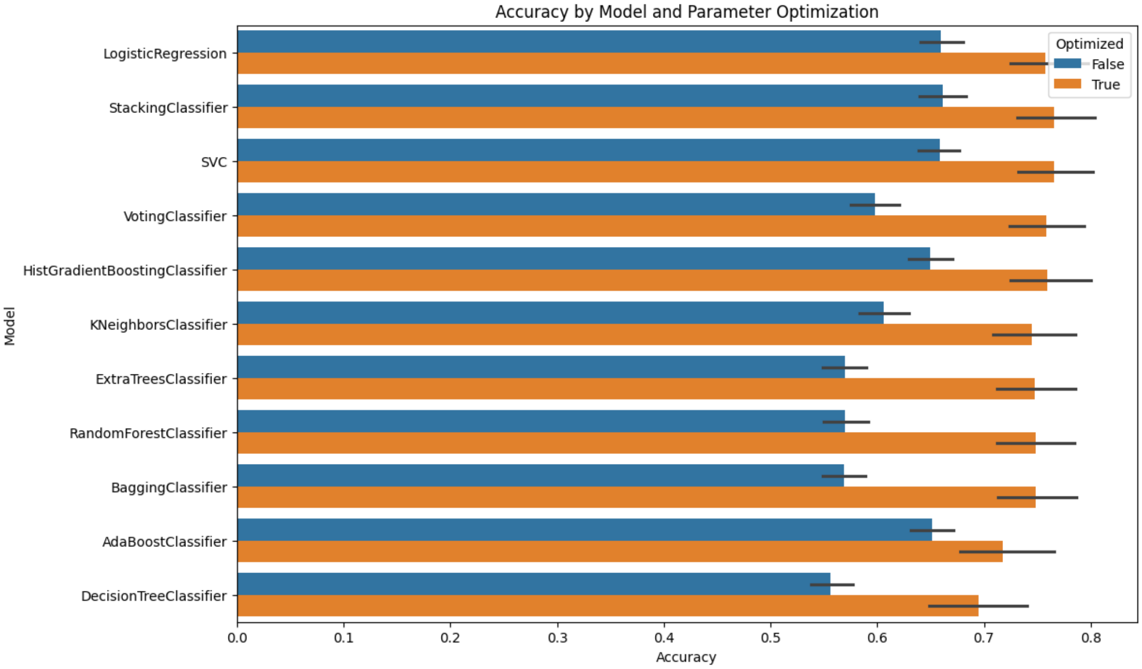


Figure 15 - Accuracy comparison of classification models before and after parameter optimization

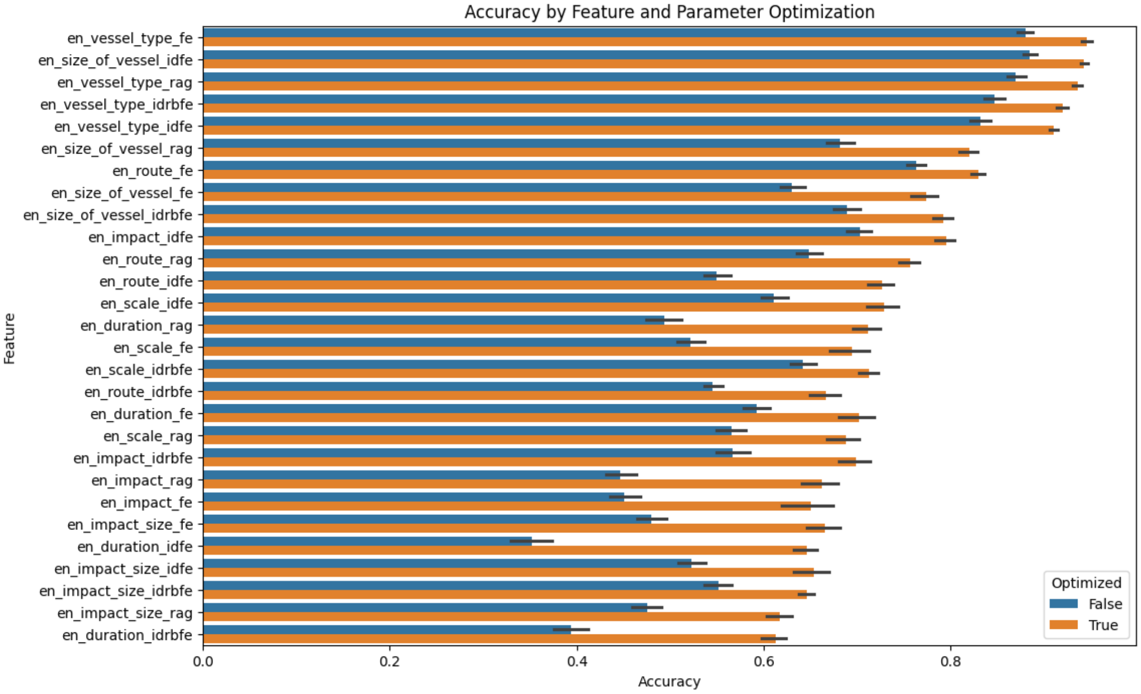


Figure 16 - Accuracy of different features before and after parameter optimization

4.2.1 Classification of FE method

The results show that certain features can be modeled with higher accuracy. These are the “Vessel type” and “Route”. However, the important features, such as “Impact” and “Magnitude”, are shown to have been modeled with very low accuracy.

From the results it can be seen that the number of training data can influence the predictive accuracy of classification models in very eratic way. The accuracy of the training set of 2000 points of data has the lowest value across all clasification models.

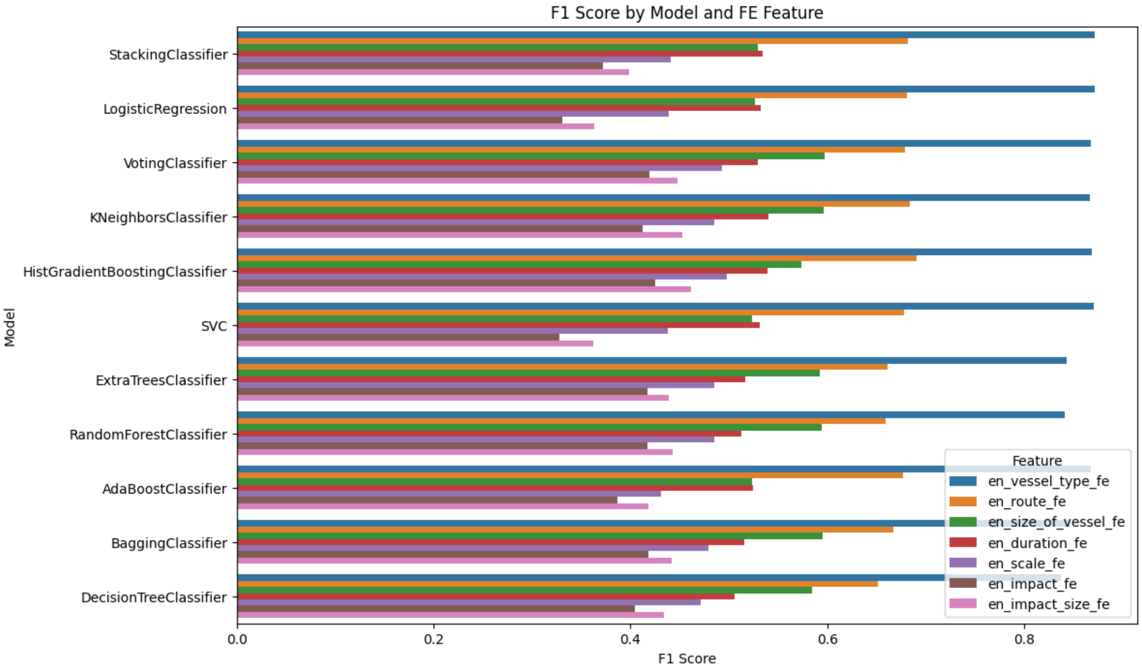


Figure 17 - F1 scores of different classification models evaluated across features

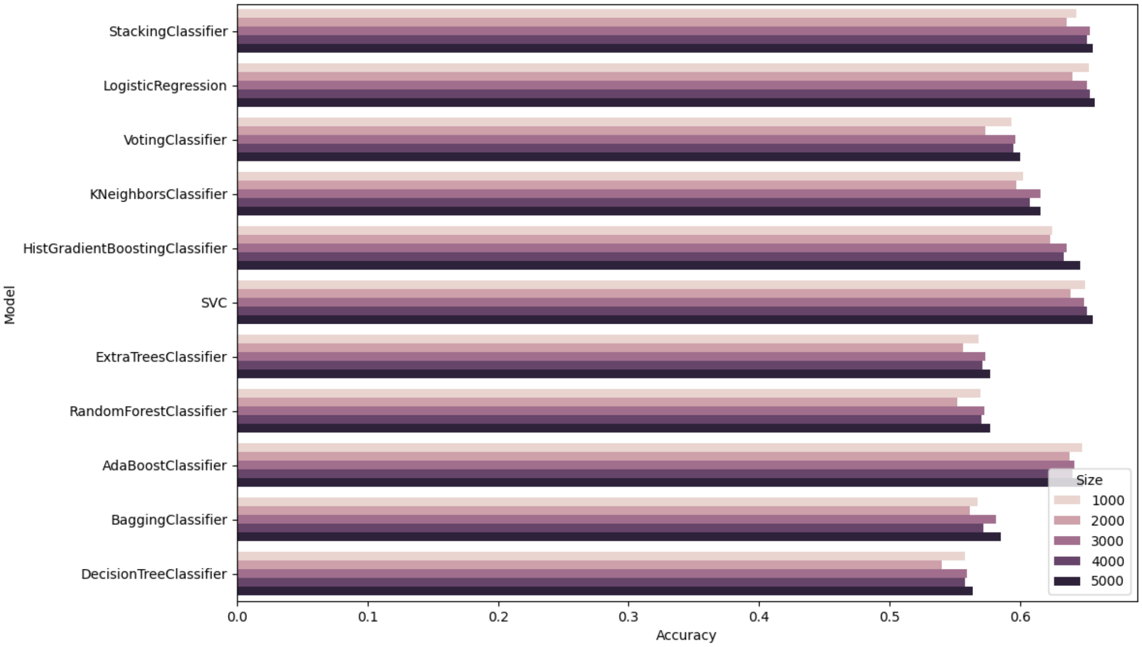


Figure 18 - Accuracy of classification models across different training dataset sizes

4.2.2 Classification of RAG method

In terms of model accuracy, very similar results for the classification of the features extracted with the RAG approach. Nontheless, the accuracy across models for different length of training data shows a clear negative linear relationship.

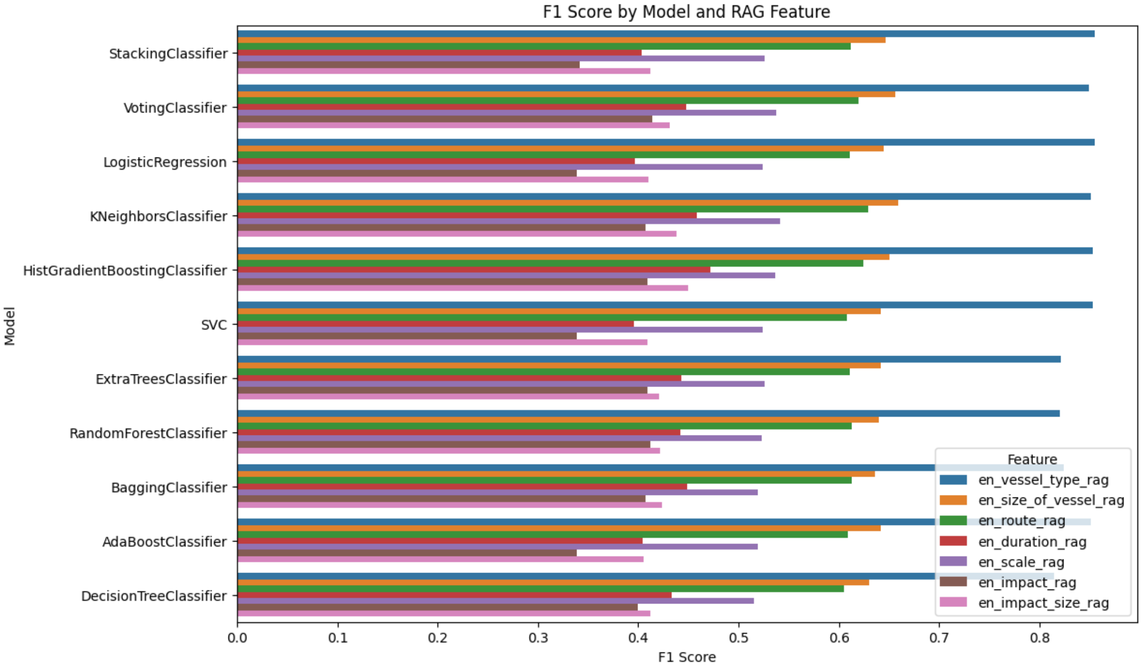


Figure 19 - F1 scores of different classification models evaluated across features

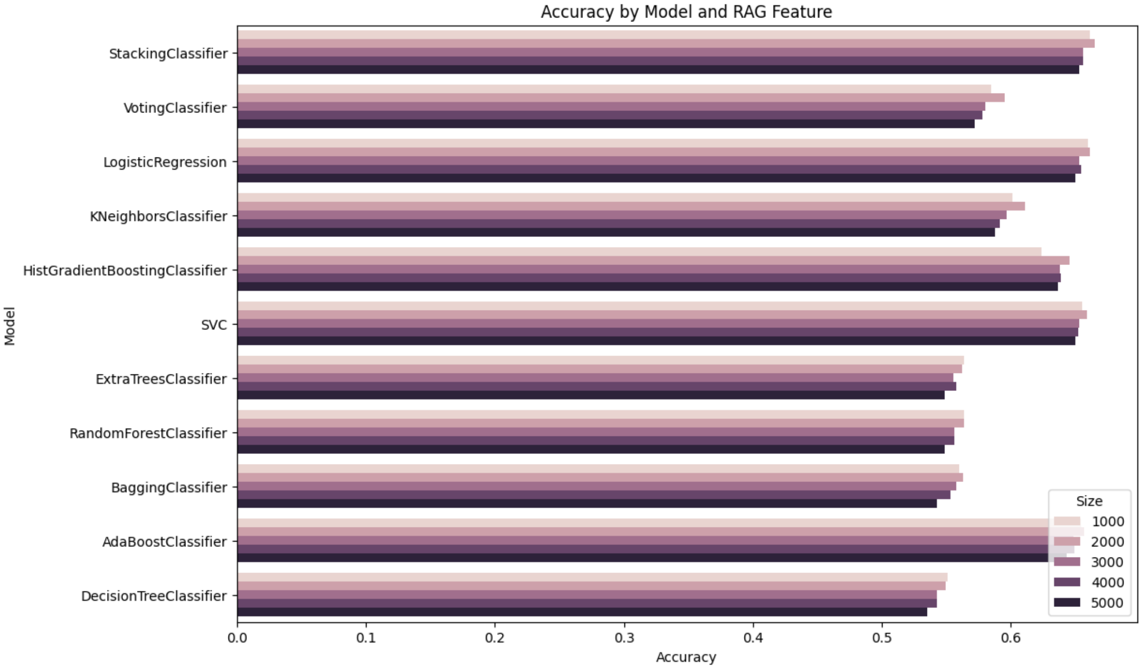


Figure 20 - Accuracy of classification models across different training dataset sizes

4.2.3 Classification of IDFE method

There is a clear decline in predictive ability for two features extracted with the IDFE approach. The “Impact” and “Magnitude” feature have a lower accuracy values compared to the previous two approaches. Similar to the previous two approaches, the Decision Tree, Bagging, Random Forest, and ExtraTrees classification models are performing worse compared to the remaining models.

