

EFFICIENCY OF THE CRUDE OIL MARKET AND FORECASTING CRUDE
OIL RETURNS USING NEWS SENTIMENTS ¹

by

Tianyu Du

A thesis submitted in conformity with the requirements
for the degree of Honours Bachelor of Science
Department of Economics
University of Toronto

© Copyright 2020 by Tianyu Du

¹Compile Date: 01:42 Tuesday 21st April, 2020

Abstract

This paper focuses on the efficiency of the crude oil market and forecasting crude oil returns using news sentiments. We construct a weighted event sentiment score (WESEN) for each news article to capture the novelty and potential impact of the news. Moreover, this paper designs a framework to capture the dynamic of news and the market and proposes methods to test efficient market hypotheses. Experiment results show that the market is efficient and unpredictable with respect to support vector machines and random forests. In contrast, the market exhibits inefficiency and predictability while using long short-term memory recurrent neural networks. However, incorporating sentiment data does not help improve model performance in most experimental settings.

Contents

1	Introduction	5
1.1	Research Questions	5
1.2	Market Efficiency	7
2	Literature Review	10
3	Data	14
3.1	West Texas Intermediate (WTI) Crude Oils	14
3.2	Crude Oil Returns	15
3.3	Day of the Week Effect in Crude Oil Dataset	22
3.3.1	Difference in Returns across the Week	22
3.3.2	Kolmogorov-Smirnov test for Distributional Similarities	24
3.4	News Sentiment Dataset	26
3.4.1	Event Sentiment Scores	29
3.4.2	Weighted Event Sentiment Scores	32
3.4.3	Time of News Arrival	36
3.5	Classifying News Type	39
3.6	Case Studies	42
3.6.1	November 30, 2016: Positive Spike	42
3.6.2	December 6, 2018: Negative Spike	42
3.6.3	June 12~13, 2019: Positive Spike in Down Period	43
4	Model	45

4.1	Framework: An Example	45
4.2	Formal Framework	46
4.2.1	Timestamps	46
4.2.2	States of World	47
4.2.3	Information Flow	47
4.2.4	Characteristic Functions for Feature Construction	50
4.2.5	Gathering-First Paradigm and Summarizing-First Paradigm . . .	52
4.2.6	Intertemporal Dependency	53
4.3	Empirical Model	53
5	Experiments	58
5.1	Procedures	58
5.1.1	Feature Constructions	58
5.1.2	Rolling-Window Method	61
5.1.3	Performance Metrics	63
5.1.4	Searching Technology (Model Selection) and Randomized Cross Validation	64
5.2	Baseline Models: The Naive Predictor and Moving Average Predictors .	66
5.3	Linear Models: Autoregressive Integrated Moving Average	69
5.4	Support Vector Regression	71
5.5	Random Forests	75
5.6	Long Short-Term Memory Recurrent Neural Networks	78
5.7	Taking the Day-of-the-Week Effect into Consideration	81
6	Conclusions	85
	Bibliography	88
7	Appendix	92

Chapter 1

Introduction

1.1 Research Questions

Algorithmic trading systems are playing a significant role in nowadays financial markets. Strategies in most of these trading systems are based on predictive models: the system sells (buys) assets if the model predicts the price will fall (rise). Therefore, profitabilities of algorithmic trading systems highly depend on predictive models. This paper focuses on the crude oil market since crude oil prices serve not only as investing instrument but also as an important predictor of other economic indicators. Studying whether it is possible to build new predictive models and improve current models is relevant to both investors and economists.

The contribution of this paper to current literature is three-fold:

- This paper examines whether the crude oil market is predictable (efficient) with respect to historical crude oil price/return movements and sentiments of news related to crude oils.
- With the series of crude oil returns from Energy Information Administration (EIA) and the dataset of news sentiments from Ravenpack News Analytics (RPNA), this paper constructs a return series from spot prices of crude oil and identifies the day-of-the-week effect in crude oil returns. The autocorrelation function and partial autocorrelation function suggest that the intertemporal correlation with return

series is weak, therefore, future returns are essentially unpredictable using historical returns only. Results of Kolmogorov-Smirnov test suggest that the empirical distribution of Mondays' returns are significantly different from distributions of returns on other days of the week. To utilize news sentiments in return forecasting, this paper constructs a multivariate time series of news sentiments from individual news articles, such series allows machine learning models to predict returns based on news sentiment simply by including more independent variables.

- Lastly, this paper proposes a framework for forecasting crude oil returns using news sentiments. Using this framework, results of experiment suggest that incorporating news sentiments does make crude oil returns more predictable but this is not true for all classes of models

There are three major benchmarks for crude oil price: West Texas Intermediate (WTI), Brent Crude and Dubai/Oman and they measure tradings of crude oils produced in the U.S., Europe, and the Middle East respectively. Mann and Sephton (2016) examine the relationship between these three crude oil price benchmarks and indicate that these benchmark prices are tied by a long-run relationship. Moreover, their analysis shows that all three benchmarks are moving to restore the observed long-run relationship in at least one regime (Mann and Sephton 2016). Therefore, the same predictive model should achieve similar performances on three markets at least in the long run, and conclusions drawn on one market can be extended to the other two markets. Because the news sentiment dataset used in this paper consists of news published by four U.S. based publishers, this paper focuses on the WTI crude oil spot price. Instead of predicting the spot price of crude oil, I am going to forecast the series of empirical returns since returns can better reflect the profitability. This paper aims to answer the following research questions:

Whether the daily return of crude oil is efficient or not? Can we better predict (i.e., exploit market inefficient) returns by incorporating news sentiments?

In particular, this paper aims to examine the feasibility of predicting the empirical return on crude oil the next day based on information up to the current day.

1.2 Market Efficiency

Research questions can be answered by testing different versions of the efficient market hypothesis (Fama 1970, Fama 1991). Jensen (1978) provides a minimal definition of market efficiency and the EMH holds if the market is efficient.

Definition 1.2.1 (Market Efficiency I). A market is **efficient** with respect to an information set Ω if it is not profitable to trade solely using this information set.

Definition 1.2.2 (Efficient Market Hypothesis I). The **efficient market hypothesis** (EMH) holds for one market with respect to information set Ω if the market is efficient according to Definition 1.2.1.

The strength of one particular version of EMH depends on the content of Ω in its definition.

Definition 1.2.3. For convenience, this paper defines the following two information sets.

- Ω_{partial} : the information set containing historical returns only,
- Ω_{complete} : the information set consisting of both historical returns and news sentiment.

The weaker version of EMH consists of a Ω_{partial} as its information set and suggests that any trading strategy using a forecaster based on historical price movement would only generate zero (expected) economic profit.

Similarly, this paper uses Ω_{complete} to define the stronger EMH, which suggests that the current crude oil market has already absorbed and reflected all price movements and news. Empirically, the stronger EMH suggests that a trading strategy built upon predictive models using both historical returns and news will not generate positive (expected) economic profit.

Timmermann and Granger (2004) review the concept of EMH in the context of forecasting and extend Jensen's definition to include types of predictive models (called **model class**) and model selection method (termed **search technology**):

Definition 1.2.4 (Market Efficiency II). A market is **efficient** with respect to the information set Ω , search technologies S , and the class of predictive models \mathcal{M} if it is impossible to make economic profits by trading on the basis of signals produced from a predictive model defined over predictor variables constructed from Ω and selected using S (Timmermann and Granger 2004).

Definition 1.2.5 (Efficient Market Hypothesis II). The **efficient market hypothesis** (EMH) holds for one market with respect to information set Ω , class of predictive model \mathcal{M} , and searching technology S if the market is efficient according to Definition 1.2.4.

I select best models from various classes of machine learning models such as random forests, support vector machines and deep neural networks using randomized cross-validation techniques (search technology).

To simplify the problem, this paper assumes the transaction cost to be negligible.

Assumption 1.2.1. Transaction costs are negligible.

Consequently, if we can construct a model that attains higher than 50% accuracy in forecasting crude oil returns, we can construct a simple trading algorithm based on this model and gain positive expected profits.

Proposition 1.2.1. A market is efficient if and only if it is unpredictable.

Using Proposition 1.2.1, this paper is testing EMH by examining whether the market is predictable or not.

For each information set Ω , predictive model \mathcal{M} , and searching technology S , the corresponding EMH in Definition 1.2.5 can be tested by comparing the test time performances of the optimal \mathcal{M} identified by S and other benchmark predictors. For instance, take $\Omega = \Omega_{\text{partial}}$, \mathcal{M} to be random forests, and S to be cross validation. Firstly, we can construct a naive predictor predicting zero crude oil returns all the time. Then if the best-performing random forest chosen using S and defined on Ω_{partial} out-performs the native predictor significantly, then we can conclude crude oil market is predictable (equivalently, inefficient) with respect to (Ω, \mathcal{M}, S) .

Because contents of Ω_{partial} , predictive models and the search technology in this paper are all publicly available, the predictive power of models trained on the partial information should be self-destructive: as more traders find this predictive power, they will trade accordingly and this advantage eventually vanishes (Timmermann and Granger 2004).

This paper looks into the 20 year period from 2000 to 2019, we do not expect predictive powers (if any) of models based on publicly available information to be persistent over 20 years. Therefore, we expect the crude oil market to be efficient on the partial information set.

Given the class of predictive models \mathcal{M} and search technology S , I am going to test both the weaker EMH defined by $(\Omega_{\text{partial}}, \mathcal{M}, S)$ and the strong EMH defined by $(\Omega_{\text{complete}}, \mathcal{M}, S)$.

If both versions of EMH hold, this paper could (i) conclude the market is unpredictable (efficient) using information, models and searching technologies defined above and (ii) news sentiments do not help predict crude oil returns.

In contrast, if the weak EMH holds but strong version fails, this servers as an evidence suggesting (i) the crude oil market is predictable and (ii) incorporating news sentiments helps predict returns.

Subsequent sections of this thesis consist of the following parts: literature review, data analysis, framework, experiments, and conclusion. The literature review section provides a brief review of current time series forecasting methodologies and research on forecasting crude oil prices/returns. In the data analysis section, we compute empirical returns of crude oil from the series of spot prices. In addition, this paper analyzes the return series and news sentiment datasets in detail. Then, this paper proposes a structured model capturing the interdependencies among states of the world, crude oil return series, and flow of news. Using the proposed framework, this paper formulates the forecasting problem into a generic supervised learning problem that fits into a vast majority of existing machine learning models. We then test our null hypothesis by comparing the performances of different models under the proposed framework. Lastly, we conclude our findings and discuss the limitations and potential improvements.

Chapter 2

Literature Review

Traditional time series methods explore endogenous patterns encoded in the series of returns and use lagged values of returns to forecast future returns. Mohammadi and Su (2010) examine the performance of autoregression integrated moving average-generalized autoregressive conditional heteroskedasticity (ARIMA-GARCH) on eleven international crude oil market. The authors apply ARIMA-GARCH models to forecast the value and volatility of weekly crude oil returns in those markets and conclude that the return is characterized by a MA(1) process (Mohammadi and Su 2010).

Besides using historical returns as features, many are utilizing other technical indicators of the market as well. These methods aim to transform lagged values of return into meaningful predictors such as moving averages (MA) and moving average divergence convergence (MADC). Baker and Wurgler (2006) construct an investor sentiment index for the stock market using a collection of technical indicators: the closed-end fund discount, NYSE share turnover, numbers of IPOs, average first-day returns, and share of equity issues in total equity and debt issues, dividend premium. Moreover, another recent study constructs market sentiment indices for both WTI and Brent oil future markets, and these constructed indices have shown significant predictive power while controlling external variables such as stock indices and exchange rates (Deeney et al. 2015).

Besides exploring the predictive power of market indices, scholars are paying more attention to alternative data sources such as news. The inceptions of most market partitioners are in fact shaped by what news they have heard, and these partitioners trade

commodities, stocks, and other financial derivatives based on news they receive. In Bybee and others' recent work, they analyze the full-text contents of over 800,000 articles in the Wall Street Journal over the past 30 years. They have demonstrated that text-based features from news articles can track economic activities accurately. Moreover, these text-based features have additional predictive powers to traditional macroeconomic indicators for macroeconomic forecasting (Bybee et al. 2019). Most works exploring and utilizing the predictive power of news sentiments are focusing on the stock market. Tetlock analyzes the interaction between articles in the "Abreast of the Market" column in the Wall Street Journal and the stock market. Using vector autoregression (VAR), Tetlock models the intertemporal correlation between the stock market and a measure of media pessimism constructed using principal component analysis (PCA). The author finds that pessimistic signals in news media can be a precursor of downward pressure on stock price and high trading volume (Tetlock 2007). Mudinas, Zhang, and Levene demonstrate that news sentiments extracted from Financial Times and tweets Granger cause prices of several stocks in S&P500. Experiments in their paper suggest that the prediction accuracies of support vector machines and recurrent neural networks are improved by utilizing additional news sentiment features (Mudinas, Zhang, and Levene 2019). Hu and others design a Hybrid Attention Network (HAN) to extract information and forecast price movements. Beyond accurately predicting trends in the stock market, trading algorithms based on the proposed predictive model demonstrate superior annualized return in the Chinese stock market compared with other algorithms (Hu et al. 2018).