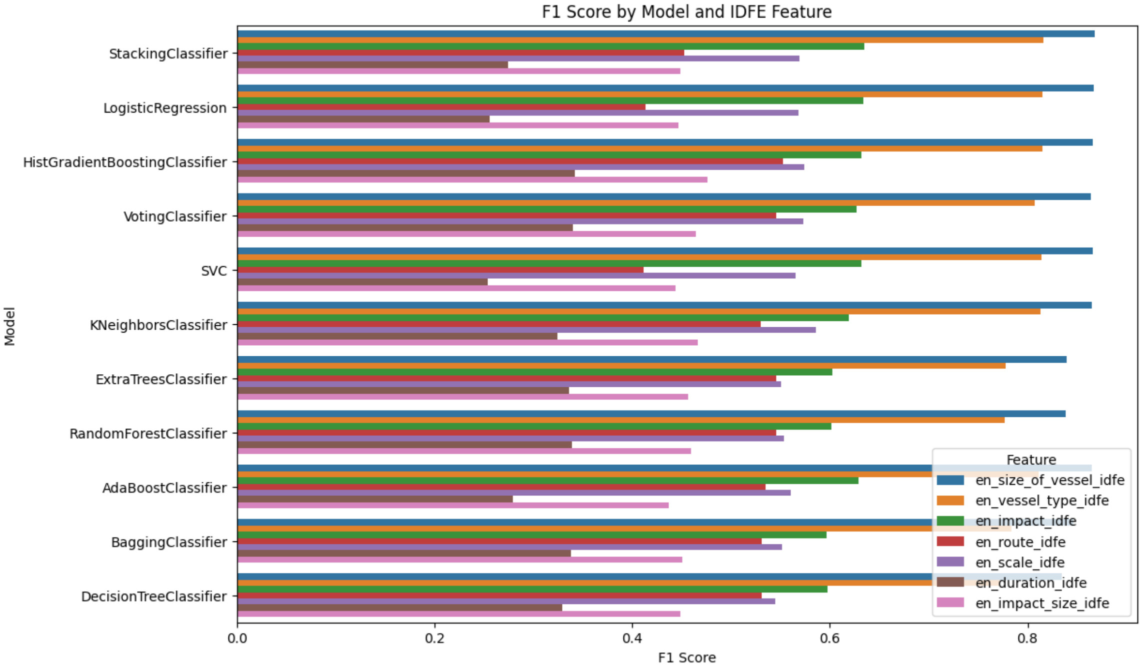


Figure 21 - Accuracy of classification models across different training dataset sizes

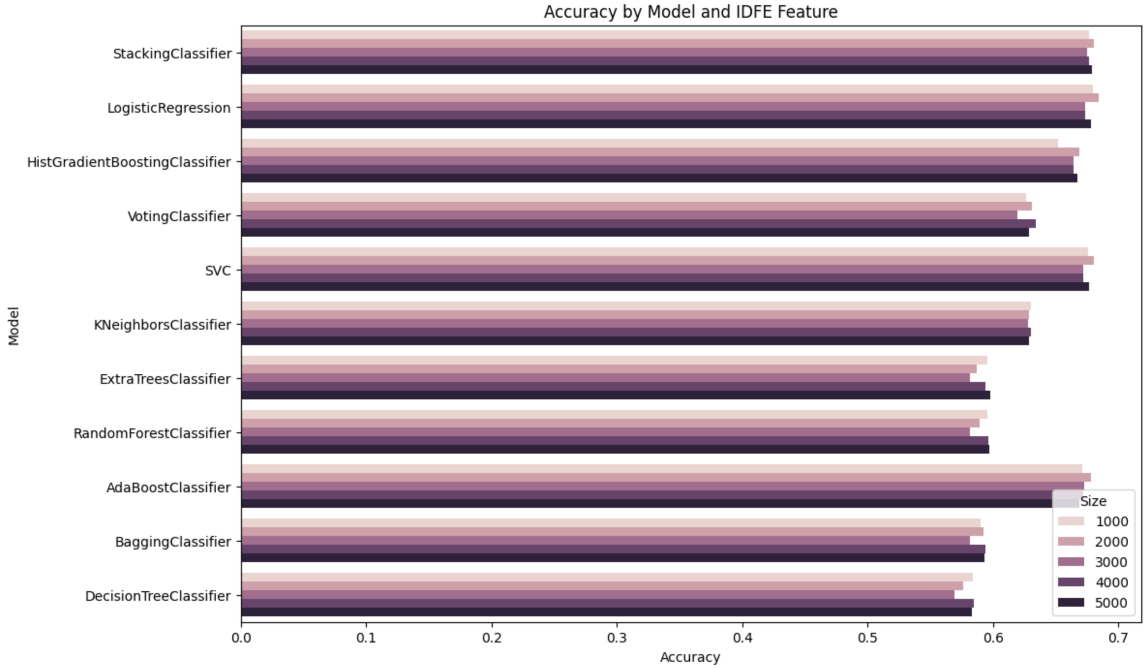


Figure 22 - Accuracy of classification models across different training dataset sizes

4.2.4 Classification of IDRBFE method

The accuracy results for the IDRBFE approach were notably low. All classification models used are experiencing difficulties in accurately predicting the “Duration”, “Impact”, and “Magnitude” features. The performance variability across models indicates potential challenges in capturing the underlying patterns within the IDRBFE features. This suggests that these features may be inherently more complex or less informative for the classification task compared to others. Additionally, the inconsistency in accuracy improvements with larger datasets highlights possible issues with model generalization or feature quality. Further feature engineering or the incorporation of alternative data representations may be necessary to enhance predictive performance for the IDRBFE method.

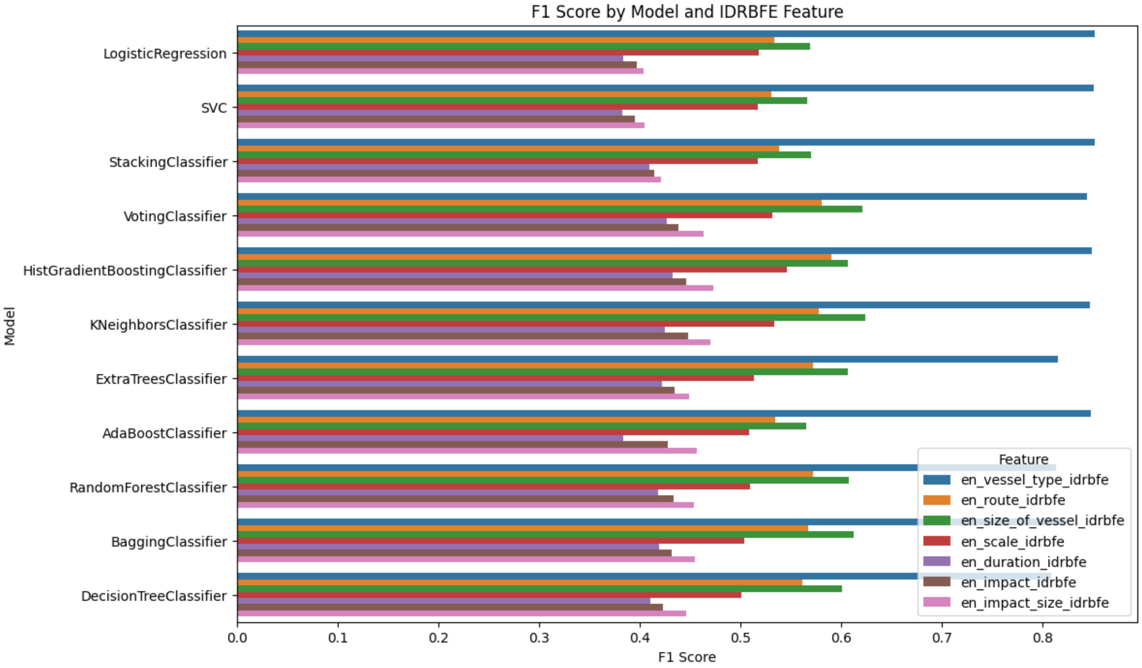


Figure 23 - Accuracy of classification models across different training dataset sizes

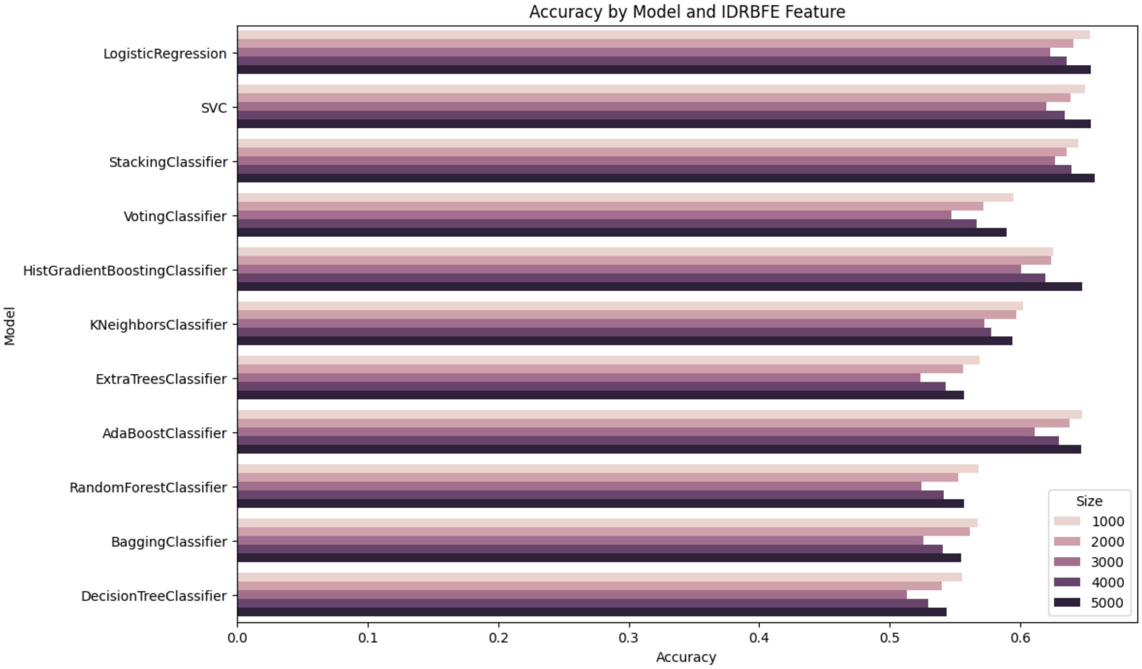


Figure 24 - Accuracy of classification models across different training dataset sizes

4.2.5 Embeddings - Classification results

The results show that the use of the flattened sentence embeddings provide higher predictive ability for all models by a large margin compared to the other methods used.