Instead of the stock market, Roache and Rossi analyze the impacts of macroeconomic announcements on daily prices of 12 commodity futures including oil, heating oil and natural gas between 1997 and 2009. Their experiments show that commodities are in general insensitive to macroeconomic news and models based on news perform poorly on forecasting daily prices of commodities (Roache and Rossi 2010). Brandt and Gao examine the potentially different impacts on crude oil markets from news about macroeconomic fundamentals and geopolitical events. News about geopolitical events shows strong short-run impacts on the crude oil market. In the long run, news about macroeco-

nomic fundamentals acts as a significant predictor of crude oil returns. In contrast, news about geopolitical events only induces uncertainty and higher trading volume (Brandt and Gao 2019).

The sentiment of one news article is highly subjective, assigning sentiment scores to articles manually will inevitably lead to biased sentiment scores. Instead of asking an expert in finance to evaluate the article after reading it, each article's sentiment score should be based completely on a pre-defined scoring rule which is (mostly) independent of each individual article so that the score is as objective as possible. Modern natural language processing (NLP) techniques allow researchers to construct sentiment indices for a large volume of news articles without actually reading all texts. One simplest method of constructing objective news sentiment is the dictionary-based method: researchers firstly build a dictionary mapping frequently observed words in financial news articles to their general sentiment. Loughran and McDonald propose a dictionary classifying words into six categories: negative, positive, uncertainty, litigious, modal and constraining (Loughran and McDonald 2011, Bodnaruk, Loughran, and McDonald 2015, Loughran and McDonald 2016). For instance, the word ‘bankrupt’ is classified as a word carrying negative sentiment in the Loughran-McDonald sentiment dictionary. Then, the algorithm divides the article into single words (the tokenization step) and reduce each word to their lemma (the lemmatization step). For example, the word ‘bankruptcy’ would be replaced by ‘bankrupt’. Afterward, the algorithm counts occurrences of words in the article and calculates the sentiment score based on frequencies of words belonging to each category (e.g., positive and negative words). One potential drawback of the dictionary-based method is that it can cover frequently used words only: the Loughran-McDonald covers around 86,000 vocabularies. Another issue is that the dictionary is domain-specific, the dictionary built for the stock market may not be optimal for the crude oil market. Models designed for one market cannot be easily transferred to another market without rebuilding the dictionary. Another method of constructing sentiment is based on word embedding techniques in NLP. The embedding algorithm maps each word to a high-dimensional vector, termed embedding vector, in the embedding space so that words with close meanings would have close embedding vectors. Pennington, Socher,

and Manning introduce a Global Vectors for Word Representation (GloVe) algorithm to embed 2.2 millions of commonly used words to 300-dimensional embedding vectors (Pennington, Socher, and Manning 2014). Embedding techniques cover a much wider range of vocabularies compared with dictionary-based methods so that models can be migrated easily.

Chapter 3

Data

In order to answer research questions, this paper involves two datasets (i) the daily spot price of crude oil of the West Texas Intermediate (WTI) from which returns are computed, (ii) the news sentiment dataset from Ravenpack News Analytics (RPNA).

3.1 West Texas Intermediate (WTI) Crude Oils

West Texas Intermediate (WTI) is a class of light and sweet crude oil that has served as a benchmark for crude oil prices over the past few decades. Cushing, Oklahoma, where the Cushing oil field locates, has been the delivery point for commodities behind crude oil contracts traded at New York Mercantile Exchange (NYMEX). The U.S. Energy Information Administration (EIA) provides daily closing spot prices of WTI crude oil delivered from Cushing. This time series can serve as a benchmark of measuring activities in the global crude oil market.

This paper focuses on crude oil prices between January 1, 2000 and October 31, 2019. Baumeister and Kilian (Baumeister and Kilian 2016) suggest that the spot price is highly responsive to news and other macroeconomic shocks, which is exactly the tricky part of forecasting financial time series. If the proposed forecasting algorithm performs well on the crude oil dataset, such an algorithm is conceivably promising on other datasets as well.

3.2 Crude Oil Returns

This paper focuses on crude oil returns instead of prices for two reasons, (i) accuracy on return prediction can better reflect the potential profitability and (ii) the series of returns is more well-behaved compared with the series of prices.

Specifically, The augmented Dickey-Fuller test on the raw price series gives a p -value of 0.26, which suggests that the movement of crude oil prices exhibits significant non-stationarity. Models designed for non-stationarity are much more complex than models for stationary series. Moreover, most classical time series models, which serve as benchmark models, assume the series to be stationary and the non-stationarity violates such assumption. The efficient market hypothesis cannot be tested without benchmark models.

The closing spot prices of crude oils are available at a daily frequency for weekdays only. Besides weekends, observations are missing on holidays when the exchange market is closed. In following sections, this article refers to these days with valid spot price as **trading days**.

Table 3.1 reports dates that are most frequently associated with a missing data over the span of 20 years. The set of days with missing data is consistent over these years: the market is always closed on January 1, July 4 (Independence Day) and December 25 (Christmas). Because price data range from January 3, 2000 to October 31, 2019, missing data problems on December 25 are only detected for 19 times in the table. Lastly, the group of dates in late November are responsible for missing data on Thanksgiving holidays since Thanksgiving holiday varies year by year.

Table 3.1: Top Days with Missing Data

Date	Number of Days with Missing Data
July 4	20
January 1	20
December 25	19
July 3	10
November 23	10
November 24	10
November 25	10
November 22	9
November 26	9

There are only ten weekdays with missing data problem each year on average (3.77% of the entire dataset). The insignificant percentage of missing data allows us to drop those dates without hurting the generalizability of models and experiments in subsequent sections.

For one particular trading day t with closing price p_t , let Δ denotes the gap (in terms of the number of calendar days) between date t and the previous trading day, so that $t - \Delta$ is the last trading day before trading day t . This paper defines the return on day t , denoted as r_t , as the continuously compounded rate of return in equation (3.2.1).

$$r_t = \frac{\ln(p_t) - \ln(p_{t-\Delta})}{\Delta} \times 100\% \quad (3.2.1)$$

Moreover, all returns are expressed in percentage points.

The time gap between two observed prices are not uniform. For instance, the return on a Monday can be computed by taking difference between the log close price on Monday and the previous Friday, if available. In this case, $\Delta = 3$ and $r_t = \frac{\ln(p_{\text{Mon}}) - \ln(p_{\text{Prev Fri}})}{3}$. If the previous Friday was not a trading day with valid spot price, then $\Delta = 4$ and the return r_t will be $\frac{\ln(p_{\text{Mon}}) - \ln(p_{\text{Prev Thu}})}{4}$.

Table 3.2: Distribution of Δ by Weekdays

Day of the week	Num. Days.	Num. Trading Days	$\Delta=1$	2	3	4	5
Monday	1,034	931	0	0	887	33	11
Tuesday	1,035	1,023	926	0	0	97	0
Wednesday	1,035	1,027	1,016	5	0	0	6
Thursday	1,035	1,007	999	8	0	0	0
Friday	1,034	990	973	17	0	0	0
Saturday	1,035	0	0	0	0	0	0
Sunday	1,035	0	0	0	0	0	0
Total	7,243	4,978	3,914	30	887	130	17

Table 3.2 summarizes the distribution of Δ values. The Δ values for Mondays are at least 3 because weekend data are always unavailable. One extreme case is that none of Monday and Tuesday is a trading day, so that the Δ value for the Wednesday in this week would be 5. The extreme case occurs rarely for only 6 times during the period of 20 years.

The movement of crude oil returns in the past two decades has exhibited volatile patterns. Figure 3.1 plots the pattern of returns, in which shaded areas indicate U.S. recessions (March 2001 to November 2001 and December 2007 to June 2009). Noticeably, crude oil returns are more volatile during recession periods and are becoming more volatile in recent years.

Figure 3.1: Crude Oil Returns

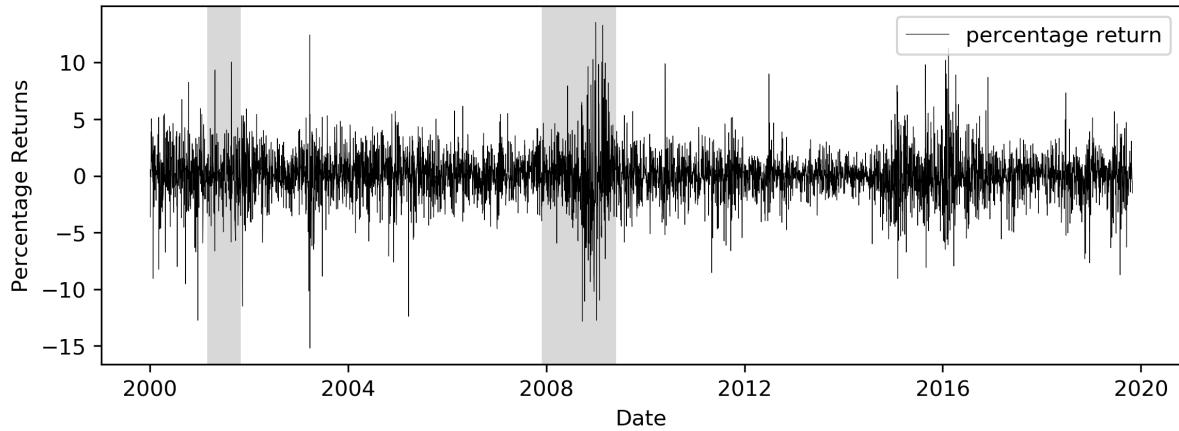


Table 3.3 reports summary statistics for the percentage crude oil returns, in which normalized skewness and excess kurtosis are defined as

$$\hat{m}_3 := \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^3}{\sigma_x^3} \text{ (normalized skewness)} \quad (3.2.2)$$

$$\hat{m}_4 := \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^4}{\sigma_x^4} - 3 \text{ (excess kurtosis)} \quad (3.2.3)$$

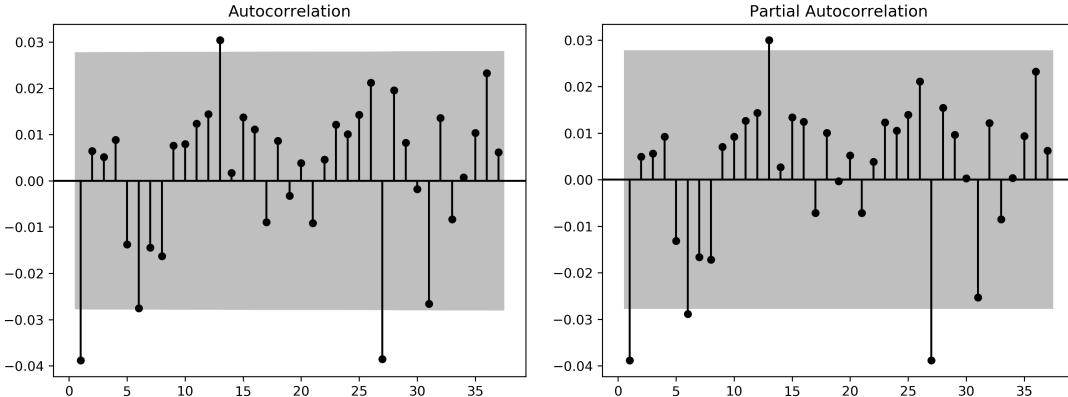
$$\text{where } \sigma_x := \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.2.4)$$

Summary statistics in the Table 3.3 suggest that the mean return within each year are nearly zero, which agrees with the conventional expectation that returns are zero on average. During periods of recessions, the average returns are below -0.2% and the series becomes significantly more volatile as well. Moreover, given the high kurtosis between 2008 and 2009, one are more likely to encounter extreme returns, both positive and negative, during recession periods.

Table 3.3: Summary Statistics for Crude Oil Returns

Year	Obs.	Mean	Median	Std.	Min	Max	Normalized Skewness	Excess Kurtosis
2000	249	0.03433	0.20148	2.61996	-12.74152	8.26343	-0.92174	3.45580
2001	250	-0.02409	-0.04434	2.54058	-11.48581	10.05107	-0.06444	3.15304
2002	250	0.15535	0.15221	1.70283	-5.86460	5.43272	-0.22297	0.62431
2003	250	0.07861	0.13203	2.57315	-15.19090	12.44253	-0.89439	7.30189
2004	249	0.08918	0.11605	2.08792	-7.60501	5.70121	-0.38117	1.01395
2005	251	0.05257	0.11019	1.96717	-12.39009	5.02715	-1.04498	5.84007
2006	249	-0.00539	0.12995	1.58949	-4.45214	6.15402	0.13487	1.03258
2007	252	0.23400	0.09798	1.69800	-4.66915	5.51381	0.13705	0.65946
2008	253	-0.29945	-0.07920	3.34992	-12.82672	13.54551	-0.01650	2.60308
2009	252	0.26537	0.19157	2.92040	-12.74310	13.29544	0.29333	4.25972
2010	252	-0.02077	0.03198	1.74554	-5.18874	9.89802	0.39313	3.82001
2011	252	0.00583	0.10994	1.94170	-8.53498	5.18170	-0.69170	2.27400
2012	252	-0.04164	0.03600	1.51078	-4.76060	9.00091	0.54820	5.53225
2013	252	0.01455	0.04489	1.06690	-3.46951	3.20999	0.05495	0.67398
2014	252	-0.16510	-0.05343	1.36052	-5.98638	4.91592	-0.76983	3.16348
2015	252	-0.03610	-0.25616	2.63361	-9.05140	9.81397	0.24129	1.25225
2016	252	0.20931	0.00000	2.79698	-7.95603	11.28922	0.70466	2.11826
2017	250	0.06564	0.17286	1.40987	-5.56187	3.32016	-0.87368	2.07271
2018	249	-0.10076	0.07393	1.81925	-7.67683	7.33414	-0.64252	3.38603
2019	210	0.04359	0.10073	1.93931	-8.72444	5.67862	-0.66251	2.87153
2000~2019	4978	0.02754	0.06307	2.15250	-15.19090	13.54551	-0.16152	5.12757

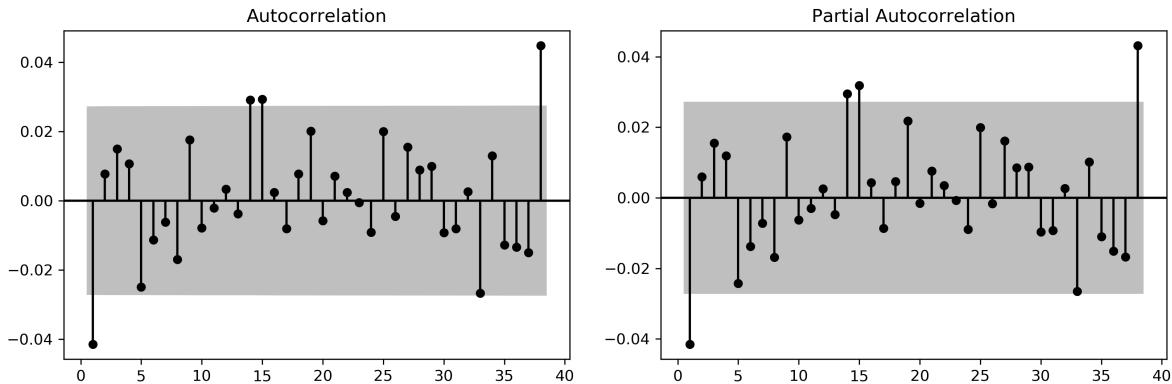
Figure 3.2: ACF and PACF for Crude Oil Returns (missing data dropped)



The autocorrelation function (ACF) and partial autocorrelation function (PACF) plots in Figure 3.2 explore the intertemporal correlation within the return series. Since only a few lags are statistically significant in the ACF and PACF plots, we do not expect linear time series models are capable to achieve high performances in this return prediction task.

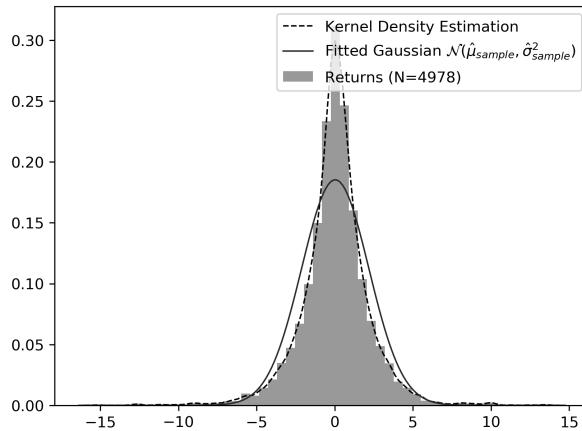
It is worth noticing that lag 1, 6, 13, 27 are significant in both ACF and PACF plots, which may indicate seasonalities with period of one week. However, regularity in missing data can lead to this observation as well. Hence, instead of dropping days with missing data, we fill up these missing data using random values from a Gaussian distribution parameterized by the mean and variance of the entire dataset. Figure 3.3 plots the ACF and PACF of return series with missing values filled using random noise, the significance at lag 6 disappeared, but the significance of bi-weekly lag persists and another spike at lag 36 emerges. This observation indicates that there might be seasonality with bi-weekly periods.

Figure 3.3: ACF and PACF for Crude Oil Returns (missing data filled)



The histogram in Figure 3.4 suggests that the empirical distribution of crude oil returns is much clustered near zero than a Gaussian distribution. With this clustering feature, conventional metric for evaluating regression models, such as mean squared error (MSE), will not be sufficient in this task. For instance, a dummy model consistently predicting zero will attain a fair MSE (to be specific, the variance of entire dataset). Therefore, in later sections, we introduce another directional accuracy to assess the fitness of models.

Figure 3.4: Distribution of Crude Oil Returns



3.3 Day of the Week Effect in Crude Oil Dataset

3.3.1 Difference in Returns across the Week

Gibbons and Hess (1981) examined returns on stocks from S&P 500, Dow Jones 30, and Treasury Bills. They find strong negative mean returns on Monday compared with other weekdays. The seasonality persisted even after taking market adjustment measures, such as using mean-adjusted returns instead (Gibbons and Hess 1981). Analysis in this paper unveils a similar daily seasonality presents in crude oil returns as well. Panels in Figure 3.5 demonstrate the empirical distributions of returns on each day of the week and N s within parentheses in captions denote the number of observations. We can see that Mondays and Wednesdays have relatively larger variances, which again matches Gibbons and Hess' observations.

Figure 3.5: Distributions of Returns on Each Day of the Week

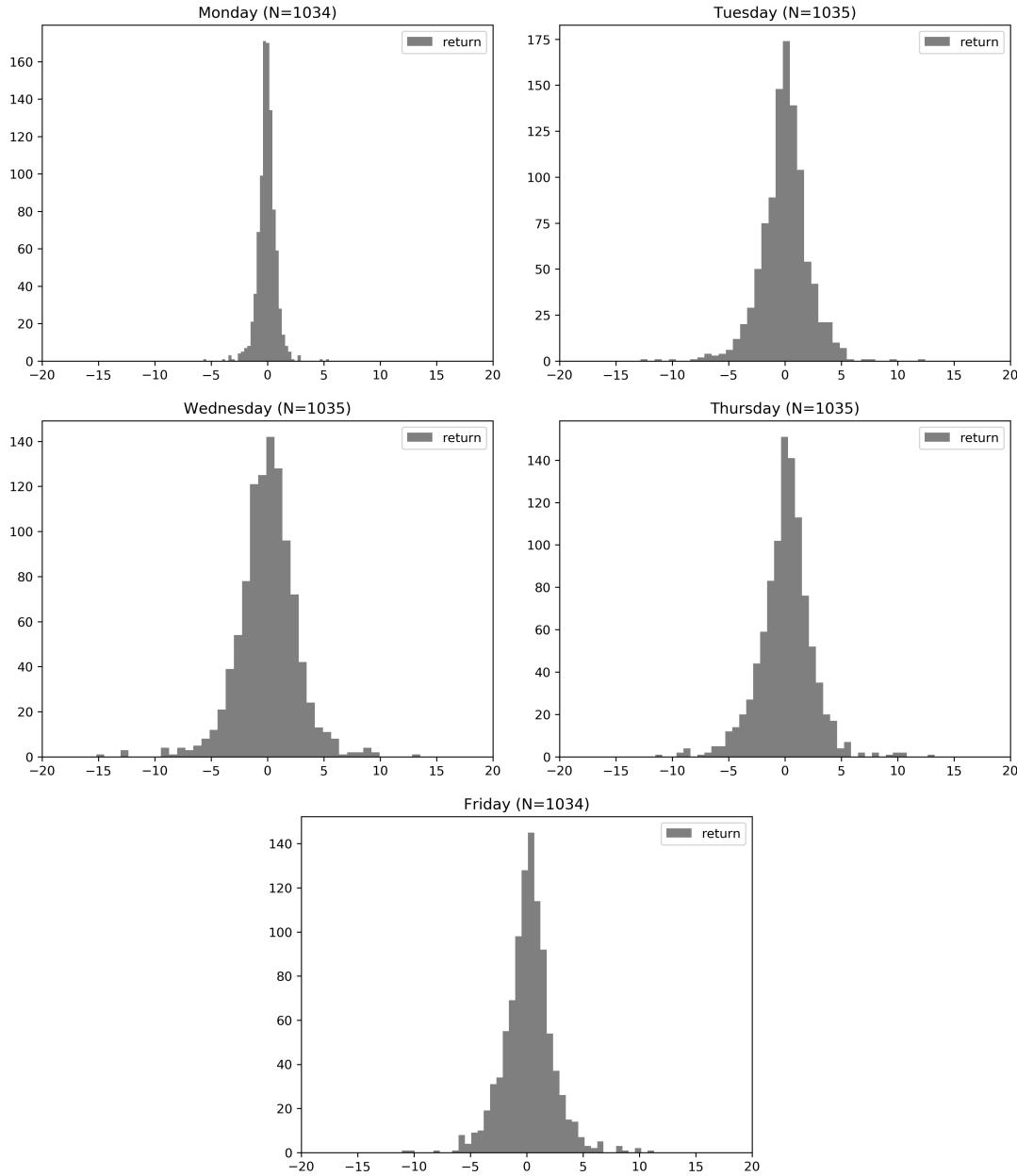


Table 3.4 provides summary statistics for prices and returns on each day.¹ It turns out that at a significance level of 0.05, Monday and Friday are the only two weekdays with a mean return significantly different from zero. And the *t*-test suggests Mondays are more likely to associate with negative returns, meanwhile, Friday is more often associated with

¹In Table 3.4, a value of -0.000 indicates a negative value with magnitude less than 0.0005 . P -values are calculated in a two-tailed *t*-test with null hypothesis $\mu_0 = 0$. Bold fonts indicate statistically significance at level $\alpha = 0.05$.

positive returns.

Table 3.4: Summary Statistics of Crude Oil Returns on Each Day of Week

Day of the week	Num. Obs.	Mean (<i>P</i> -Value)	Std.	Normalized Skewness	Excess Kurtosis
Monday	931	-0.055 (0.042)	0.816	-0.134	7.088
Tuesday	1,023	-0.034 (0.615)	2.141	-0.335	4.214
Wednesday	1,027	-0.000 (0.998)	2.660	-0.325	3.798
Thursday	1,007	0.069 (0.361)	2.378	-0.041	3.64
Friday	990	0.155 (0.026)	2.194	0.128	3.243
Total	4,978				

3.3.2 Kolmogorov-Smirnov test for Distributional Similarities

Smirnov develops a non-parametric method of testing the equality between two continuous distributions, with CDFs $F(x)$ and $G(x)$ respectively (1939). Hodges (1958) provides more details on the Kolmogorov-Smirnov test and relevant methods. I am using the two-tailed version of Kolmogorov-Smirnov test to check whether distributions of two different days are similar. Given two datasets, take returns on Mondays and Tuesdays for example, the null hypothesis says those two datasets are drawn from the same distribution, and the alternative says they are from different distributions.² Firstly, the Kolmogorov-Smirnov test constructs the empirical CDFs $F_{Mon,927}(x)$ and $F_{Tue,1018}(x)$ from the dataset. Then, the Kolmogorov-Smirnov statistic measures the maximum discrepancy between two distribution functions, which is

$$D := \sup_x |F_{Mon,927}(x) - F_{Tue,1018}(x)| \in [0, 1] \quad (3.3.1)$$

A smaller D -statistic implies stronger distributional similarity between two distributions. For instance, when $F_{Mon,927}(x)$ and $F_{Tue,1018}(x)$ are exactly the same, the D -statistic is

²Different alternative hypotheses can be used in Kolmogorov-Smirnov test: i) $H_1 : F(x) \geq G(x)$, ii) $H_1 : F(x) \leq G(x)$, and iii) $H_1 : F(x) \neq G(x)$. This paper is using the third (two-tailed) alternative hypothesis.

zero. In contrast, let $X = 0$ and $Y = 1$ be two "deterministic" random variables, in this case, there distributions are completely different, and $D_{X,Y} = 1$.

The test rejects H_0 at a significance level of α if

$$D > \sqrt{-\frac{1}{2} \ln \frac{\alpha}{2}} \sqrt{\frac{n+m}{nm}} \quad (3.3.2)$$

where m and n denote sizes of two datasets.

Table 3.5: D -Statistics in Kolmogorov-Smirnov Tests

D -Statistic (P -Value)	Monday	Tuesday	Wednesday	Thursday	Friday
Monday	0.000(1.000)	0.193(0.000)	0.243(0.000)	0.189(0.000)	0.180(0.000)
Tuesday		0.000(1.000)	0.064(0.030)	0.064(0.030)	0.071(0.010)
Wednesday			0.000(1.000)	0.058(0.062)	0.084(0.001)
Thursday				0.000(1.000)	0.030(0.729)
Friday					0.000(1.000)

Table 3.5 presents the Kolmogorov-Smirnov D -Statistic for distributions of every pairs of days. Bold font in this table indicates the null hypothesis is rejected at a significance level of 0.01, which implies discrepancy in distributions. At a significance level of 0.01, we can see that Mondays follow a distribution significantly different from distributions that other weekdays follow. Because the dataset does not contain weekend data, returns on Mondays is always computed using the difference between log prices on Monday and the previous Friday (Thursday if Friday is not a trading day and so on). Therefore, one explanation to the discrepancy in distributions is that returns on Mondays is directly impacted by the weekend effect. Moreover, the distribution of returns on Mondays (over weekends) is the only one with negative mean among distributions of all five days according to results of t -tests reported in Table 3.4.

3.4 News Sentiment Dataset

The event sentiment dataset from RavenPack News Analytics (RPNA) tracks and analyzes all information of companies, organizations, countries, commodities, and currencies from four major sources: Dow Jones Newswires, Wall Street Journal, Barron's and MarketWatch. This dataset covers events from January 1, 2000, to October 30, 2019. RavenPack records the exact date and coordinated universal time (UTC) when each news article is published. Since the crude oil prices are from New York Exchange (NYEX), which uses US Eastern time, this UTC time is converted into Eastern time.

Definition 3.4.1. This paper defines one **news item** to be one news article included in the RPNA dataset. One news item is a concrete published article associated with a headline, body text, a date when it is published and other information.