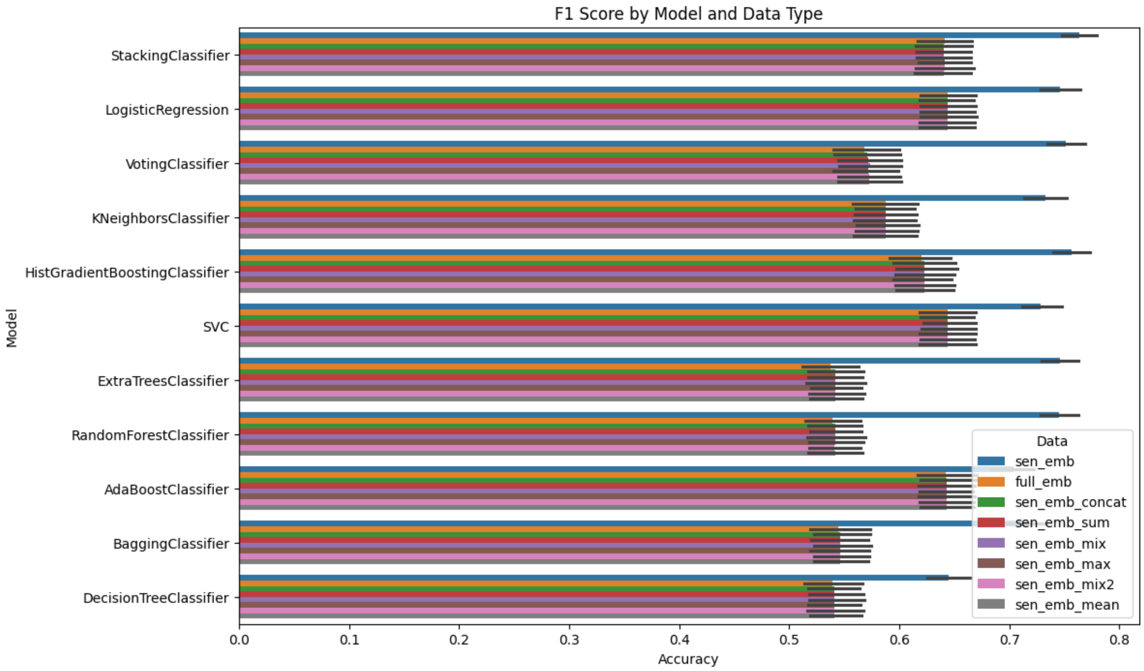


Figure 25 - F1 scores of classification models evaluated across different data types and embedding combinations

4.2.6 Deep Neural Network Classification Results

Similar to traditional machine learning classifiers, the deep neural network (DeepNN) achieves higher accuracy when using flattened embeddings. However, the relationship between training set size and prediction accuracy is not strictly linear, suggesting that increasing data volume does not always guarantee improved performance.

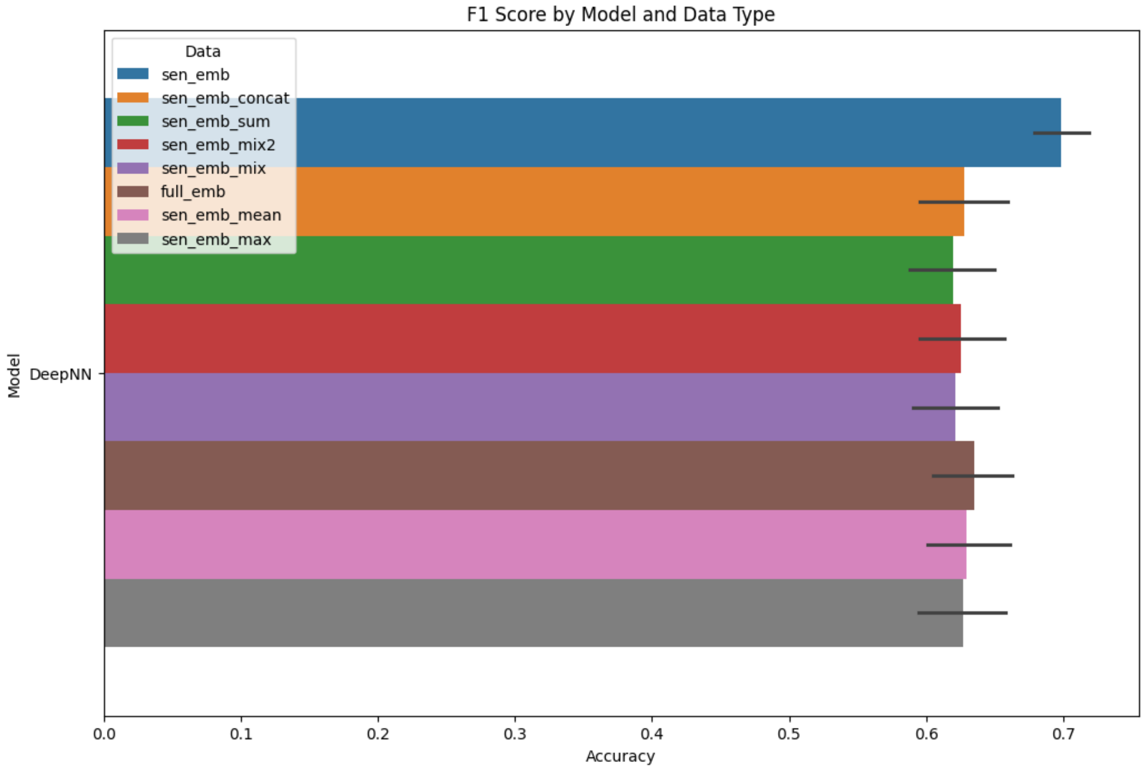


Figure 26 - Accuracy of the Deep Neural Network (DeepNN) model across different data embedding types

Figures 26 and 27 illustrate the DeepNN's accuracy across different embedding types and varying dataset sizes, respectively. Figure 28 further examines accuracy by data embedding types across different training sizes, emphasizing that the model performs best with certain embeddings regardless of dataset size.

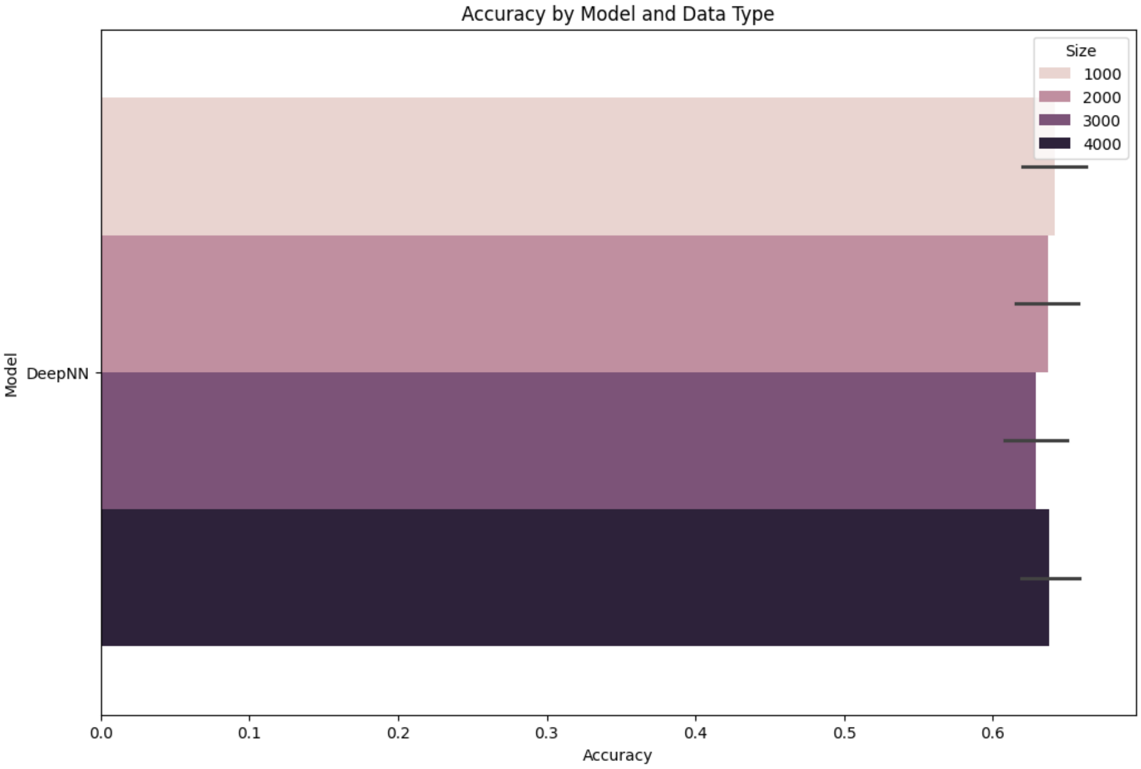


Figure 27 - Accuracy of the Deep Neural Network (DeepNN) model across different training dataset sizes

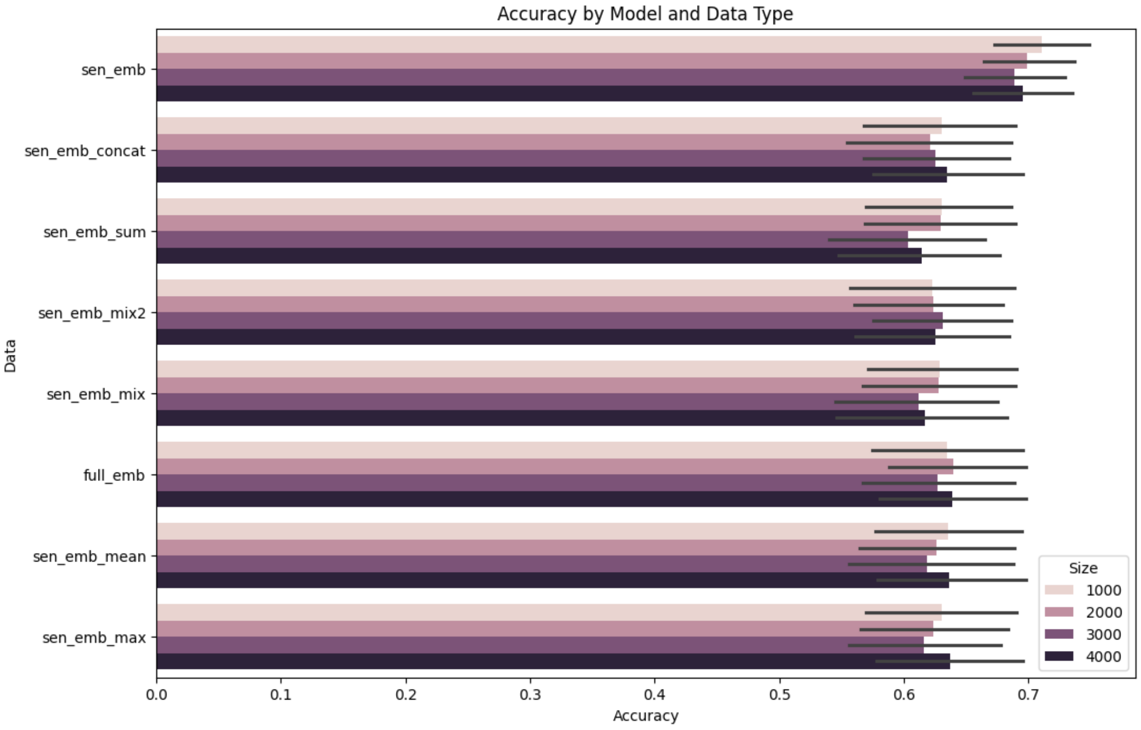


Figure 28 - Accuracy of different data embedding types across various training dataset sizes

The features that are predicted with higher accuracy are again the vessel type. It is interesting to note that the feature "size of vessel" derived from the IDFE method is being predicted with accuracy compared to the features derived from the other methods, which is also the case for the "impact" feature, as shown in the following graph.

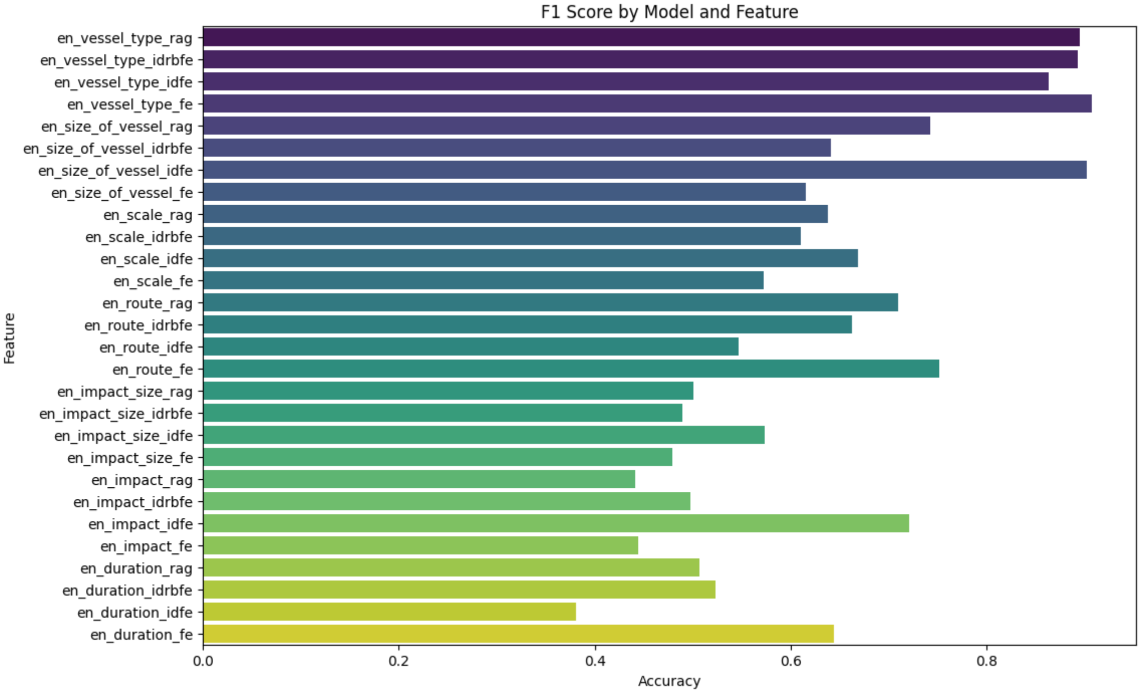


Figure 29 - F1 scores by feature

4.2.7 Particle Swarm Optimization Results

Overall, parameter tuning for DNNs leads to noticeable improvements in predictive performance for most features. Features related to vessel type and size consistently show high F1 scores, indicating their strong predictive power. Conversely, features such as impact size and certain duration metrics exhibit lower performance, but still benefit from optimization. The results highlight the importance of hyperparameter tuning in enhancing model accuracy.