In the following discussion, I am using news items, pieces of news, news articles or simply news interchangeably but all of them are equivalent to news items.

Each piece of news in the dataset is assigned with a topic based on its headline and text. In order to filter out noisy and irrelevant information, this paper works on the subset of news with crude oil topic as the main source of news. It turns out that there are 106,960 news article from the original dataset with topics "crude oil", this results in 17 piece of news per day on average.

Noticeably, there could be multiple news articles reporting the same event and this could lead to duplicate counting problems. Later in this section, an alternative measure of news sentiment is constructed to mediate this problem.

Table 3.6 provides summary statistics of numbers of news about crude oil reported each day by years. In 2004, 2008 and 2012, crude oil topic is of broad interest and there were about 20 news about crude oil per day in these years. In contrast, publishers are quieter in 2002, 2003 and recent years.

Table 3.6: Summary Statistics for Daily Numbers of News Items Arrived by Years

Year	Mean	Median	Std.	Min	Max	Normalized Skewness	Excess Kurtosis
2000	8.665	8.000	8.315	0.000	48.000	1.228	1.939
2001	11.493	11.000	10.247	0.000	51.000	0.682	0.059
2002	3.542	3.000	3.642	0.000	19.000	1.403	2.368
2003	5.126	3.000	6.145	0.000	39.000	2.058	5.646
2004	20.776	19.000	17.680	0.000	84.000	0.728	0.193
2005	17.473	17.500	13.796	0.000	57.000	0.403	-0.460
2006	18.615	19.000	14.272	0.000	58.000	0.247	-0.862
2007	16.781	16.000	13.669	0.000	66.000	0.567	-0.187
2008	20.500	22.000	15.141	0.000	66.000	0.304	-0.562
2009	14.499	14.000	10.988	0.000	48.000	0.296	-0.761
2010	15.564	17.000	11.437	0.000	52.000	0.247	-0.753
2011	19.187	20.000	14.175	0.000	65.000	0.231	-0.610
2012	20.077	22.000	14.682	0.000	65.000	0.206	-0.688
2013	14.526	15.000	11.364	0.000	57.000	0.413	-0.374
2014	13.353	11.000	13.445	0.000	69.000	1.502	2.596
2015	18.663	18.000	15.974	0.000	80.000	0.738	0.188
2016	19.956	18.000	17.454	0.000	101.000	0.837	0.661
2017	12.479	11.000	10.927	0.000	58.000	0.797	0.619
2018	13.277	13.000	11.490	0.000	93.000	1.350	5.481
2019	10.505	9.000	10.608	0.000	65.000	1.067	1.569

In general, weekends are quiet period of news arrival, while much more news arrive in the middle of each week. Figure 3.6 summarizes the average numbers of news on each day of the week, and Table 3.7 summarizes the average number of news on each day in each year. It turns out that weekends are much quieter than weekdays and only less than 5 % of all news are reported on weekends. Moreover, the number of news arrivals peaks on Wednesday in all years but 2000 and 2002.

Figure 3.6: Average Numbers of News on Each Day of Week

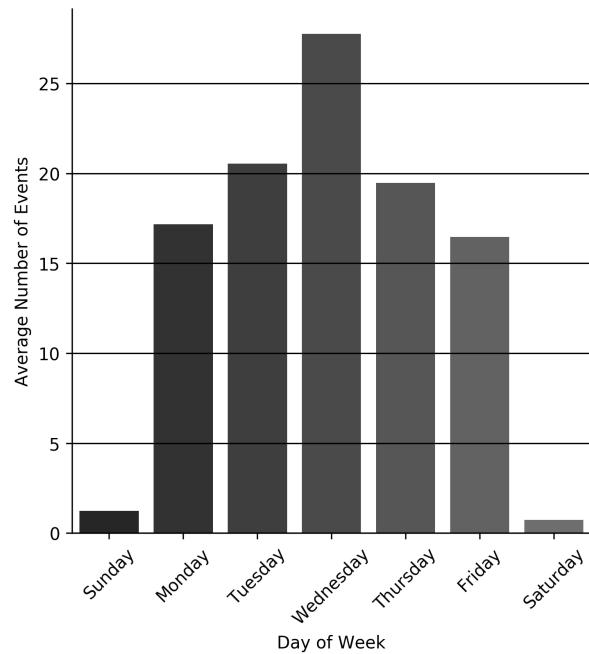


Table 3.7: Average Numbers of News Items Published on Each Day of Week in Each Year

Year	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
2000	11.157	14.135	13.077	11.885	9.769	1.643	1.500
2001	12.547	17.569	21.327	15.058	14.078	1.000	1.200
2002	5.771	5.019	5.224	3.980	5.469	1.200	1.600
2003	7.080	6.529	9.942	6.863	5.490	1.200	1.136
2004	24.058	28.981	39.250	28.660	22.302	2.182	2.240
2005	21.462	21.846	33.596	24.654	19.000	1.765	2.259
2006	22.981	24.885	35.904	24.846	19.731	1.346	2.161
2007	19.792	21.385	33.577	23.846	16.769	1.941	2.212
2008	24.788	26.415	36.415	26.269	25.250	2.207	3.065
2009	16.058	21.346	29.192	16.925	15.538	1.688	2.366
2010	16.327	23.058	28.654	20.596	17.135	2.261	2.932
2011	23.769	28.577	32.904	25.750	19.942	2.053	3.441
2012	22.340	26.654	36.423	26.981	25.118	3.783	2.756
2013	16.673	19.642	28.588	19.038	15.846	2.500	2.366
2014	15.510	18.846	25.113	16.923	15.529	2.167	2.467
2015	23.019	27.135	35.558	23.189	19.843	2.091	2.957
2016	23.333	29.192	38.462	24.808	23.077	2.190	2.105
2017	14.220	16.788	25.192	16.077	14.039	1.696	1.667
2018	13.654	19.059	24.712	18.635	15.235	2.586	2.143
2019	11.263	15.872	24.600	15.026	13.795	1.923	1.500

3.4.1 Event Sentiment Scores

To estimate the potential economic impact upon news arrival and afterwards, Ravenpack assigns each piece of news an **Event Sentiment Score** (ESS) between 0 and 100 using a proprietary algorithm which combines results from surveying financial experts and pattern matching. An ESS of 100 indicates extreme positive short-term financial or economic impact. In contrast, an ESS with zero value indicates extreme negative impact.

And a ESS of 50 indicates exact neutral news, which is regarded as noise in this paper.

Raw scores (range from 0 to 100) are calibrated by subtracting 50, so that positive (negative) news items always have positive (negative) score and a zero score represents a neutral news.

The first panel in Figure 3.7 plots the distribution of (normalized) ESS for all news about crude oil, while second and third panels focus on two tails of the distribution. From the histogram in Figure 3.8, one can see that only a small portion of news is purely neutral with zero ESS (3,479 news items, 3.25 % of all news). Moreover, ESS scores of most news are clustered around -15 (39,347 news items, and 36.8 % of all news) and 18 (34,574 news items and 32.3 % of all news). Analyzing contents of these news suggests they are simply objective reports of past price/return movements of crude oil commodities and futures. Therefore, I do not expect these news to provide as much information on predicting returns as other breaking news like OPEC export restrictions. In order to emphasize "fresh" events other than reports of past price movements, models proposed in this paper will focus on extreme events by assigning them higher weights. Specifically, models are designed to pay more attention to news carrying sentiment scores of high absolute values, meanwhile, models would discriminate news whose sentiment scores are near zero.

Figure 3.7: Distribution of Event Sentiment Scores (ENS)

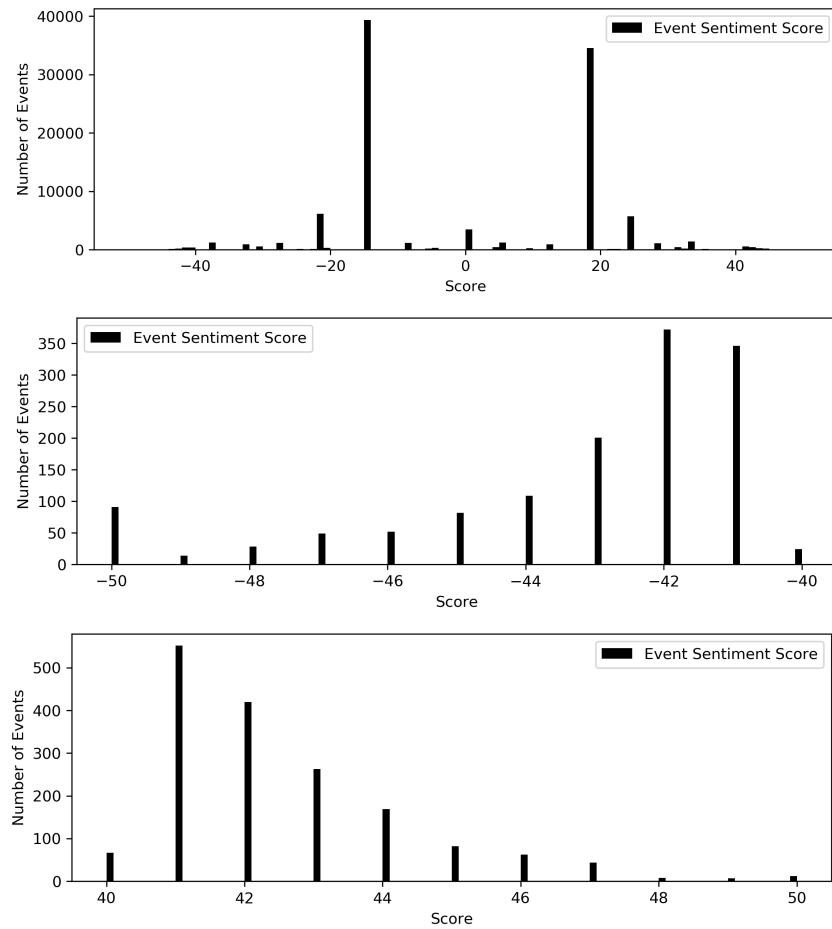
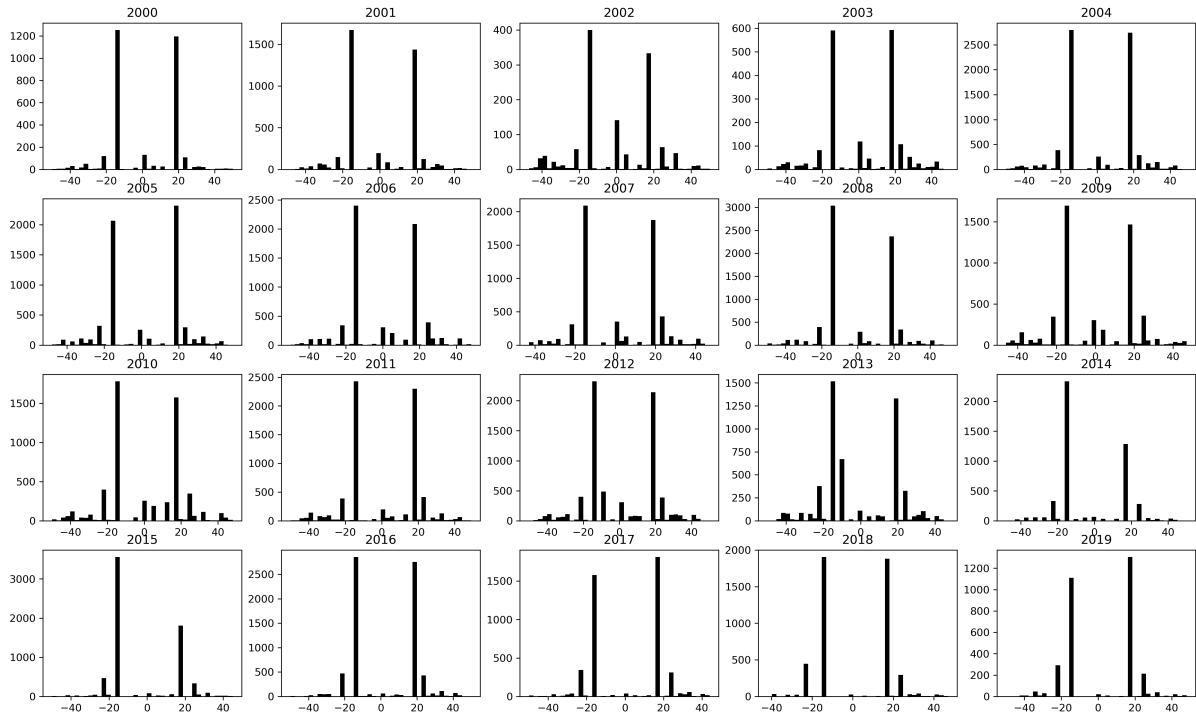


Figure 3.8 plots the distribution of ESS scores in each year. The pattern of clustering around -15 and 18 is pretty consistent over the span of 20 years: the majority of news are simply reporting the crude oil market instead of events outside the market.

Figure 3.8: Distribution of Event Sentiment Scores (ENS) each Year



3.4.2 Weighted Event Sentiment Scores

Different news sources can report the same event, so there are duplicate entries about the same event in this dataset, which aggravates the problem of noise. The number of news published can be much higher than the number of events actually happened. RPNA dataset computes an **Event Novelty Score** (ENS) to measure how novel a news story is within a 24-hour period.

Suppose that OPEC announces an export resection after a conference finished at 11:00 a.m. January 10. After 30 minutes (11:30 a.m., January 10) one news source reports this export cut, and this news article is one entry in the dataset. To determine the ENS of this news article, the algorithm looks into the 24-hour period prior to the news arrival, that is, from 10:30 a.m. January 9 to 10:30 a.m. January 10. If there is no news about this export restriction in this period, then this news article is the first report of this export cut event and receives an ENS score of 100. In contrast, if there are another two articles about the same events published before this articles, this article is the third one

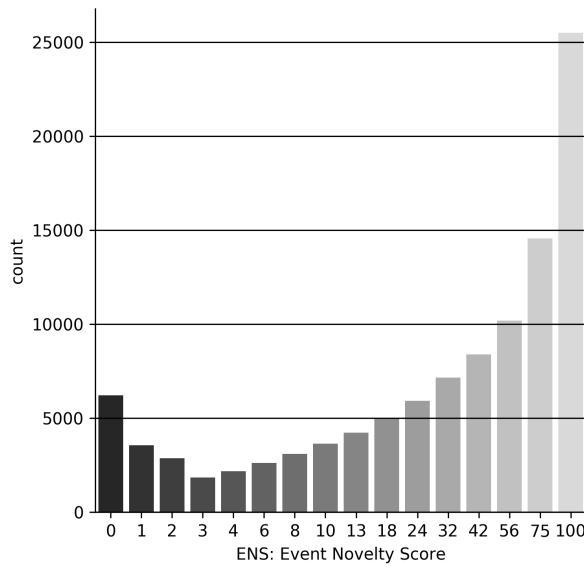
and receives a decayed novelty score of $100 \times 0.75 \times 0.75 = 56$ instead.

In general, the ENS decays exponentially as there are more news reporting the event. For an arbitrary news article i , if there are another k articles of the same topic published within the 24-hour period before article i arrives, article i is therefore the $k + 1^{th}$ articles on this topic and would receive a novelty score of

$$\text{ENS}_i = 100 \times 0.75^k \quad (3.4.1)$$

Figure 3.9 plots the distribution of ENS, the histogram suggests that most news have relatively high novelty scores.

Figure 3.9: Distribution of Event Novelty Score



To address the duplication issue, this paper constructs an alternative metric of sentiment, **Weighted Event Sentiment Score** (WESS), from both ESS and ENS.

$$\text{WESS} := \frac{\text{ESS} \times \text{ENS}}{100} \quad (3.4.2)$$

We divide the product of ESS and ENS in equation (3.4.2) by 100 so that WESS ranges from -50 to 50 as well.

The constructed WESS scores have several advantages for modelling. Using the original

ESS, one piece of news receives an ESS close to zero (a nearly neutral score) only if the event is reports has little economic impact. A sentiment score near zero indicates that this piece of news only contains little predictive information for forecasting. In contrast, as one could expect, the 10th news reporting a breaking (positive) would receive a high ESS and a lower ENS. However, such news comes too late and the market should have already reacted to the previous nine news about this event. As a result, the 10th news only contains little informative content for predicting oil returns. Consequently, ESS is essentially a biased predictor to proxy the impact of news on the market.

Unlike ESS, WEES discriminates against duplicate news articles by assigning them scores closer to zero. For example, if one extreme negative event happened, many sources report this event within 24 hours after it happened. Simply totalling ESS of all these news would overestimate how bad the scenario is because the negative event only occurs once but it is reported for several times. Weighting ESS of articles using their novelty scores helps mitigate this problem so that WEES allows models to pay more attention on novel news rather than other ones. Secondly, WEES preserves the sign of ESS, so that an event carries positive sentiment, in terms of ESS, if and only if its WEES score is positive.

The histograms in Figure 3.10 illustrate the overall distributions of WEES as well as the two tails of it. It turns out that the clustering pattern in Figure 3.7 disappears and much more news are now with zero sentiment scores. Therefore, WEES provides a stricter filter to filtering out noises (i.e., news with zero sentiment scores) and helps models better focus on meaningful news only.

Figure 3.10: Distribution of Weighted Event Sentiment Scores (WEES)

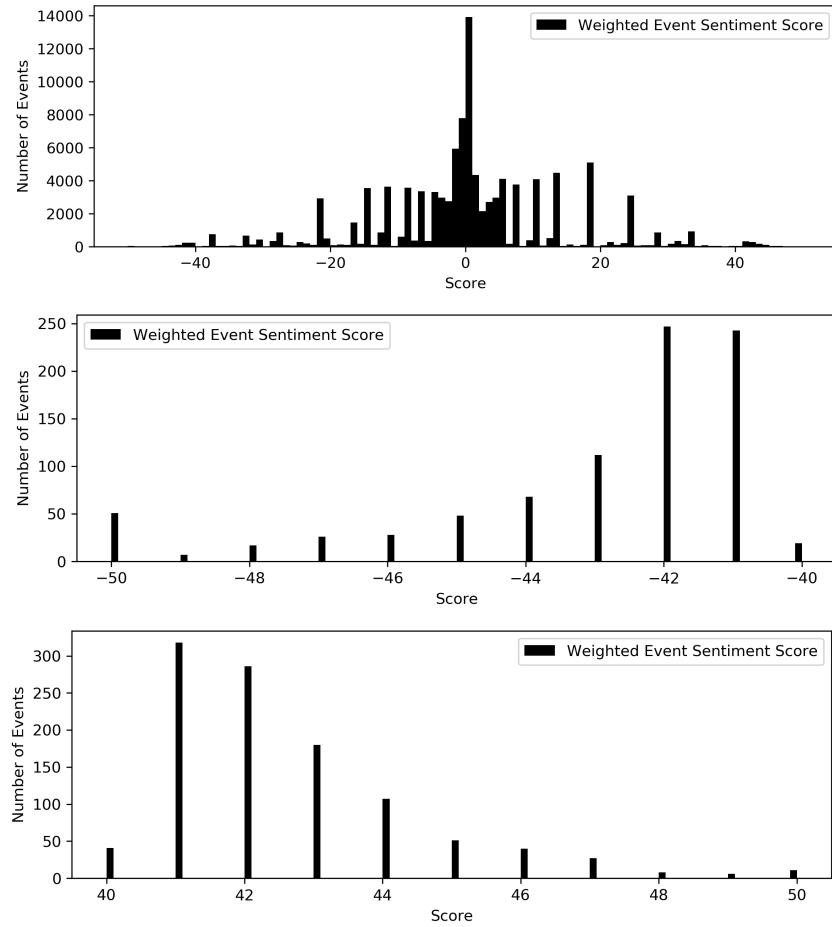
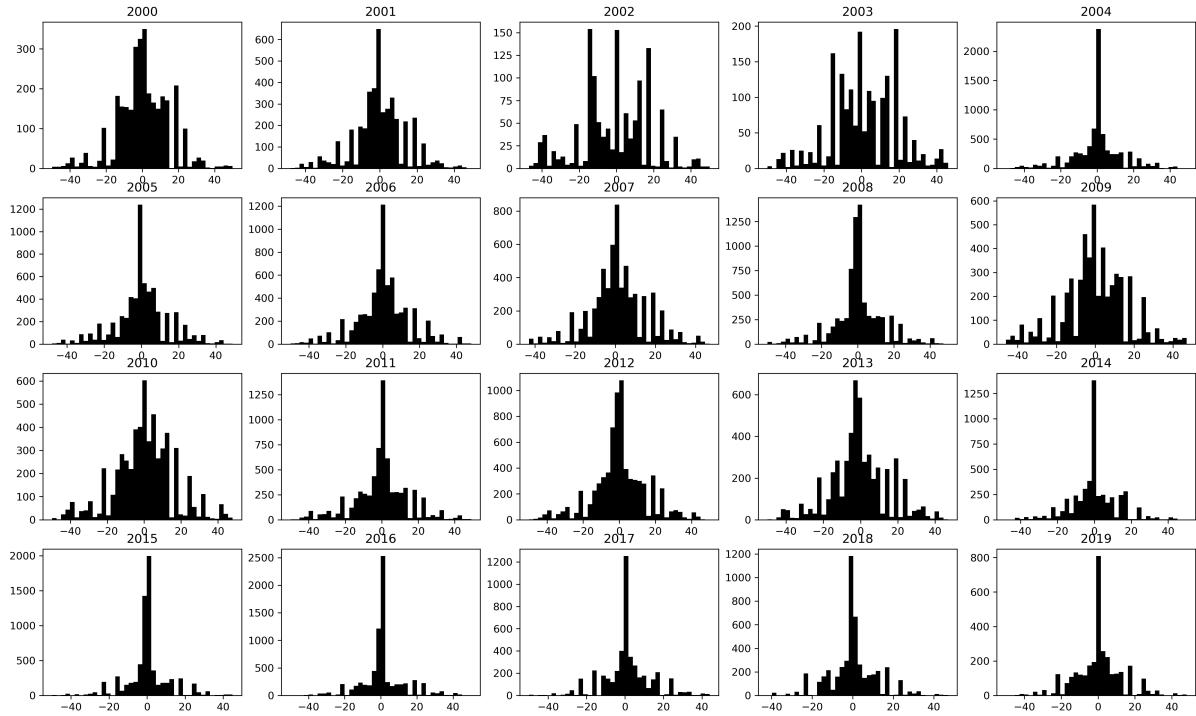


Figure 3.11 plots the yearly distributions of WEES. The yearly distributions suggest that there are more events with negative sentiments in 2001 as well as in 2008-2009, such observation matches the US recession records in Figure 3.1.

Figure 3.11: Distribution of Weighted Event Sentiment Scores (WESS) each Year



3.4.3 Time of News Arrival

The numbers of news arrived are not evenly distributed across the timeline, there are always busy hours as well as quiet hours. Figure 3.12 summarizes the average number of news arriving on each day over the period of 20 years. The trench at the end of February corresponds to leap years. Other trenches are in general correspond to holidays, for example, average numbers of news on the Independence Day and Christmas are significantly less than other days.

Figure 3.12: Average Number of Events on Each Day

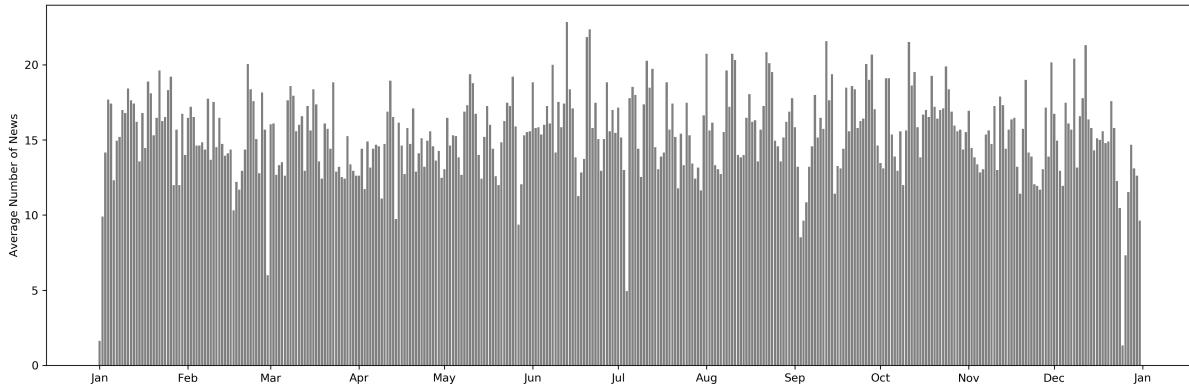
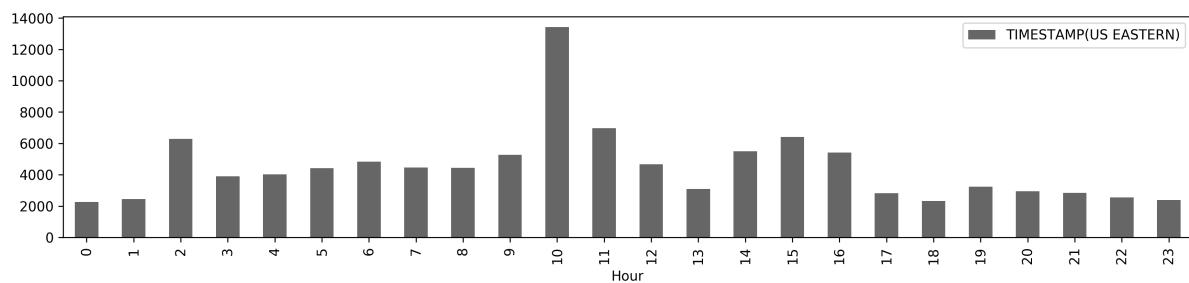


Figure 3.13 has a look at the distribution of news arrival within 24 hours. It is worth noticing that, in the RPNA dataset, the original timestamps recording when news arrives are using Universal Time Coordinated (UTC). To incorporate the crude oil dataset, we convert raw timestamps to Eastern Standard Time (EST) timezone³, where crude oil commodities are traded. From the distribution of news arrival, one can see that most news arrive during day time between 10:00 and 16:00. There is an unusual spike at 2:00, this could correspond to morning news at 7:00 in the UK. But because all four news sources in RPNA dataset are the US based publishers, the news arrival process is quiet again between 3:00 and 9:00 as less reporters are actively writing during this time.