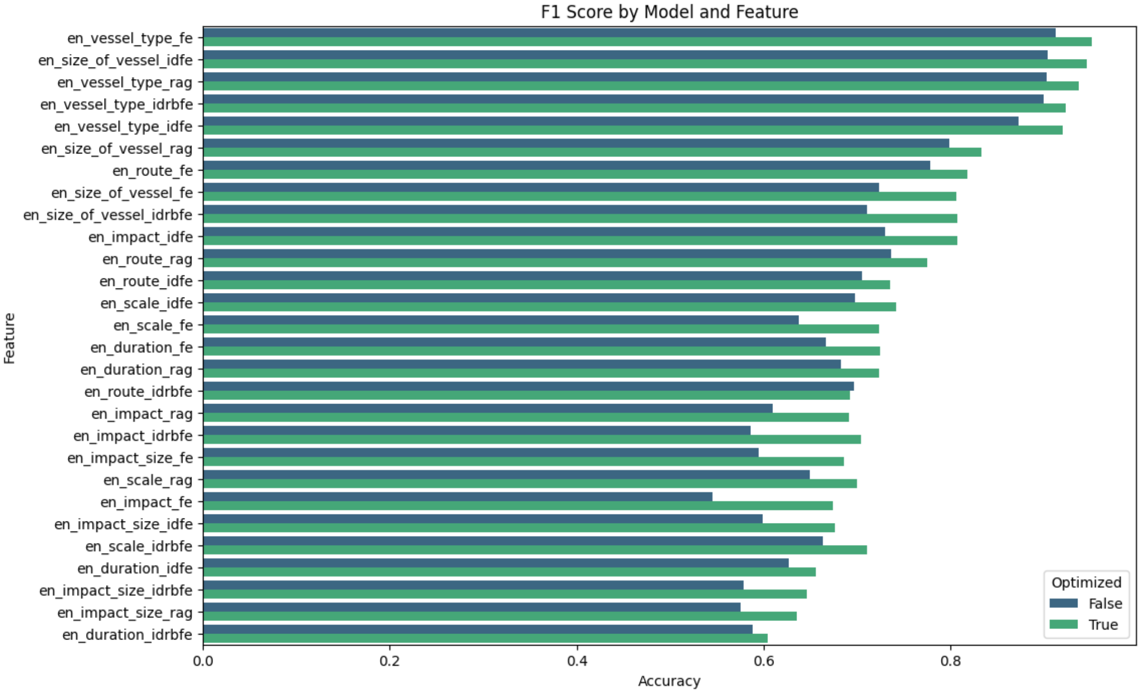


Figure 30 - F1 scores by feature before and after parameter optimization

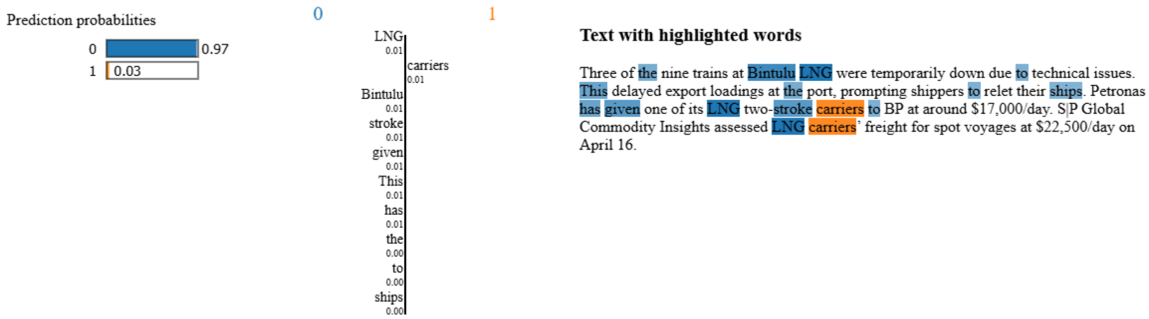
4.3 Classification XAI Results

This section presents explainable AI (XAI) results illustrating model decision-making through correct and incorrect classifications of key features. Figures 31 to 42 show text samples with highlighted words contributing to predictions, enabling a better understanding of model behavior.

- Figures 31, 33, 35, 37, and 39 display correct classifications for features such as vessel type and vessel size, with relevant terms highlighted.

- Figures 32, 34, 36, 38, 40, and 42 illustrate misclassifications for the same features, revealing which words may have led to errors.
- Figure 41 focuses on the correct classification of the “route” feature, while Figure 42 shows its corresponding misclassification.

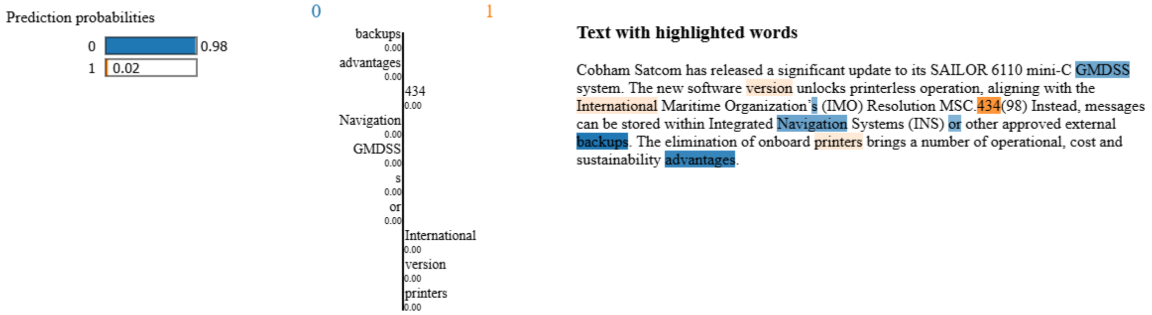
These visualizations provide insight into the linguistic cues the models rely on, enhancing interpretability and trust in the classification process.



Text with highlighted words

Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers’ freight for spot voyages at $22,500/day on April 16.

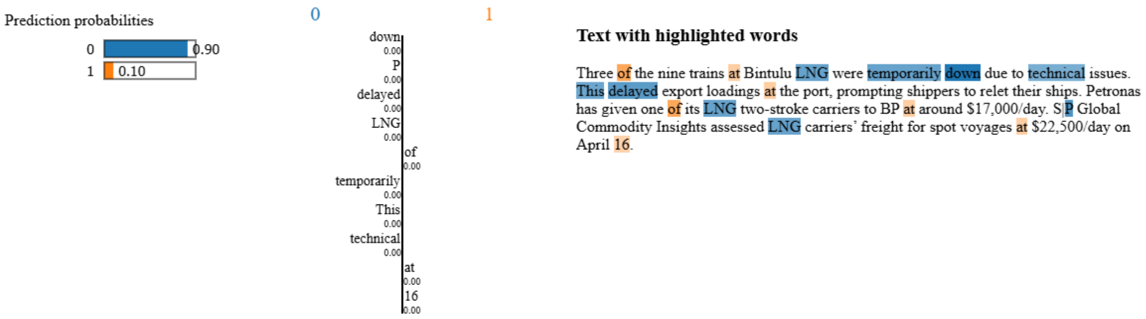
Figure 31 – Correct Classification of feature “vessel type”



Text with highlighted words

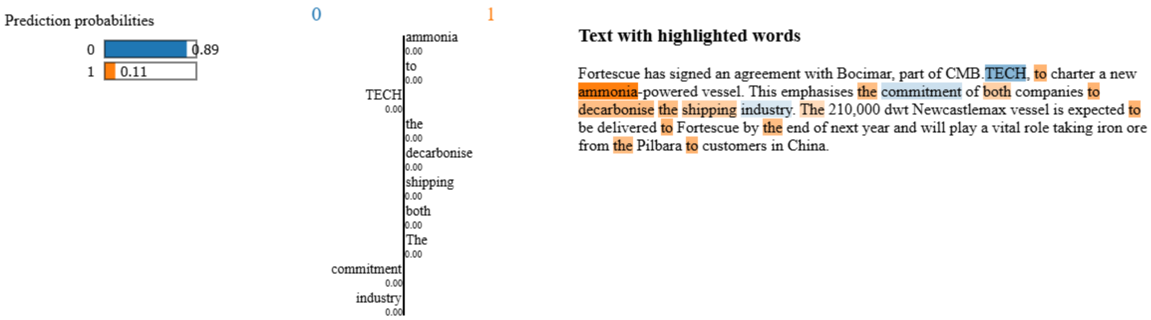
Cobham Satcom has released a significant update to its SAILOR 6110 mini-C GMDSS system. The new software version unlocks printerless operation, aligning with the International Maritime Organization’s (IMO) Resolution MSC.434(98) Instead, messages can be stored within Integrated Navigation Systems (INS) or other approved external backups. The elimination of onboard printers brings a number of operational, cost and sustainability advantages.

Figure 32 – Miss-Classification of feature “vessel type”



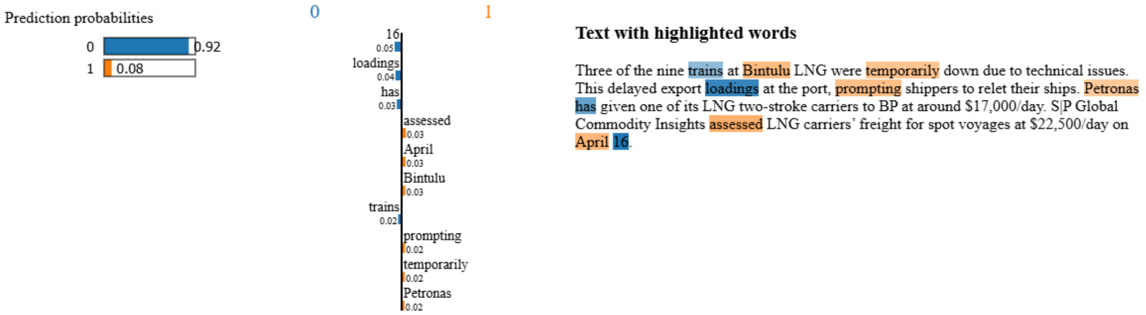
Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers' freight for spot voyages at $22,500/day on April 16.

Figure 33 – Correct Classification of feature "vessel size"



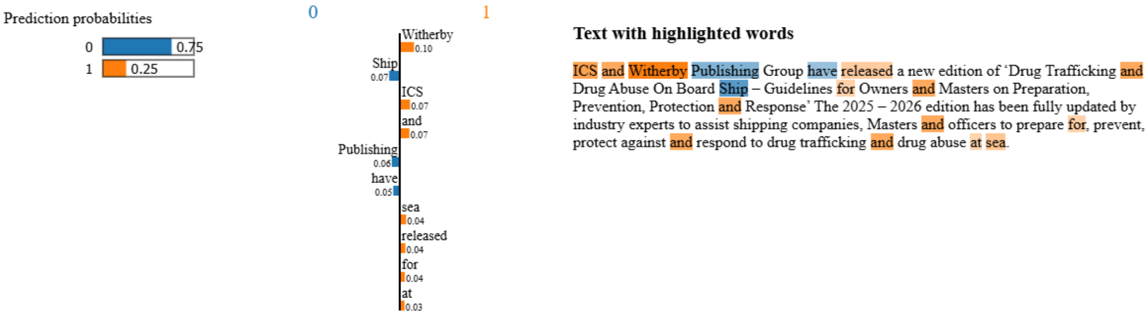
Fortescue has signed an agreement with Bocimar, part of CMB.TECH, to charter a new ammonia-powered vessel. This emphasises the commitment of both companies to decarbonise the shipping industry. The 210,000 dwt Newcastlemax vessel is expected to be delivered to Fortescue by the end of next year and will play a vital role taking iron ore from the Pilbara to customers in China.

Figure 34 – Miss-Classification of feature "vessel size"



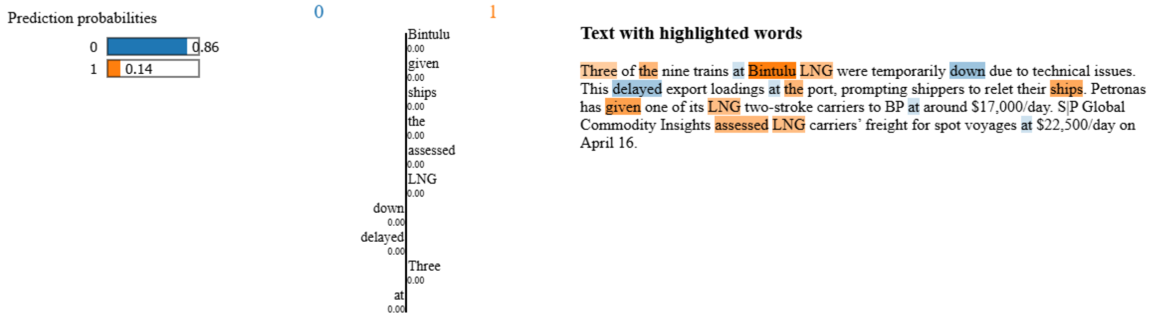
Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers' freight for spot voyages at $22,500/day on April 16.

Figure 35 – Correct Classification of feature "vessel type"



ICS and Witherby Publishing Group have released a new edition of 'Drug Trafficking and Drug Abuse On Board Ship – Guidelines for Owners and Masters on Preparation, Prevention, Protection and Response' The 2025 – 2026 edition has been fully updated by industry experts to assist shipping companies, Masters and officers to prepare for, prevent, protect against and respond to drug trafficking and drug abuse at sea.

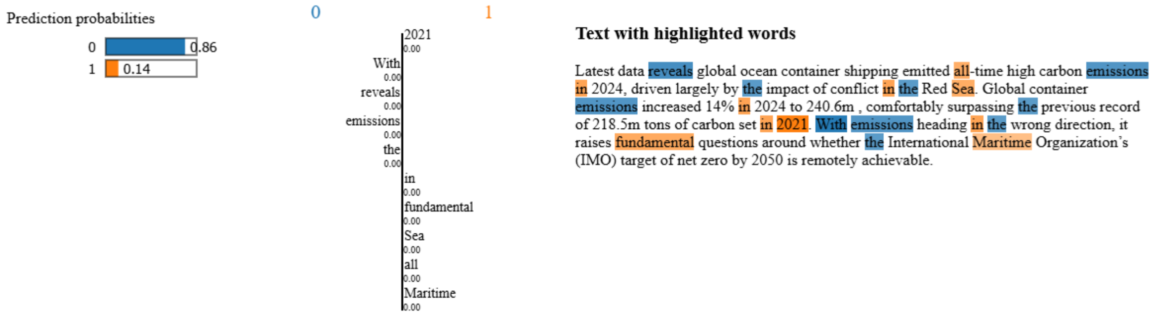
Figure 36 – Miss-Classification of feature "vessel type"



Text with highlighted words

Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers' freight for spot voyages at $22,500/day on April 16.

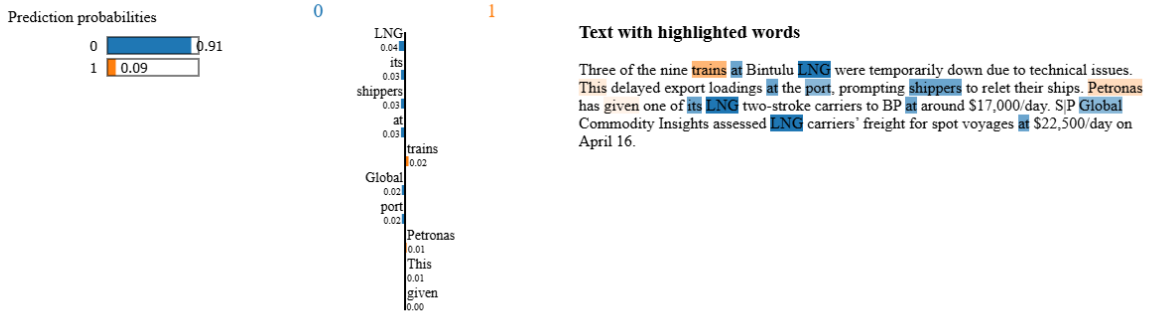
Figure 37 – Correct Classification of feature "vessel type"



Text with highlighted words

Latest data reveals global ocean container shipping emitted all-time high carbon emissions in 2024, driven largely by the impact of conflict in the Red Sea. Global container emissions increased 14% in 2024 to 240.6m , comfortably surpassing the previous record of 218.5m tons of carbon set in 2021. With emissions heading in the wrong direction, it raises fundamental questions around whether the International Maritime Organization's (IMO) target of net zero by 2050 is remotely achievable.

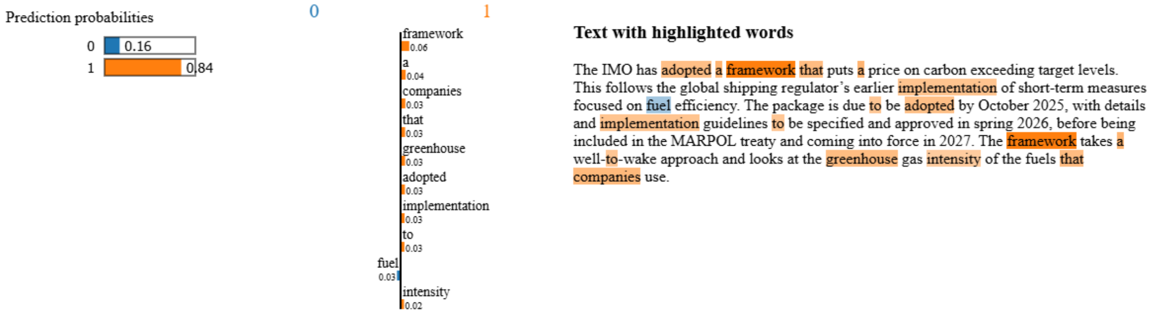
Figure 38 – Miss-Classification of feature "vessel type"



Text with highlighted words

Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers' freight for spot voyages at $22,500/day on April 16.

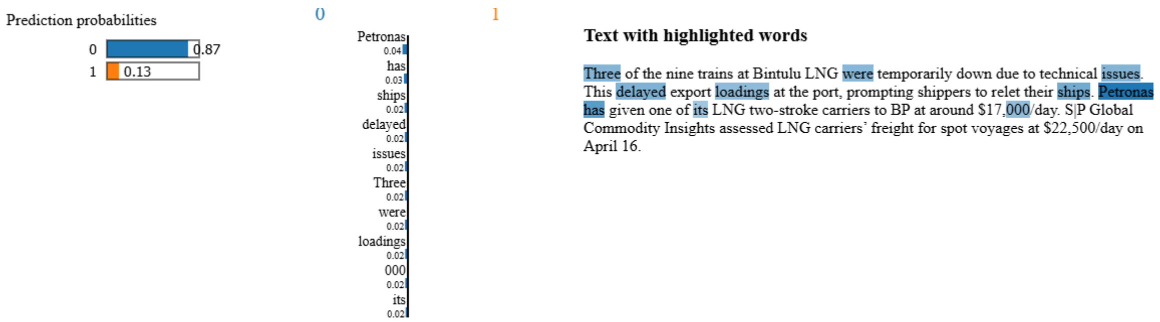
Figure 39 – Correct Classification of feature "vessel size"



Text with highlighted words

The IMO has adopted a framework that puts a price on carbon exceeding target levels. This follows the global shipping regulator's earlier implementation of short-term measures focused on fuel efficiency. The package is due to be adopted by October 2025, with details and implementation guidelines to be specified and approved in spring 2026, before being included in the MARPOL treaty and coming into force in 2027. The framework takes a well-to-wake approach and looks at the greenhouse gas intensity of the fuels that companies use.

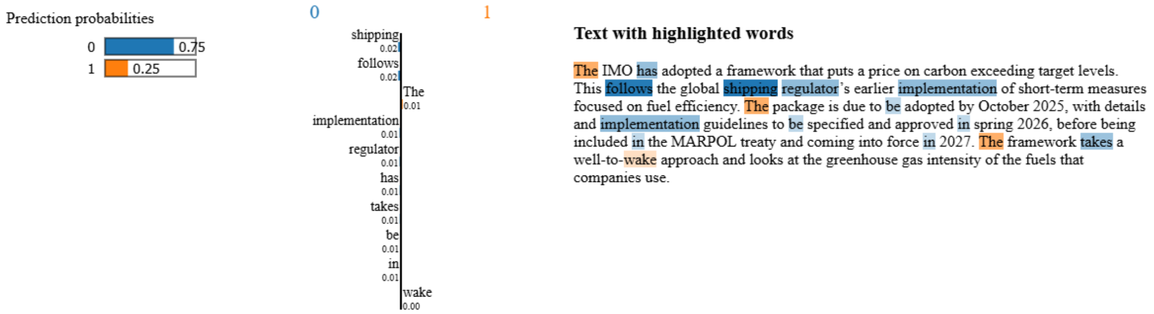
Figure 40 – Miss-Classification of feature "vessel size"



Text with highlighted words

Three of the nine trains at Bintulu LNG were temporarily down due to technical issues. This delayed export loadings at the port, prompting shippers to relet their ships. Petronas has given one of its LNG two-stroke carriers to BP at around $17,000/day. S|P Global Commodity Insights assessed LNG carriers' freight for spot voyages at $22,500/day on April 16.

Figure 41 – Correct Classification of feature "route"



Text with highlighted words

The IMO has adopted a framework that puts a price on carbon exceeding target levels. This follows the global shipping regulator's earlier implementation of short-term measures focused on fuel efficiency. The package is due to be adopted by October 2025, with details and implementation guidelines to be specified and approved in spring 2026, before being included in the MARPOL treaty and coming into force in 2027. The framework takes a well-to-wake approach and looks at the greenhouse gas intensity of the fuels that companies use.

Figure 42 – Miss-Classification of feature "route"

The highlighted words in correctly classified examples often correspond to domain-relevant terms such as vessel types, sizes, and specific maritime locations, indicating that the models effectively capture meaningful contextual information from the textual data.

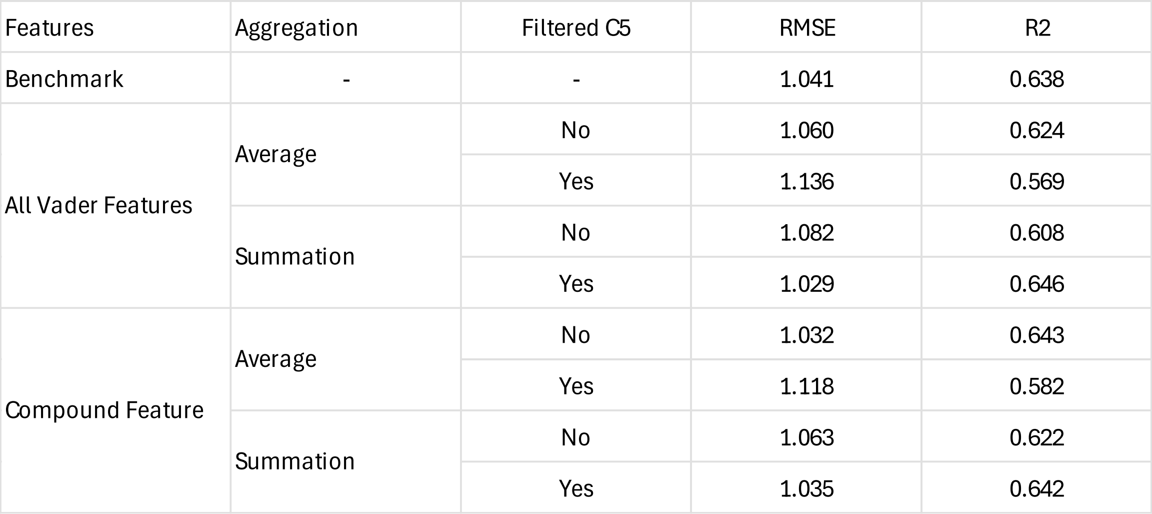
Conversely, the misclassified instances reveal potential areas of ambiguity or confusion, where the models may have overemphasized less relevant words or phrases, or where the language used is inherently vague. This suggests that while the models are generally adept at identifying critical features, certain linguistic nuances or overlapping terminology present challenges.

The XAI results highlight that incorporating domain knowledge and refining the training corpus could further enhance model accuracy and reduce misclassification rates.

4.4 Regression Results

4.4.1 Lexical Sentiment Analysis Features Regression

The lowest root mean squared error (RMSE) was 1.029, achieved by including all VADER features, aggregating them first daily with summation, and then filtering them for the C5 route feature, extracted by the IDRBFE method and predicted with classification. This method resulted in a coefficient of determination (R squared) of 0.646. The difference from the benchmark was 0.012 dollars per metric ton on average prediction.

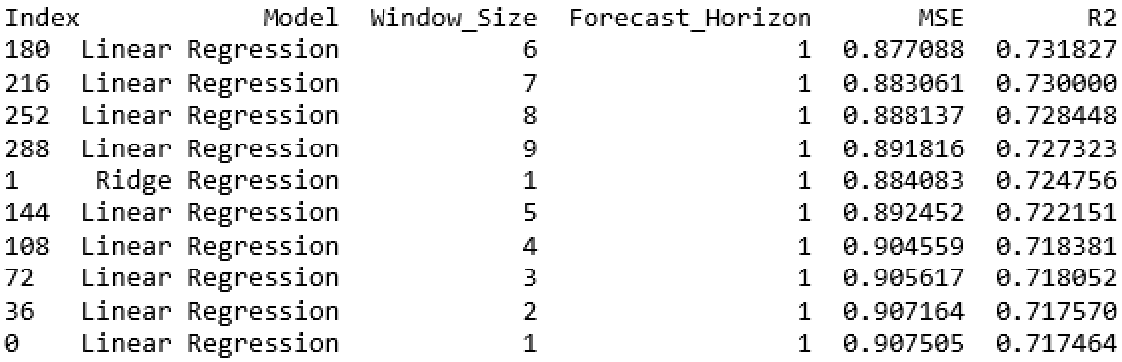


Features	Aggregation	Filtered C5	RMSE	R2
Benchmark	-	-	1.041	0.638
All Vader Features	Average	No	1.060	0.624
		Yes	1.136	0.569
	Summation	No	1.082	0.608
		Yes	1.029	0.646
Compound Feature	Average	No	1.032	0.643
		Yes	1.118	0.582
	Summation	No	1.063	0.622
		Yes	1.035	0.642

Table 4 - Summary of classification model performance metrics

4.4.2 Large Language Model Features Regression

4.4.2.1 Window Size and Forecasting Horizon



```
Index              Model  Window_Size  Forecast_Horizon       MSE        R2
180  Linear Regression            6                 1  0.877088  0.731827
216  Linear Regression            7                 1  0.883061  0.730000
252  Linear Regression            8                 1  0.888137  0.728448
288  Linear Regression            9                 1  0.891816  0.727323
1     Ridge Regression            1                 1  0.884083  0.724756
144  Linear Regression            5                 1  0.892452  0.722151
108  Linear Regression            4                 1  0.904559  0.718381
72   Linear Regression            3                 1  0.905617  0.718052
36   Linear Regression            2                 1  0.907164  0.717570
0    Linear Regression            1                 1  0.907505  0.717464
```

Table 5 - Performance comparison of regression models with varying window sizes and forecast horizons

The optimal combination of window size for the training data and forecasting horizon are 6 weeks and 1 week. As expected, the models are more accurate when forecasting for only 1 week ahead. It is interesting to note that there is a cut-off period that makes using the impact of news articles beneficial for predictions. Different combinations of window size and forecasting horizon with and without the impact feature result in mixed accuracy results.