Figure 3.13: Total Number of News Arrived

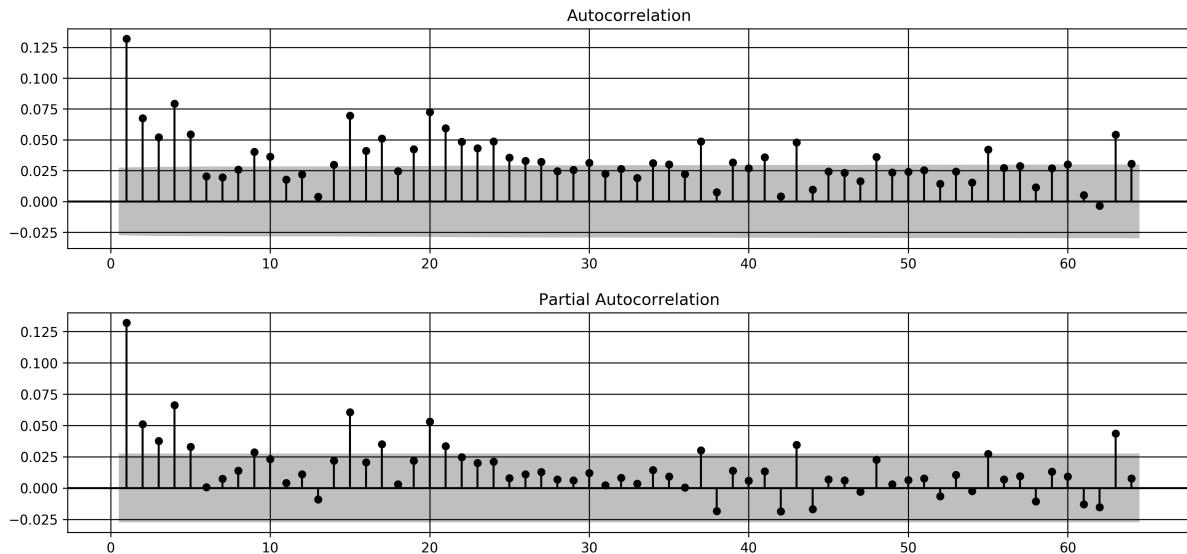


In order to have a closer look at the intertemporal correlation of event sentiment,

³EST is five hours behind UTC during autumn and winter. During spring and summer (daylight saving time), EST is four hours behind UTC.

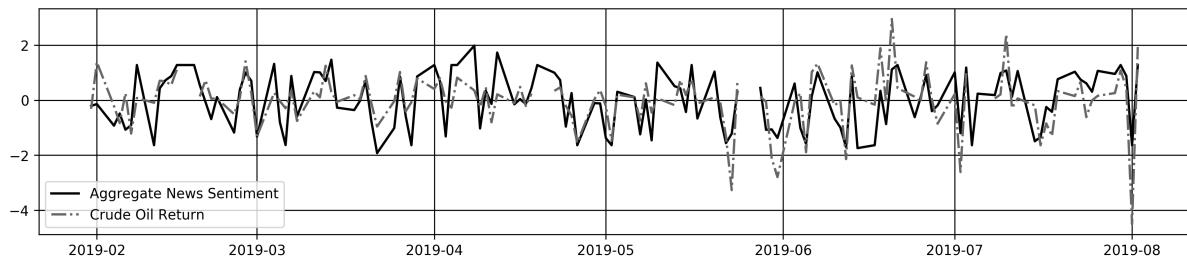
this paper firstly compute the mean event sentiment score of all events within each day, denoted as $\overline{\text{ESS}}$ and $\overline{\text{WEES}}$ respectively, in the span of 20 years. The ACF and PACF plots of the daily average ESS in Figure 3.14 suggest the intertemporal correlation here is much more salient than the series of returns, which has only a few significant lags.

Figure 3.14: ACF and PACF of $\overline{\text{ESS}}$



Moreover, there exists significant correlation between the price movement series and news sentiment. Figure 3.15 plots the trends of daily average sentiment and crude oil returns in 2019, in which these two series have shown significant co-movement pattern. It turns out that the Spearman correlation between these two series is 0.562 with p-value zero.

Figure 3.15: Movements of $\overline{\text{ESS}}$ and Return in 2019



This co-movement provides justification of using the series news sentiment to predict crude oil returns.

3.5 Classifying News Type

Based on the distributions shown in previous sections, we shall see that a great number of events carry nearly natural sentiment or are just description of past price movement. This paper wishes to allow models to differentiate different types of news instead of taking the average sentiment score of all news. As seen in the histograms of sentiment scores (figure 3.7 and 3.10), the distributions are pretty much symmetric about zero, therefore, for simplicity, this paper assumes the region of neutral news to be symmetric around zero. Specifically, the classification procedure firstly determines a radius $r \geq 0$. Afterward, the algorithm classifies all news based on their (weighted) event sentiment scores. News with score $(W)ESS \in [-50, -r]$ are negative news, and all news with $(W)ESS \in (r, 50]$ are positive news, and, news in $[-r, r]$ are neutral news. Figure 3.16 and Figure 3.17 plots the composition of news types while classifying these news using two criterions, event sentiment scores and weighted event sentiment scores.

Figure 3.16: Composition of News Type based on ESS

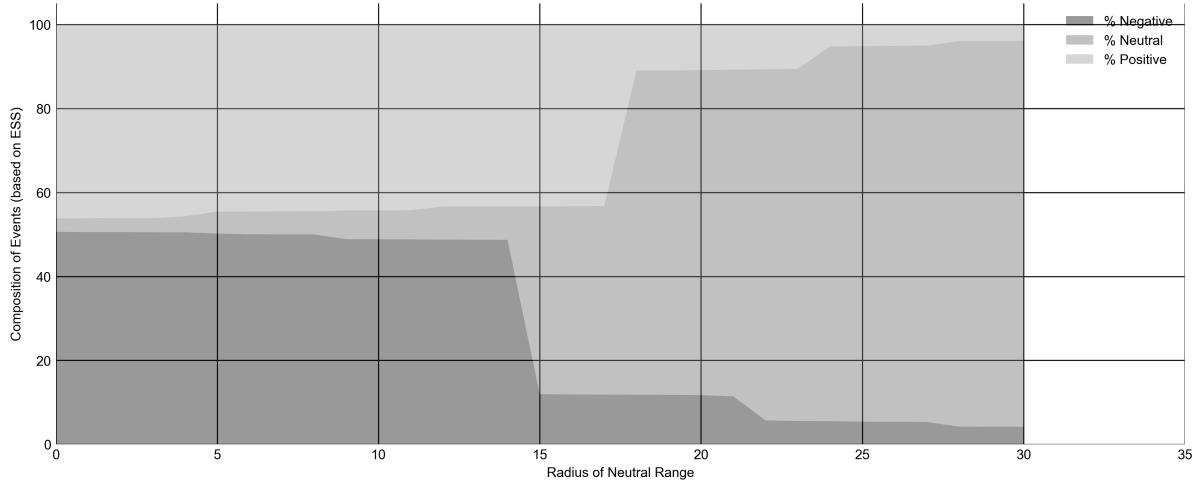
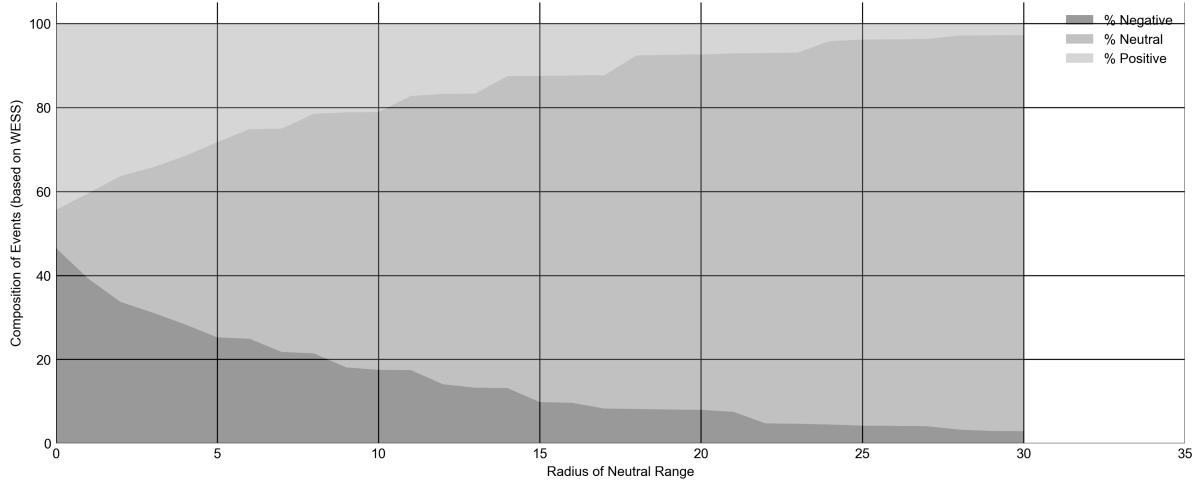


Figure 3.17: Composition of News Type based on WESS



In Figure 3.16, the two sharp breaking points at $r = 15$ and $r = 18$ correspond to the two clusters of events with sentiment scores 35 and 68 observed in Figure 3.7. Table 3.8 and Table 3.9 summarizes how the portions of different classes of news change while applying different value of threshold.

For example, if one decides to classify news based on their WESS scores and a radius $r = 0$. Then news with strictly positive (negative) WESS would be classified as positive (negative) and one piece of news is neutral only if it has exactly zero WESS. In this case, 46.54 % (44.4 %) of all news are classified as positive (negative) and the rest 9.06 % of news are neutral (the first row in Table 3.9). This paper refers this composition of news resulted from using $r = 0$ as the reference composition. Similarly, the classification criterion using WESS and $r = 10$ will firstly construct a closed interval $[-10, 10]$. Then any news with ESS scores fall in $[-10, 10]$ would be classified as neutral news and news with ESS scores less than -10 and greater than 10 are classified as negative and positive respectively. In this case, 17.51 % of news are classified as negative news, this percentage shrinks to $\frac{17.51\%}{46.54\%} = 37.64\%$ compared with the reference composition. In contrast, 61.39 % of news are now labelled as neutral news and this percentage is $\frac{61.39\%}{9.06\%} = 677.67\%$ of the neural percentage in the reference composition (the sixth row in Table 3.9).

In the appendix, Table 7.3 and 7.4 provide a more complete summary on the composi-

tion of news under various thresholds r . The threshold variable r is a hyper-parameter in our model, the optimal classification threshold depends on specific type of models used. In most experiments, we are using $r = 0$ for ESS scores and $r = 0.3$ for WESS scores.

Table 3.8: Composition of News Classes with Different Thresholds on ESS Scores

r	Num Negative	Num Neutral	Num Positive
0	50.59% (100.00%)	3.25% (100.00%)	46.15% (100.00%)
0.3	50.59% (100.00%)	3.25% (100.00%)	46.15% (100.00%)
1	50.57% (99.96%)	3.29% (101.24%)	46.13% (99.96%)
3	50.52% (99.85%)	3.39% (104.08%)	46.09% (99.87%)
5	50.20% (99.23%)	5.24% (161.14%)	44.55% (96.54%)
10	48.84% (96.53%)	6.91% (212.45%)	44.25% (95.88%)
15	11.93% (23.58%)	44.76% (1376.20%)	43.31% (93.83%)
20	11.73% (23.18%)	77.41% (2379.79%)	10.87% (23.55%)
25	5.41% (10.70%)	89.43% (2749.47%)	5.16% (11.17%)

Table 3.9: Composition of News Classes with Different Thresholds on WESS Scores

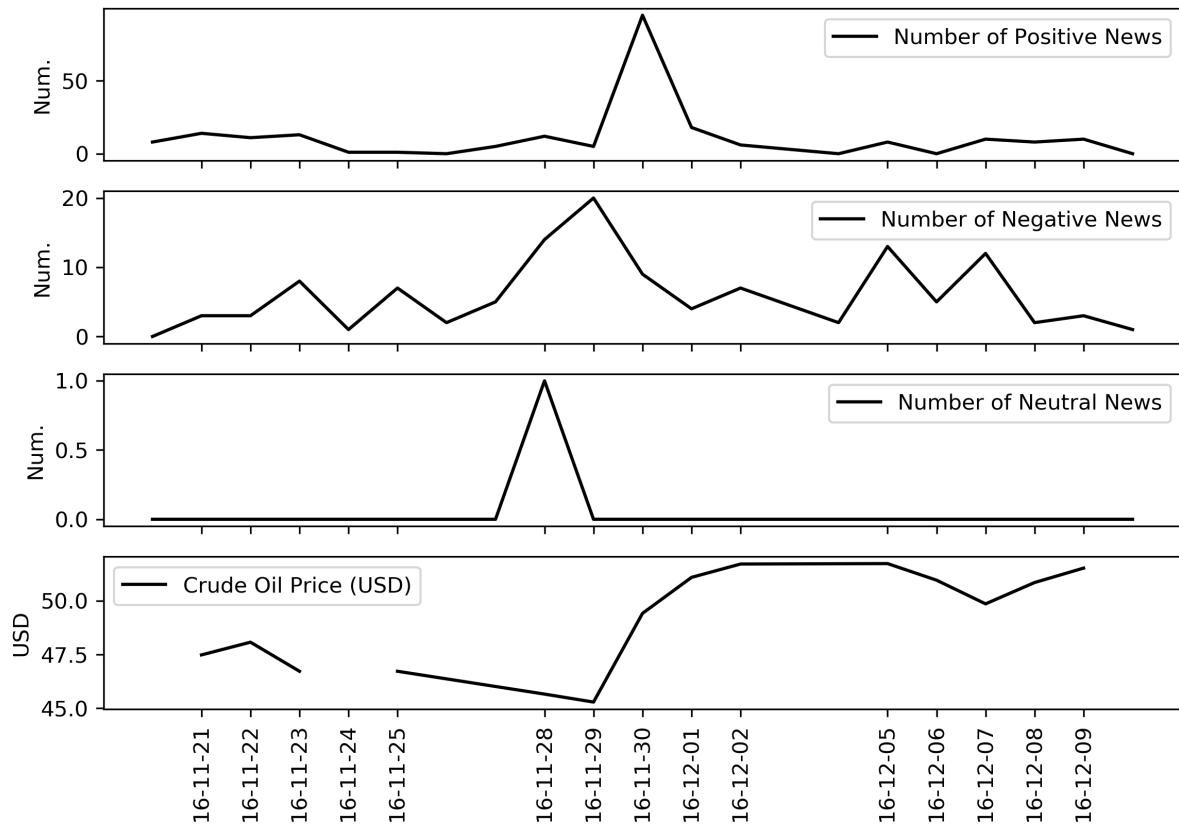
r	Num Negative	Num Neutral	Num Positive
0	46.54% (100.00%)	9.06% (100.00%)	44.40% (100.00%)
0.3	42.85% (92.08%)	13.99% (154.43%)	43.16% (97.19%)
1	39.25% (84.33%)	20.32% (224.32%)	40.43% (91.06%)
3	31.12% (66.87%)	34.62% (382.22%)	34.25% (77.14%)
5	25.24% (54.24%)	46.49% (513.20%)	28.27% (63.66%)
10	17.51% (37.64%)	61.39% (677.67%)	21.10% (47.52%)
15	9.83% (21.13%)	77.68% (857.58%)	12.48% (28.11%)
20	7.98% (17.15%)	84.63% (934.20%)	7.39% (16.65%)
25	4.22% (9.06%)	91.95% (1015.06%)	3.83% (8.63%)

3.6 Case Studies

3.6.1 November 30, 2016: Positive Spike

The first case study investigates the event of an expected production cut by OPEC. On the 30th of November, 2016. Reports concerning this shock were considered as positive news for crude oil price since upcoming negative supply shock generally leads to expectation on soaring prices.