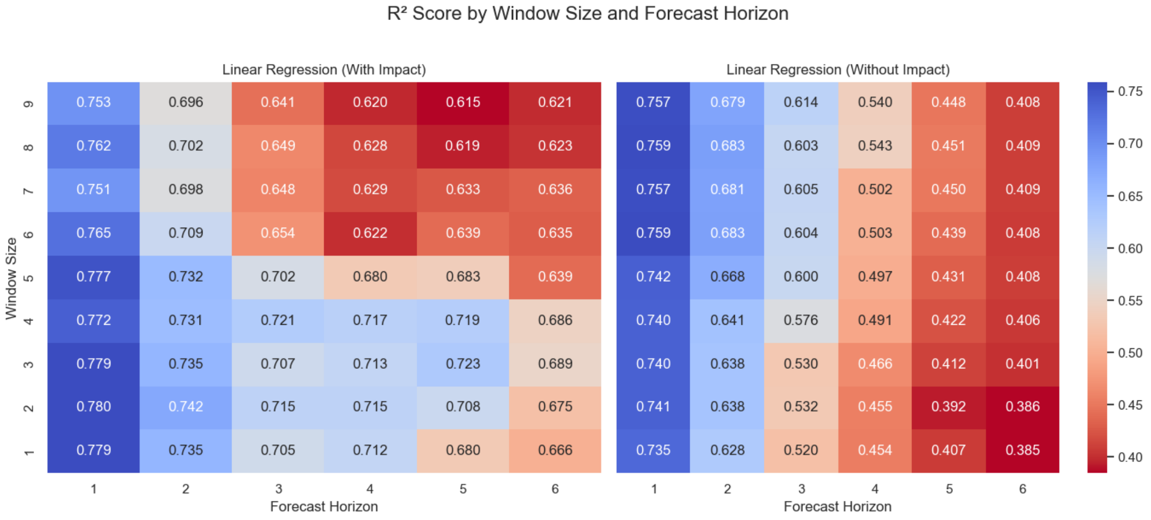


Figure 43 – Heatmap of R² scores for linear regression models with and without the impact feature

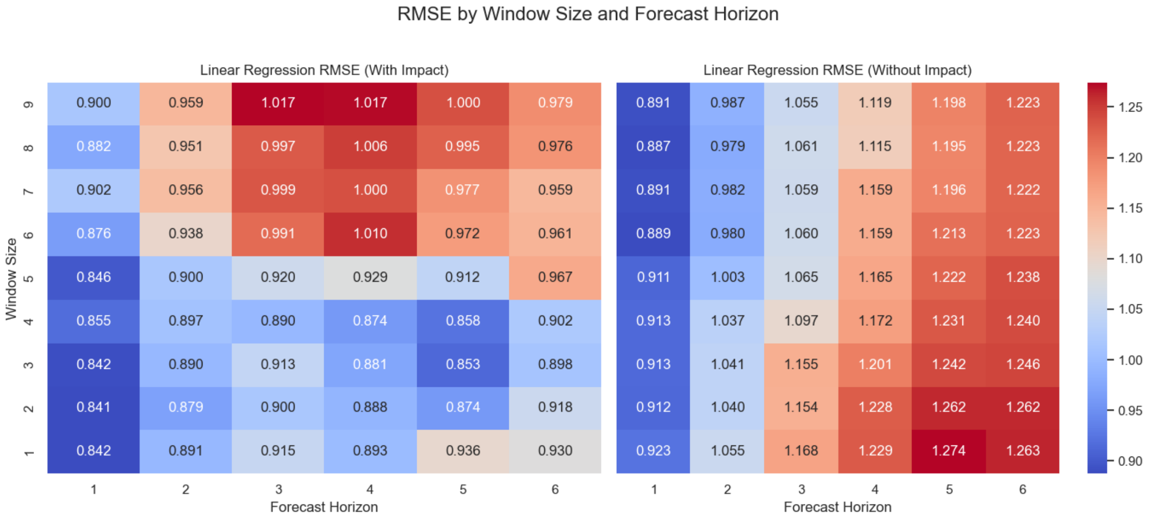


Figure 44 – Heatmap of RMSE for linear regression models with and without the impact feature

A model with a window size greater than 5 weeks makes better predictions with the impact feature included, for any forecast horizon. Moreove, any forecast for 5 weeks horizon is more accurate if the impact feature is included.

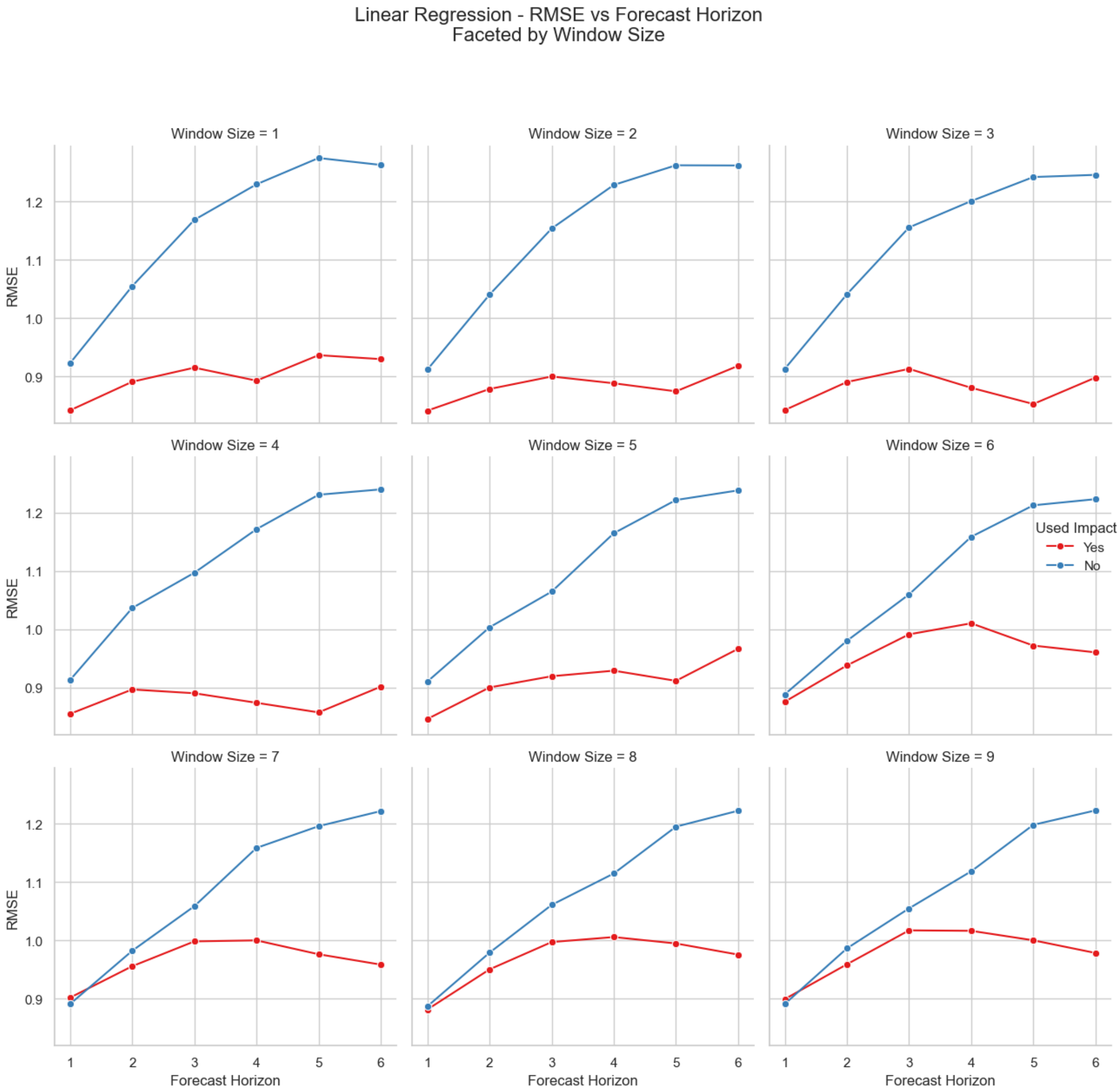


Figure 45 - RMSE of linear regression models across different forecast horizons and window sizes

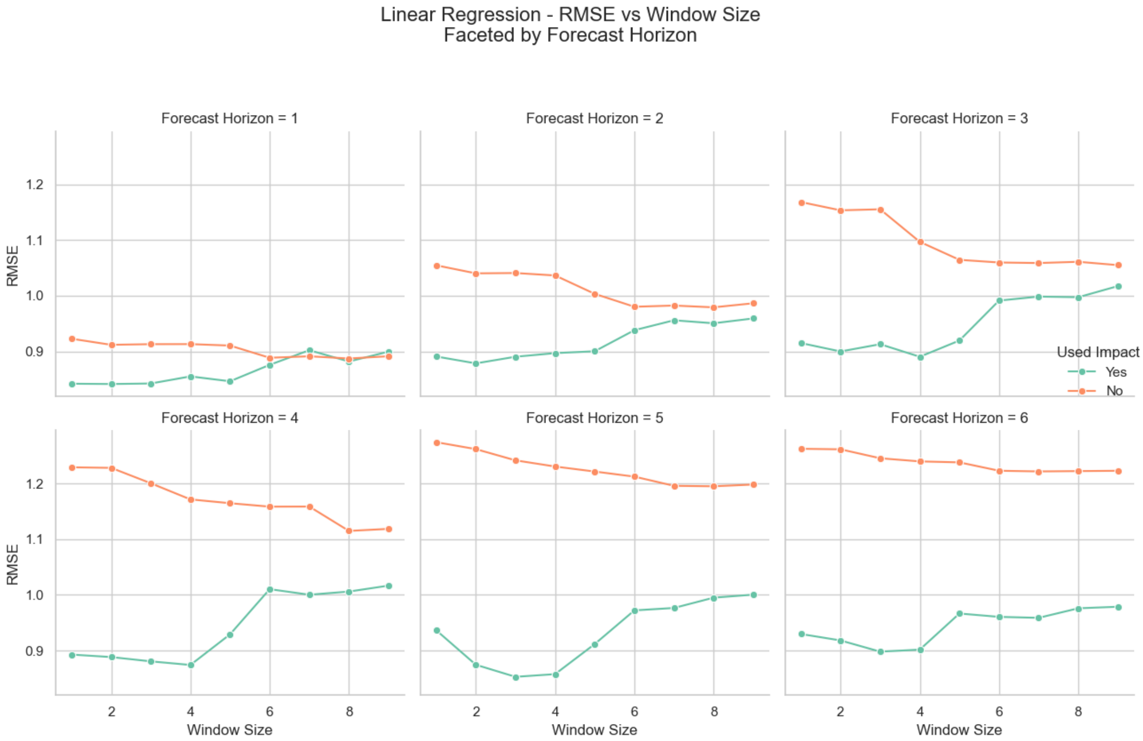


Figure 46 - RMSE of linear regression models across different forecast horizons and window sizes

4.4.2.2 Regression Results – 5 year period

The regression analysis over the five-year period reveals consistent improvements when incorporating impact features across all four feature extraction methods (FE, RAG, IDFE, and IDRBFE) without their lagged values. Models that utilized impact variables generally achieved lower Root Mean Squared Error (RMSE) values and higher R² scores, indicating enhanced predictive accuracy and better explanation of variance in freight rates. Among the methods, the FE and IDFE approaches demonstrated particularly strong performance, with notable reductions in RMSE and corresponding increases in R² compared to models without impact features. These results emphasize the value of including impact-related features in forecasting freight rates, as they contribute in the explanatory power and improve overall model fit.

Feature	Filter	Lagged	RMSE	R2	Model
Benchmark			0.980	0.682	Linear
FE	C5	Yes	0.910	0.726	SVR
		No	0.854	0.758	SVR
	-	Yes	1.440	0.314	LGBM
		No	1.303	0.438	LGBM
RAG	C5	Yes	0.898	0.733	Linear
		No	0.959	0.696	Linear
	-	Yes	1.135	0.574	Linear
		No	1.138	0.572	Linear
IDFE	C5	Yes	0.745	0.816	Ridge
		No	0.925	0.717	Ridge
	-	Yes	0.938	0.709	Linear
		No	0.967	0.690	Linear
IDRBFE	C5	Yes	1.070	0.618	SVR
		No	0.990	0.676	Gradient Boosting
	-	Yes	1.117	0.588	XGB
		No	1.104	0.597	Linear

Table 6 – Parameters and regression results

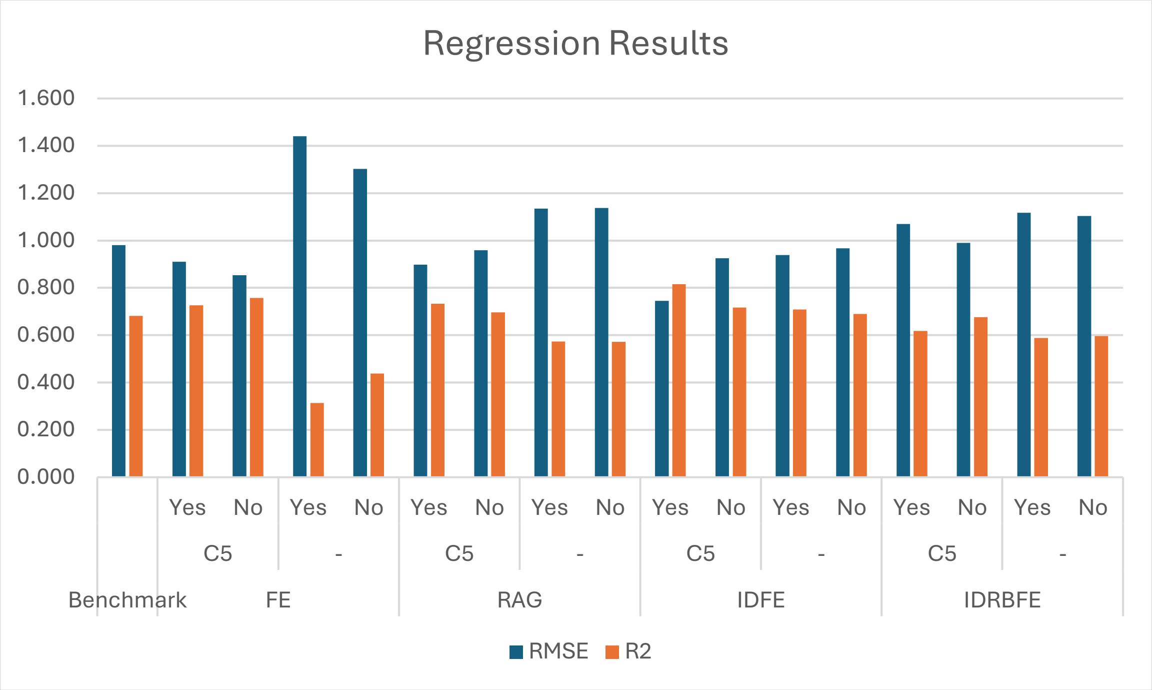
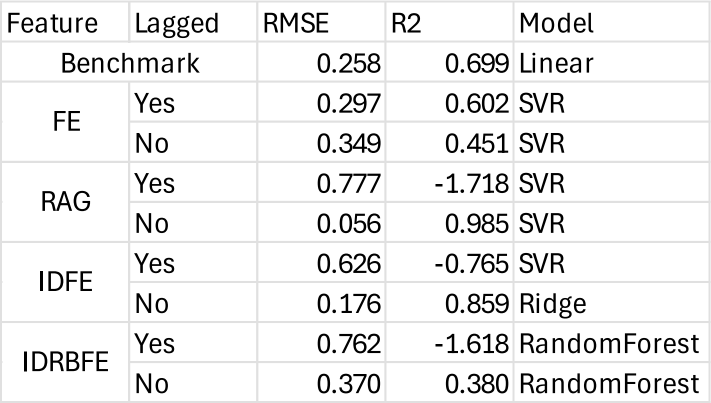


Figure 47 – Parameters and regression results

4.4.2.2 Regression Results – LLM-only features

The regression results, when using only the features extracted with the LLM, demonstrate that incorporating lagged values ("Yes") generally reduces model performance compared to models without lagged inputs ("No"). This is evident from the consistently lower RMSE and higher R² scores when lagged values are not included across all feature extraction methods. Notably, the RAG method benefits most from lagging, yielding the best predictive accuracy. In contrast, other methods such as FE and IDFE show more variable results, with some negative R² values indicating poorer fits with lagged inputs. These outcomes underscore the importance of omitting lagged variables, while also highlighting differences in the effectiveness of various feature extraction techniques.



Feature	Lagged	RMSE	R2	Model
Benchmark		0.258	0.699	Linear
FE	Yes	0.297	0.602	SVR
	No	0.349	0.451	SVR
RAG	Yes	0.777	-1.718	SVR
	No	0.056	0.985	SVR
IDFE	Yes	0.626	-0.765	SVR
	No	0.176	0.859	Ridge
IDRBFE	Yes	0.762	-1.618	RandomForest
	No	0.370	0.380	RandomForest

Table 7 – Parameters and regression results

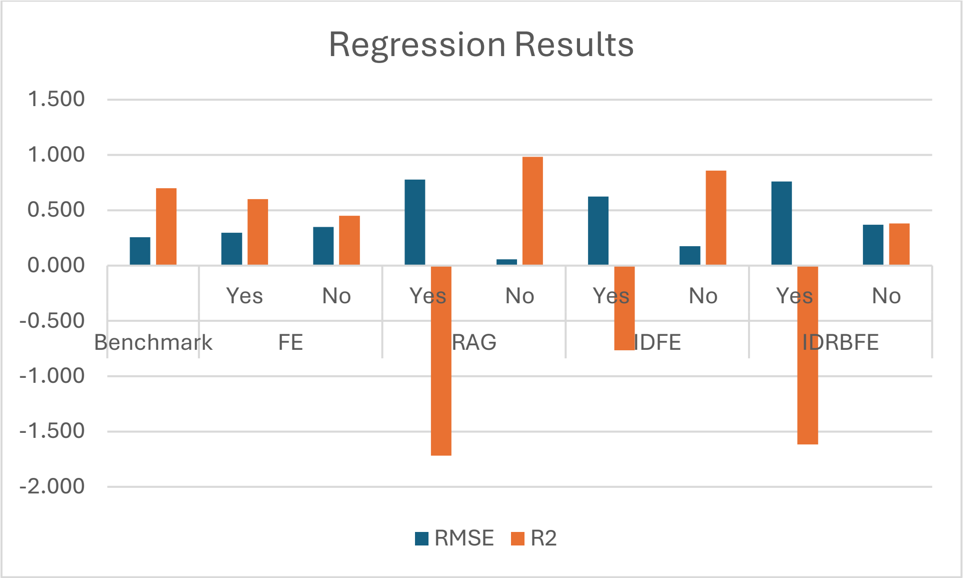


Figure 48 - Parameters and regression results

4.5 Causality Results

4.5.1 Granger Causality

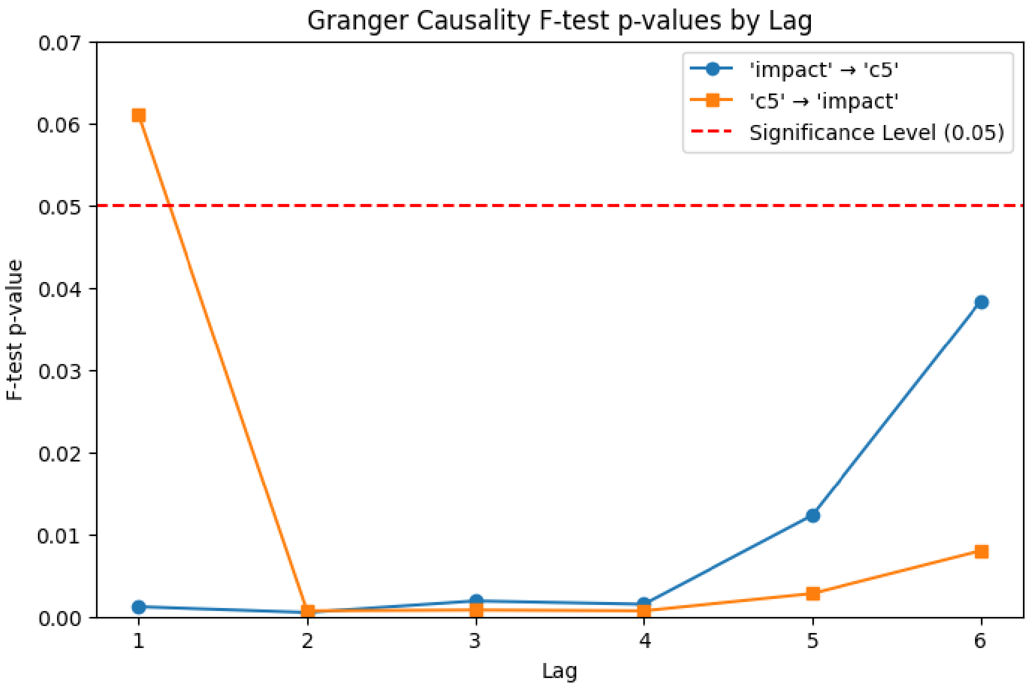


Figure 49 - Granger causality F-test p-values for the relationship between the 'impact' feature and the C5 freight rate across different lags

There is bidirectional Granger causality between impact and c5, but the effect of impact on c5 is stronger and more consistent across lags. This supports the idea that the impact variable contains predictive information for c5, but some feedback from c5 to impact may also exist.

Granger causality Wald-test. H_0: impact does not Granger-cause c5. Conclusion: reject H_0 at 5% significance level.

Test statistic: 15.28

Critical value: 7.815

p-value: 0.002

df: 3

Granger causality Wald-test. H_0: c5 does not Granger-cause impact. Conclusion: reject H_0 at 5% significance level.

Test statistic: 17.32

Critical value: 7.815

p-value: 0.001

df: 3

There is statistically significant, bidirectional predictive causality between the positive impact feature and C5 rates. This means past values of each series help predict the other, confirming that impact features are not only correlated with, but also predictive of, C5 rate changes.

4.5.2 Transfer Entropy

The transfer entropy from c5 to impact (0.1024) is slightly higher than from impact to c5 (0.0921).

This suggests that knowing the past values of c5 provides a bit more information about the future of events with a positive impact than the other way around. However, the difference is small, indicating that the information flow is relatively balanced and there is no strong unidirectional influence. Both series influence each other, but the effect of c5 on impact is marginally stronger than the reverse. This is consistent with a feedback relationship or mutual dependence, rather than a clear cause and response dynamic.

4.5.3 Multicollinearity

Feature	VIF
C5	12.551
Positive impact	5.995
Negative impact	7.646

The VIF (Variance Inflation Factor) results show that c5 has severe multicollinearity with the other features and the positive impact and negative impact have moderate to high multicollinearity.

This means that the predictors are highly correlated. This can make regression coefficients unstable and unreliable, and may inflate standard errors. In practice, it means it will be difficult to interpret the individual effect of each variable.

5. Discussion of Findings

The forecasting results show a clear indication that feature extraction from news articles with the use of LLM prompting has merits in time-series modeling. The accuracy improvement is not great enough to make this practice essential. The small improvement could be due to the lack of sophistication of LLMs. However, if the LLMs are complex enough, then the low performance could be due to other oversights in this methodology. The key-take away from this process is that prompting LLMs can assist in creating a sentiment index, which could complement a regression model for time-series data. Additionally, the methodologies applied in this study could be a substitute for fine-tuning of the LLMs, a process that is costly and time-consuming.

5.1 Classification

The purpose of using models for classification was to supplement the data points acquired from the use of the LLM. Some features were successfully modeled. However, every classification model, optimized deep neural networks included, underperformed in capturing the nuance of the more complex features, such as the impact and the magnitude of a news article. It is also important to note that classic models for classification outperformed the deep neural networks across all features. That can be attributed to the relatively short data set of 5,000 points.

The XAI highlighted the issue of nuanced features that can hardly be captured through embeddings and machine learning models. It appeared that in many cases, the models were capturing the wrong meaning from the text. A more complex vectorization of the textual data could have been employed to avoid such miss-classifications.

5.2 Regression

There was a slight improvement of accuracy for some feature-extracting methods compared to the benchmark. It is interesting to note that the Lexical Sentiment Analysis improved the forecasting only when the C5 filtering, from the FE method, was applied.

The results also show that for larger forecasting horizons the benchmarks are severely outperformed from the enhanced models.

5.3 Causality

The results of this study clearly show that there is a causal relationship between the features derived from the articles and the variations in the C5 route freight rates. This indicates that changes in the textual features, such as sentiment and event characteristics extracted from maritime news, can predict movements in freight rates on the C5 route. The use of advanced causality testing methods, including Toda-Yamamoto Granger causality, Transfer Entropy, and Convergent Cross Mapping, strengthens the evidence supporting this directional influence.

In some cases, a reversed causality effect is also observed, suggesting that changes in freight rates may, at times, influence the nature or tone of maritime news coverage. This bidirectional relationship highlights the complex interplay between market dynamics and information flow, where freight rate fluctuations can drive media reporting, which in turn impacts market expectations and behavior. Understanding this feedback loop is essential for developing more robust predictive models and for interpreting the causality results within the broader context of maritime economics and market sentiment analysis.

And it is this demonstrated causality that validates the merit of the proposed methodology. By establishing a clear directional influence between news-derived features and freight rate variations, the approach proves its effectiveness in capturing meaningful market signals. This causal linkage supports the use of textual data and advanced feature extraction techniques as valuable components in forecasting models, reinforcing the methodology’s relevance and potential for practical application in maritime freight rate prediction.

6. Conclusion and Implications

In conclusion, this study demonstrates that feature extraction from maritime news articles using Large Language Model (LLM) prompting holds promise for enhancing time-series forecasting models. While the accuracy improvements observed were modest and not sufficient to deem the practice indispensable, the results highlight the potential value of sentiment indices derived from LLMs as complementary inputs to regression models. Moreover, the methodologies developed here offer a practical alternative to costly and time-intensive LLM fine-tuning, making them an accessible option for incorporating textual data into forecasting frameworks.

The classification results further underscore the challenges of modeling complex features such as impact and magnitude within news articles. Despite employing a variety of models including optimized deep neural networks, capturing the nuanced meaning embedded in textual data remains difficult, especially with relatively limited data sizes. Traditional machine learning classifiers performed better than deep learning approaches, reflecting the constraints imposed by dataset size and the current vectorization techniques. The explainable AI analyses revealed misclassifications linked to insufficient semantic understanding, suggesting that more sophisticated text representation methods may be required to fully exploit the richness of news data.

Finally, the study's robust causality analyses provide strong evidence of a directional relationship between news-derived features and freight rate fluctuations on the C5 route. This finding validates the core methodology and reinforces the importance of integrating textual data into predictive models. The observed bidirectional causality further emphasizes the dynamic interplay between market events and media coverage, which should be considered in future forecasting efforts. Overall, this research lays the groundwork for more effective use of qualitative news information in maritime economic modeling, paving the way for improved decision-making and market insight.

III. References

Angrist, J. D., & Pischke, J.-S. (2009). *Mostly harmless econometrics: An empiricist's companion*. Princeton University Press.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. *LREC*, 10, 2200–2204.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics, 31*(3), 307–327. https://doi.org/10.1016/0304-4076(86)90063-1

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123–140. https://doi.org/10.1007/BF00058655

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

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). https://doi.org/10.1145/2939672.2939785

Chatham House. (2017). Chokepoints and vulnerabilities in global food trade. Retrieved from https://www.chathamhouse.org/2017/06/chokepoints-and-vulnerabilities-global-food-trade-0/2-chokepoints-global-food-trade

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

DeepSeek, Inc. (2024). *DeepSeek-V3-0324* [Large Language Model]. Retrieved July 16, 2025, from https://www.deepseek.com/models/deepseek-v3-0324

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 4171–4186). https://doi.org/10.18653/v1/N19-1423

Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A., & Vapnik, V. (1997). Support vector regression machines. In *Advances in Neural Information Processing Systems* (Vol. 9, pp. 155–161).