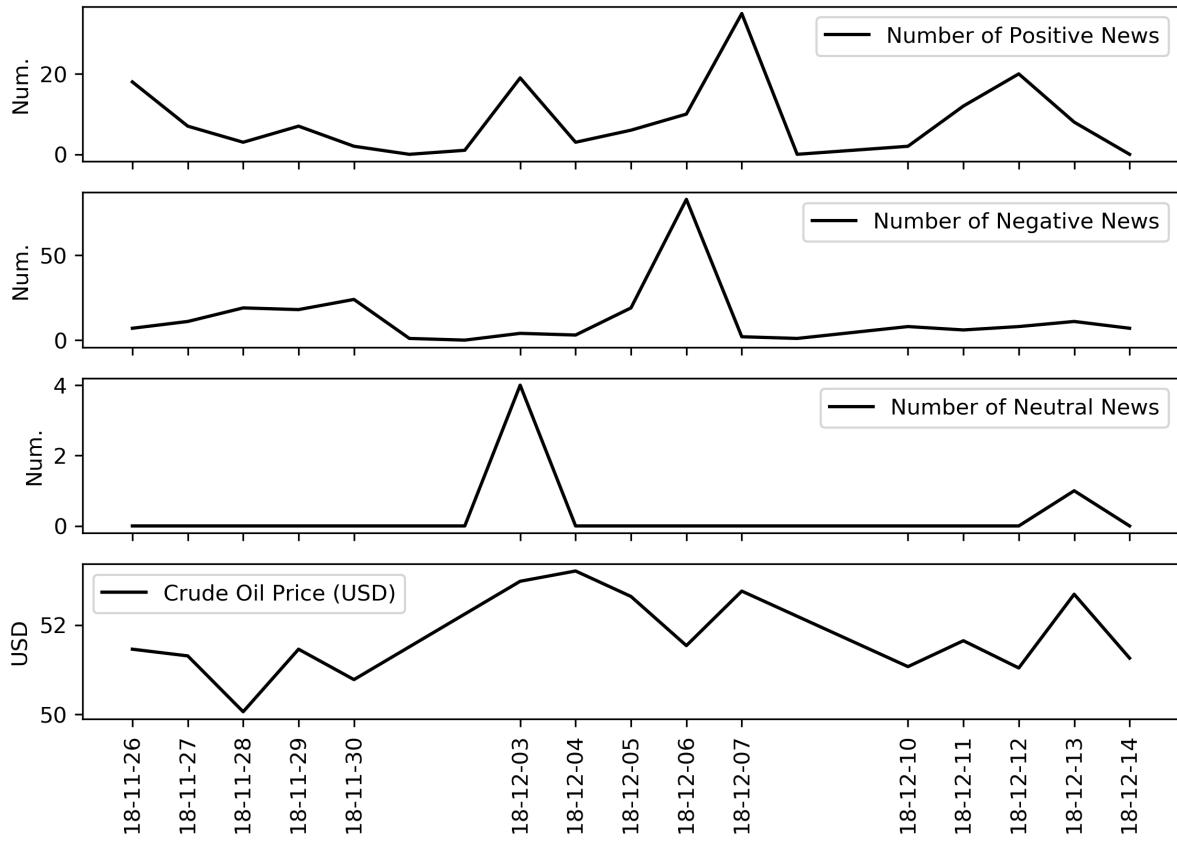
Figure 3.18: Crude Oil Price and Number of Events within 10 Days



3.6.2 December 6, 2018: Negative Spike

The US had become a net oil-exporting country in the week of Dec. 6, for the first time in 75 years (citation: Bloomberg). This major shift marks a potential negative shock in the demand side of the crude oil market and news reporting this fact was all considered as negative events for the crude oil price.

Figure 3.19: Crude Oil Price and Number of Events within 10 Days



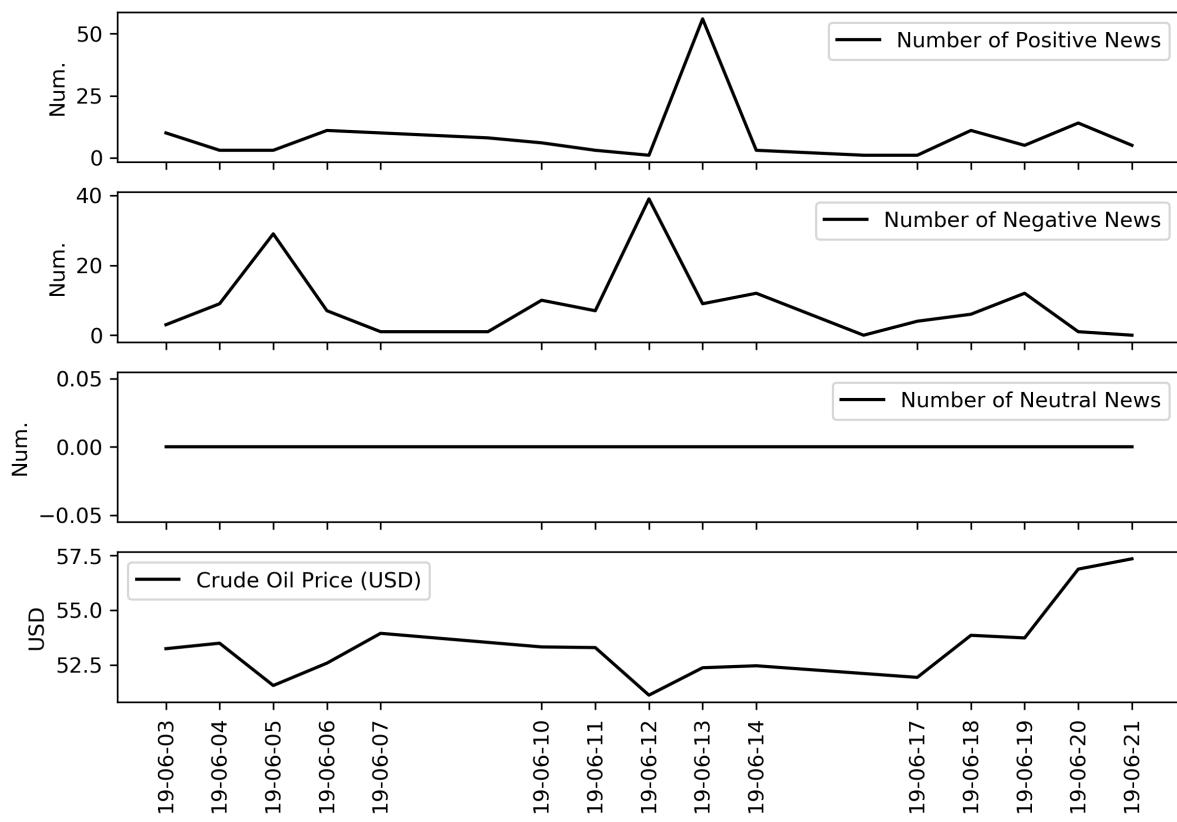
3.6.3 June 12~13, 2019: Positive Spike in Down Period

Unlike in the two previous cases, the third case investigates the impact of positive news spike in a period with falling oil prices. Table and Table in the appendix include more complete lists of news about crude oil published on Bloomberg on June 12 and 13, 2019.

The crude oil price had been decreasing since April 2019. The tension between the US and Iran had been accumulating since the US withdrew from the Joint Comprehensive Plan of Action on the 8th of May and alleged Iran for the first Gulf of Oman incident occurred on the 12th. In response, Iran had threatened to close the Strait of Hormuz, which is an important channel for international oil shipping. However, as we can see from the Figure 3.20, before Jun. 12, the major theme of news available was positive, this can be explained by the stable oil supply from Saudi Arabia and other oil-exporting countries.

The story changed on Jun. 13, when the second Gulf of Oman incident happened. Two oil tankers were attacked while passing the Gulf of Oman, which further escalated the tension between the US and Iran, and the market had sufficient reason to expect a negative supply shock. With the arrival of such a cluster of positive news (positive for crude oil prices), the price increased significantly after the incident but returned to its normal decreasing trend after approximately one week.

Figure 3.20: Crude Oil Price and Number of Events within 10 Days



Chapter 4

Model

4.1 Framework: An Example

Imagine there is a graduate student interested in crude oil market. Due to his limited budget, he has **subscriptions** to Wall Street Journal and the Economist only. For simplicity, assume he stays in his office the whole day and does not absorb any finance-related news from other sources.

One day, denoted as day t , OPEC announces a cut in oil export during a conference, this conference and announcement contribute to ω_t . For now, assume nothing else happens on day t so that this event can be considered as the **state of world** on day t , denoted as ω_t .

After the announcement is made, both WSJ and the Economist report this export restriction. These news articles describe ω_t and constitute the **information flow** IF_t of this student on day t . Note that IF_t depends on and reflects ω_t but it is only a qualitative proxy since IF_t is essentially a batch of news articles.

If this student wishes to build a predictive model for the future return r_{t+1} based information flow he receives, IF_t must be quantified. He can quantify the information flow based on his own knowledges. For example, one way is to manually assign each article a sentiment score: assign 10 (-10) scores to all articles about events that could increase (decrease) oil returns (based on his knowledge). Then taking the average score of all articles in IF_t gives a real-number \mathbf{x}_t , which is a **quantitative representation**

of IF_t . The process of quantifying IF_t , which is a batch of news article, is called the **characteristic function**.

4.2 Formal Framework

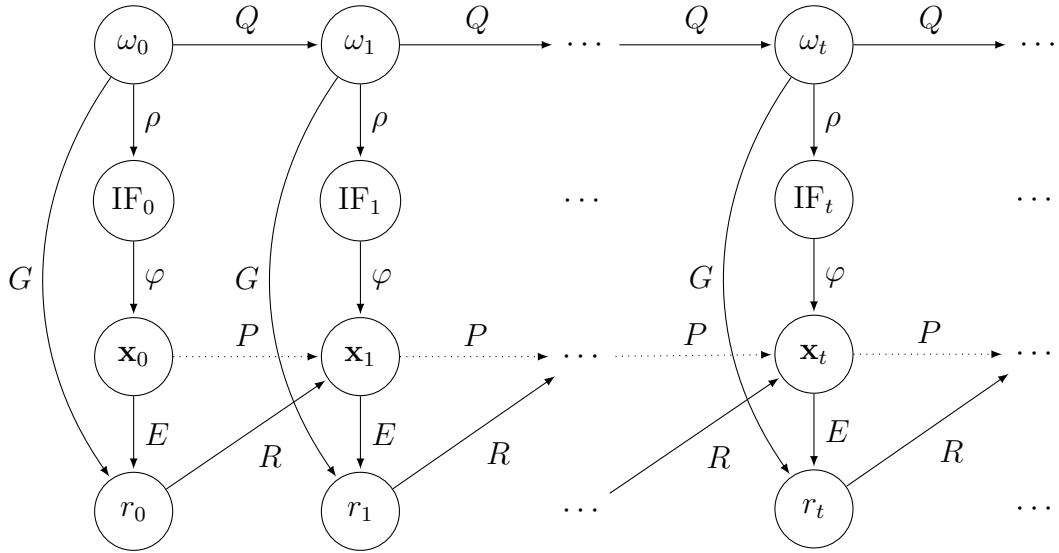
This section is devoted to describe the proposed framework but using formal mathematical language. Figure 4.1 illustrates the framework of the proposed framework.