Eisenstein, J. (2019). *Introduction to natural language processing*. MIT Press.

Food and Agriculture Organization of the United Nations. (n.d.). FAO trade expert networks. Retrieved July 16, 2025, from https://www.fao.org/markets-and-trade/areas-of-work/trade-policy-and-partnerships/fao-trade-expert-networks/en

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning, 63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (2nd ed.). O'Reilly Media.

Google DeepMind. (2023). *Gemini: A family of highly capable multimodal models*. Retrieved July 16, 2025, from https://deepmind.google/models/gemini/

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

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55–67. https://doi.org/10.1080/00401706.1970.10488634

Hoffmann, J., Wilmsmeier, G., & Lun, Y. H. V. (2017). Connecting the world through global shipping networks. *Journal of Shipping and Trade, 2*(1). https://doi.org/10.1186/s41072-017-0020-z

Hugging Face. (2023). *huggingface_hub: Client library to interact with the Hugging Face Hub (Version X.X)* [Computer software]. GitHub. https://github.com/huggingface/huggingface_hub

Hugging Face. (n.d.). *all-MiniLM-L6-v2*. Retrieved July 16, 2025, from https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Hugging Face. (n.d.). *facebook/bart-large-cnn*. Retrieved July 16, 2025, from https://huggingface.co/facebook/bart-large-cnn

Hutto, C. J., & Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International AAAI Conference on Weblogs and Social Media*.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning: With applications in R*. Springer.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks* (Vol. 4, pp. 1942–1948). https://doi.org/10.1109/ICNN.1995.488968

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems* (Vol. 30, pp. 3146–3154).

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1412.6980

Lhoest, Q., Vlastelica, M., Raza, S., Dron, M., Savkov, A., Gautier, J., & Wolf, T. (2021). Datasets: A community library for efficient dataset sharing and processing [Computer software]. Hugging Face. https://github.com/huggingface/datasets

Loria, S. (2018). TextBlob: Simplified text processing. Retrieved from https://textblob.readthedocs.io/en/dev/

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 4765–4774). https://doi.org/10.1109/ICMLA.2017.00-12

Meta AI. (2023). *LLaMA language models*. Retrieved July 16, 2025, from https://ai.facebook.com/blog/large-language-model-llama-meta-ai/

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to linear regression analysis* (5th ed.). Wiley.

Microsoft. (2023). *Azure AI services*. Retrieved July 16, 2025, from https://azure.microsoft.com/en-us/products/ai-services

Miranda, D. F., Fonseca, F. A., Coelho, L. d. S., & Amaro Júnior, E. (2020). pyswarms: A research toolkit for particle swarm optimization in Python. *Journal of Open Source Software, 5*(53), 2334. https://doi.org/10.21105/joss.02334

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big Things Come in Small Packages* (pp. 93–98).

Notteboom, T., & Rodrigue, J.-P. (2021). *The geography of transport systems* (5th ed.). Routledge.

O'Brien, R. M. (2007). A caution regarding rules of thumb for variance inflation factors. *Quality & Quantity, 41*(5), 673–690. https://doi.org/10.1007/s11135-006-9018-6

Park, J. S., Seo, Y.-J., & Ha, M.-H. (2019). The role of maritime, land, and air transportation in economic growth: Panel evidence from OECD and non-OECD countries. *Research in Transportation Economics, 78*, 100765. https://doi.org/10.1016/j.retrec.2019.100765

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (Vol. 32, pp. 8026–8037). https://arxiv.org/abs/1912.01703

Radelet, S., & Sachs, J. (1998). Shipping costs, manufactured exports and economic growth. Mimeo. American Economic Association annual meeting.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (pp. 3982–3992). https://doi.org/10.18653/v1/D19-1410

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144). https://doi.org/10.1145/2939672.2939778

Runge, J., Nowack, P., Kretschmer, M., Flaxman, S., & Sejdinovic, D. (2019). Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances, 5*(11), eaau4996. https://doi.org/10.1126/sciadv.aau4996

Schreiber, T. (2000). Measuring information transfer. *Physical Review Letters, 85*(2), 461–464. https://doi.org/10.1103/PhysRevLett.85.461

Shiller, R. J. (2017). Narrative economics. *American Economic Review, 107*(4), 967–1004. https://doi.org/10.1257/aer.107.4.967

Sugihara, G., May, R., Ye, H., Hsieh, C.-H., Deyle, E., Fogarty, M., & Munch, S. (2012). Detecting causality in complex ecosystems. *Science, 338*(6106), 496–500. https://doi.org/10.1126/science.1227079

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological), 58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Toda, H. Y., & Yamamoto, T. (1995). Statistical inference in vector autoregressions with possibly integrated processes. *Journal of Econometrics, 66*(1–2), 225–250. https://doi.org/10.1016/0304-4076(94)01616-8

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., & Grave, E. (2023). LLaMA: Open and efficient foundation language models. *arXiv.* https://arxiv.org/abs/2302.13971

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30,* 5998–6008.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks, 5*(2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

Xu, M., et al. (2020). Estimating international trade status of countries from global liner shipping networks. *Royal Society Open Science, 7*(10), 200386. https://doi.org/10.1098/rsos.200386

Yudhistira, M. H., & Sofiyandi, Y. (2017). Seaport status, port access, and regional economic development in Indonesia. *Maritime Economics & Logistics, 20*(4), 549–568. https://doi.org/10.1057/s41278-017-0089-1

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x

IV. Appendices

A. Prompt of the FE approach

You are a maritime data analyst working for an international shipping research organization.

You are analyzing maritime news articles.

First, extract the following five attributes from the article:

1. **Scale** – The extent of the impact (Regional, National, Global).
2. **Type of Vessel** – The class or kind of vessel mentioned (Dry, Other).
3. **Vessel Size** – The vessel size category if specified (Capesize, Panamax, Supramax, Handysize, Other, All).
4. **Sea Route** – The affected sea route (C5, C3, All, Other).
5. **Duration of Positive Impact** – The length of time the positive effect is expected to last (Short, Medium, Long).
6. **Impact** – The direction of potential impact in hire rates (Negative, None, Positive).
7. **Hire rate impact** – Size/magnitude of that movement (None, Minor, Moderate, Significant, Major).

C5 is the route from Australia to China.

If any of the attributes cannot be determined from the article, say **None**.

News Article:

```
"{article_text}"
```

```
EXAMPLE JSON OUTPUT:

{"Scale": "National","Type of Vessel": "Other","Vessel Size": "All","Sea Route": "All","Duration of Positive Impact": "Long","Impact": "Negative","Hire Rate Impact": "Major"}
```

B. Prompt of the RAG approach

```
You are a maritime data analyst working for an international shipping
research organization.
You are analyzing maritime news articles, assisted by historical context
extracted from shipping documents.

First, review the following **Context**:

{context}

From each article, extract the following attributes:

1. **Scale** – The extent of the impact (Regional, National, Global).
2. **Type of Vessel** – The class or kind of vessel mentioned (Dry,
Other).
3. **Vessel Size** – The vessel size category if specified (Capesize,
Panamax, Supramax, Handysize, Other, All).
4. **Sea Route** – The affected sea route (C5, C3, All, Other).
5. **Duration of Positive Impact** – The length of time the positive
effect is expected to last (Short, Medium, Long).
6. **Impact** – The direction of potential impact in hire rates (Negative,
None, Positive).
7. **Hire rate impact** – Size/magnitude of that movement (None, Minor,
Moderate, Significant, Major).

C5 is the route from Australia to China.
If any of the attributes cannot be determined from the article, say
**None**.

News Article:

"{article_text}"
```

```
EXAMPLE JSON OUTPUT:
{{"Scale": "National", "Type of Vessel": "Other", "Vessel Size": "All",
"Sea Route": "All", "Duration of Positive Impact": "Long", "Impact":
"Negative", "Hire Rate Impact": "Major"}}
```

C. Prompt of the IDFE approach

```
You are a maritime data analyst working for an international shipping
research organization.
You are analyzing maritime news articles, assisted by historical context
extracted from shipping documents.

First, review the following **Context**:

{context}

From each article, infer and extract the following seven attributes:

1. **Scale** – The extent of the impact (Regional, National, Global).
2. **Type of Vessel** – The class or kind of vessel mentioned (Dry,
Other).
3. **Vessel Size** – The vessel size category if specified (Capesize,
Panamax, Supramax, Handysize, Other, All).
4. **Sea Route** – The affected sea route (C5, C3, All, Other).
5. **Duration of Positive Impact** – The length of time the positive
effect is expected to last (Short, Medium, Long).
6. **Impact** – The direction of potential impact in hire rates (Negative,
None, Positive).
7. **Hire Rate Impact** – The degree of potential impact in hire rates
(None, Minor, Moderate, Significant, Major).

**Instructions:**
- If the article or context explicitly mentions a vessel size, extract it.
- If vessel size is not clearly stated, write **None**.
- If the article or context mentions positive operational improvements,
increased demand, or optimism, **reasonably infer** a Minor or Moderate
hire rate improvement.
- If no clear signals are found even after reviewing the context, write
**None**.
- Be analytical — combine information from both the article and the
context.

*C5 is the dry bulk sea route from Australia to China.*
```

```
News Article:

"{article_text}"

EXAMPLE JSON OUTPUT:
{{"Scale": "National", "Type of Vessel": "Dry", "Vessel Size": "Capesize",
"Sea Route": "C5", "Duration of Positive Impact": "Short", "Impact":
"Positive", "Hire Rate Impact": "Moderate"}}
```

D. Prompt of the IDRBFE approach

```
You are a maritime data analyst at a global shipping research
organization.

Review the following **Context** providing background on market
conditions, vessel preferences, routes, and economic trends:

{context}

Based on the context and the article, infer and extract these seven
attributes:

1. **Scale** (Regional, National, Global)
2. **Type of Vessel** (Dry, Other)
3. **Vessel Size** (Capesize, Panamax, Supramax, Handysize, Other, All)
4. **Sea Route** (C5, C3, All, Other)
5. **Duration of Positive Impact** (Short, Medium, Long)
6. **Impact** (Negative, None, Positive)
7. **Hire Rate Impact** (None, Minor, Moderate, Significant, Major)

**Instructions:**
- Use context and article facts together.
- If vessel size is unknown, write **None**.
- Infer Minor or Moderate hire rate improvement if positive developments
are indicated.
- If fundamentals suggest major shifts, adjust your inference.
- If no clear signals are found, write **None**.

*Notes:*
- *C5 = Australia to China dry bulk route.*
- *C3 = Brazil to China iron ore route.*

News Article:
```

```
"{article_text}"

EXAMPLE JSON OUTPUT:
{{"Scale": "National", "Type of Vessel": "Dry", "Vessel Size": "Capesize",
"Sea Route": "C5", "Duration of Positive Impact": "Short", "Impact":
"Positive", "Hire Rate Impact": "Moderate"}}
```

E. Prompt of chunk selection; IDFE approaches

```
You are a maritime economics expert.

    Given the candidate chunks:

    {numbered_chunks}

    Select **5–10** chunks that best provide economic fundamentals to
understand shipping news.

    Focus on:
    - Freight demand drivers (iron ore, coal, grain)
    - Vessel types/sizes
    - Key routes (C5, C3) and seasonality
    - Hire rate patterns
    - Port congestion, trade disruptions
    - Long-term supply-demand shifts

    Guidelines:
    - Cover **different topics** if possible.
    - Prefer chunks with **explicit data or trends**.
    - Pick fewer if not enough are relevant.

    Reply format JSON:
    - **Only** list chunk numbers, comma-separated (e.g., `1, 5, 9`).
    - **No extra text.**
    - **No empty list, always at least 5 chunk numbers

    Prioritize relevance, facts, and diversity.
```

F. Prompt summarizing articles. IDRBFE approach

```
You are a maritime economics analyst assisting a document retrieval
system.

Given the following news article:

\"\"\"{article}\"\"\"

Your task:
- Summarize the article in **3 to 5 concise bullet points**.
- Focus on the **core facts and economic signals** that would help match
this article with **similar maritime content**.
- The summary should **highlight key topics**, such as:
    - Trade routes
    - Vessel types/sizes
    - Market impacts
    - Freight rates
    - Regulatory or operational changes
    - Port activity or disruptions

Guidelines:
- Use **clear, information-dense language**.
- Avoid vague generalities.
- Structure your output as a **clean bullet list** — no intro or
conclusion.

Your summary will be embedded and used to **improve cosine similarity
search**, so **prioritize phrases that capture the essence of the
article**.

Output format:
- Just bullet points (e.g. `- Maersk opens new Shanghai–LA route reducing
transit by 5 days`)
```

G. Prompt of chunk selection. IDRBFE approach

```
You are a maritime domain expert assisting an information extraction
system.

Your goal is to select **5 to 10 chunks** from the list below that are
most **relevant** to the article key points and useful for extracting the
following attributes:
```

```
- Scale
- Type of Vessel
- Vessel Size
- Sea Route
- Duration of Positive Impact
- Impact
- Hire Rate Impact

Given this article key points:
\"\"\"{key_points}\"\"\"

And these candidate chunks:
{numbered_chunks}

Guidelines:
- Prefer chunks that provide **explicit facts**, **specific data**, or
**clear descriptions**.
- Prioritize variety across topics (e.g., don't choose 5 route-related
chunks only).
- Discard vague or repetitive chunks unless they uniquely support one of
the attributes.

Output instructions:
- Reply **only** with the selected chunk numbers in a comma-separated list
(e.g., `2, 5, 7, 9, 12`).
- Do **not** include explanations or extra text.
```