4.2.1 Timestamps

In the following discussion, this paper uses non-negative real numbers, $t \in \mathbb{R}_+$, to represent continuous timestamps accurate to seconds. Specifically, 12 am of first day in dataset (January 3, 2000) corresponds to $t = 0$ and the length of 24 hours is normalized to one. Using this timestamp convention, an integer t indicates the beginning of the t^{th} trading day in the dataset. For example, the time stamp $t = 10$ represents 12:00 am of the 10^{th} trading day in our dataset, which was January 18, 2000. Similarly, $t = 10 + \frac{10}{24}$ denotes 10:00 am of January 18, 2000.

Each news article in the dataset has one timestamp τ corresponding to the time when this piece of news is published. Because this paper works on a daily basis prediction task, we need to discretize the continuous timestamp of t (accurate to seconds) into integers representing days. That is, the exact time (hours, minutes, and seconds) when a piece of news is published is discarded, everything happens within the time period $[t, t + 1)$ is considered as it happens on timestamp t exactly. In another word, interday effects are not considered by this paper.

Figure 4.1: The Framework



4.2.2 States of World

On each day $t \in \mathbb{Z}_+$, the real state of world is denoted as $\omega_t \in \Omega$, which is a latent variable describing everything happening in the world on day t .

The dynamics of $\{\omega_t\}$ is a stochastic process governed by a transition probability Q . The process evolves following equation (4.2.1):

$$\omega_t \sim Q(\omega_t | \omega_{t-1}, \omega_{t-2}, \dots, \omega_0) \quad (4.2.1)$$

4.2.3 Information Flow

The first component in this model consists of a batch of news subscriptions to various news sources. Each of these news sources summarizes ω_t as a collection of news articles, which consists of headlines, body texts, comments, etc. Some sources provide executive summary on these articles as well as analysis on the potential economic impacts from events mentioned in these articles.

We define the collection of news articles and other contents from the news provider to be the **information flow** generated from subscriptions on day t , denoted as $IF = \rho(\omega_t)$. The functional form suggests the information flow received by an individual depends on

both the state of world ω_t and another subscription function ρ characterizing how many resources this individual has access to. One individual with subscription function ρ_1 has access to more resources than another one with ρ_2 if

$$\rho_2(\omega) \subseteq \rho_1(\omega) \quad \forall \omega \in \Omega \quad (4.2.2)$$

For a fixed set of subscriptions ρ , the information flow can be defined for any time period.

Definition 4.2.1. For an arbitrary time period $\mathcal{T} \subseteq \mathbb{R}_+$, the **information flow** of time period \mathcal{T} , denoted as $\text{IF}_{\mathcal{T}}$, is simply the collection of news articles published and their corresponding ESS, ENS, WEES (also include other information provided by RPNA dataset associated with these articles) within this time period.

In this study, our subscription function ρ is defined by the RPNA dataset, which consists of four sources: Dow Jones Newswires, Wall Street Journal, Barron's, and MarketWatch. So the information flow used in this paper is the collection of news articles and relevant analysis from the above-mentioned four sources.

Let $N = 106,960$ denote total number of news articles about crude oils ¹ in the RPNA dataset over the considered period from January 1, 2000 to October 31, 2019. One may firstly sort all N news based on the timestamp when each news arrived. Then each piece of news in the dataset can be uniquely indexed using an integer $n \in \{1, 2, \dots, N\}$. For example, news article n is the n^{th} news in the dataset. Let τ_n denote the time when the n^{th} news article was reported.

Using this indexing method, news arrives within a time period \mathcal{T} can be presented as a set of integers $\theta_{\mathcal{T}}$.

$$\theta_{\mathcal{T}} = \{n \in \{1, 2, \dots, N\} : \text{the } n^{th} \text{ news arrives within time period } \mathcal{T}\} \quad (4.2.3)$$

¹Recall that RPNA dataset includes a topic for each article, the topic of article is identified based on the headline, keywords and body text of the news article.

The corresponding information flow can be expressed as in equation (4.2.4).

$$\text{IF}_{\mathcal{T}} = \{n : n \in \theta_{\mathcal{T}}\} \quad (4.2.4)$$

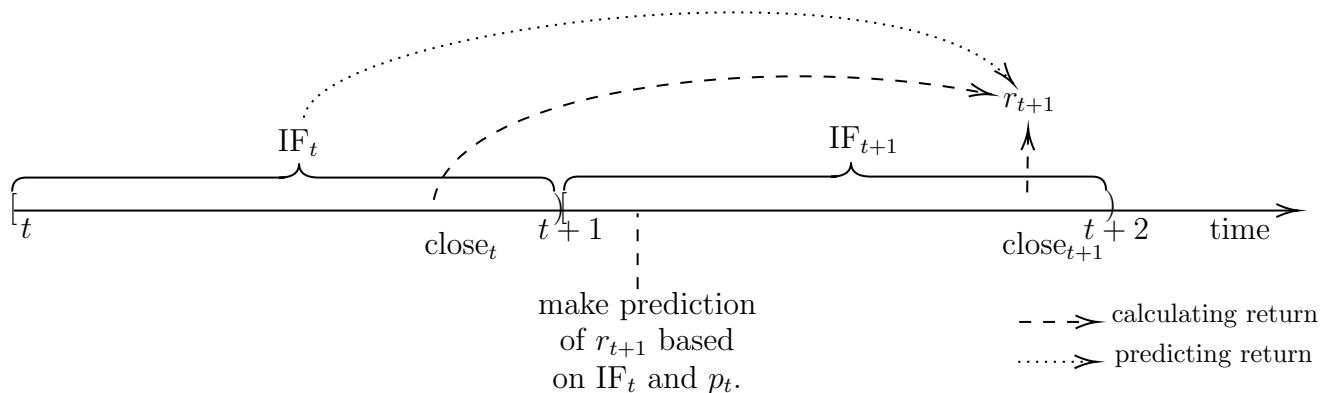
One special case is when the period of consideration is precisely one day, that is $\mathcal{T} = [t, t + 1]$. This paper introduces another definition for this special case.

Definition 4.2.2. The **information flow on day t** , IF_t , is defined to be the set of indices of news articles in the dataset that are published on day t :

$$\text{IF}_t = \{n : n \in \theta_{[t, t+1]}\} \quad (4.2.5)$$

Figure 4.2 summarizes the workflow of the prediction task in a minimal setting: predicting r_{t+1} using information on the day t only. On the time line, close_t and close_{t+1} indicate the time when the closing prices are computed. At the beginning of day $t + 1$, say 1 a.m., the prediction task is to predict the return r_{t+1} of day $t + 1$ from both p_t and IF_t using a predictive model. Let $\hat{r}_{t+1}(p_t, \text{IF}_t)$ denote the prediction for r_{t+1} , and traders may design their trading strategy on day $t + 1$ based on \hat{r}_{t+1} . Later when the market closes in the afternoon of day $t + 1$ and p_{t+1} is realized. The real return r_{t+1} can be computed from p_{t+1} and p_t using equation (3.2.1). Comparing \hat{r}_{t+1} and r_{t+1} can evaluate the predictive power of the model used.

Figure 4.2: Workflow of the Prediction Task



4.2.4 Characteristic Functions for Feature Construction

In this paper, we approach the prediction task using auto-regressive models, which means we are using lagged values to predict the future. Let \mathcal{M} denote a predictive model uses information in the past to predict the future crude oil returns.

For instance, if one is building a model conducting one step ahead forecasting based on historical information (both information flow of past days and historical returns) up to ℓ days in the past, \mathcal{M} is a map from information in the past ℓ days to a prediction:

$$\mathcal{M} \left(\begin{bmatrix} \text{IF}_{t-\ell+1} \\ r_{t-\ell+1} \end{bmatrix}, \begin{bmatrix} \text{IF}_{t-\ell+2} \\ r_{t-\ell+2} \end{bmatrix}, \dots, \begin{bmatrix} \text{IF}_t \\ r_t \end{bmatrix} \right) = \hat{r}_{t+1} \quad (4.2.6)$$

Then the model \mathcal{M} is evaluated based on how close r_{t+1} and \hat{r}_{t+1} are. However, each information flow IF_t is a set of (indices of) news articles, these indices remain abstract and do not have any meaning themselves. Simply feeding these indices to a machine learning model will not generate any meaningful result, one needs to extract some features from those news articles for predictive models. In order to quantify these abstract information flow, we propose a mapping called characteristic function.

Let \mathcal{T} be an arbitrary time period, we define a characteristic function φ ² as a mapping from the information flow of time period \mathcal{T} to a real-valued vector $\varphi(\text{IF}_{\mathcal{T}}) \in \mathbb{R}^d$, where d denotes the number of features constructed. Ideally, $\varphi(\text{IF}_{\mathcal{T}})$ should provide a quantitative summary from various aspects of news articles in $\text{IF}_{\mathcal{T}}$.

For example, one valid characteristic function φ can construct the following two features: 1) the the number of news articles in $\text{IF}_{\mathcal{T}}$ and 2) the average event sentiment score of these news:

$$\varphi(\text{IF}_{\mathcal{T}}) = \begin{bmatrix} \text{Number of News on Day } t \\ \text{Average ESS Score of News on Day } t \end{bmatrix} \in \mathbb{R}^2 \quad (4.2.7)$$

Note that in subsequent sections, the characteristic function actually used has far more

²In the context of probability, the characteristic function of a distribution fully describes the distribution, refer to (Ushakov 1999) for a review of this topic. Here we define the characteristic function to be the function mapping a collection of news to a vector \mathbf{x} of summary statistic of these news.

features than the example in equation (4.2.7).

With a chosen characteristic function φ , one can quantify the qualitative information flow on each day t using a real-valued vector \mathbf{x}_t :

$$\mathbf{x}_t := \varphi(\text{IF}_t) \quad (4.2.8)$$

After the prediction of r_{t+1} is made using \mathbf{x}_t (and p_t as well), one may design and modify trading strategy used on day $t + 1$ based on the prediction of r_{t+1} . All of these things happen at the beginning of day $t + 1$ before the market opens.

The complete workflow of forecasting and model evaluation can be summarized as

- (i) (At the beginning of day $t + 1$) gather all news articles published on and returns of previous days: $\text{IF}_{t-\ell}, \dots, \text{IF}_t$ and $r_{t-\ell}, \dots, r_t$.
- (ii) (At the beginning of day $t + 1$) quantify news articles gathered in the previous step into $\mathbf{x}_{t-\ell}, \dots, \mathbf{x}_t$.
- (iii) (At the beginning of day $t + 1$) plug $r_{t-\ell}, \dots, r_t$ and $\mathbf{x}_{t-\ell}, \dots, \mathbf{x}_t$ into a predictive model \mathcal{M} to generate prediction \hat{r}_{t+1} .
- (iv) (While market is opening on day $t + 1$) trade based on the prediction \hat{r}_{t+1} .
- (v) (When market closes on day $t + 1$) compute the actual return r_{t+1} using p_t and p_{t+1} .
- (vi) (When market closes on day $t + 1$) assess the performance of \mathcal{M} by comparing r_{t+1} and \hat{r}_{t+1} . Note that the model's performance can also be evaluated based on the profit made this day.
- (vii) Move to day $t + 1$ and repeat the whole process.

In this paper, the characteristic function used provides a summary on the sentiment scores provided by RPNA dataset, therefore, we define \mathbf{x}_t to be the **sentiment** of the information flow on day t .

Using characteristic functions, the predictive model in equation (4.2.6) can be equivalently expressed as

$$\mathcal{M} \left(\begin{bmatrix} \varphi([t-\ell+1, t-\ell+2)) \\ r_{t-\ell+1} \end{bmatrix}, \begin{bmatrix} \varphi([t-\ell+2, t-\ell+3)) \\ r_{t-\ell+2} \end{bmatrix}, \dots, \begin{bmatrix} \varphi([t, t+1)) \\ r_t \end{bmatrix} \right) \quad (4.2.9)$$

$$= \mathcal{M}(\mathbf{x}_{t-\ell+1}, r_{t-\ell+1}, \mathbf{x}_{t-\ell+2}, r_{t-\ell+2}, \dots, \mathbf{x}_t, r_t) = \hat{r}_{t+1} \quad (4.2.10)$$

4.2.5 Gathering-First Paradigm and Summarizing-First Paradigm

Exchanging the order of applying characteristic function and aggregating information induces significant difference in the model. Note that we may aggregate information flow first, that is, take $\mathcal{T}' = [t-\ell+1, t+1]$. Then, we can construct a summary of all news arrive in the past ℓ days by applying φ on \mathcal{T}' :

$$\mathbf{x}_{t-\ell+1:t} = \varphi(\mathcal{T}') \quad (4.2.11)$$

Hence, $\mathbf{x}_{t-\ell+1:t}$ is a quantitative summary of all news arrive from day $t-\ell+1$ to day t .

Using this notation, the following alternative formulation is equally valid,

$$\mathcal{M}(\varphi([t-\ell+1, t+1]), r_{t-\ell+1}, r_{t-\ell+2}, \dots, r_t) \quad (4.2.12)$$

$$= \mathcal{M}(\mathbf{x}_{t-\ell+1:t}, r_{t-\ell+1}, r_{t-\ell+2}, \dots, r_t) = \hat{r}_{t+1} \quad (4.2.13)$$

In the following parts of this paper, we call equation (4.2.10) the **summarizing-first** formulation since it apply φ on news arrived in each day first, then feeds the collection of summaries to a predictive model. We call (4.2.13) the **gathering-first** formulation, since it firstly collects all news arrive in the time period of consideration, and apply the characteristic function on all these news.

There is a trade-off between choosing which formulation to use. For example, when $\ell = 10$, the summarizing-first paradigm generates one feature vector \mathbf{x} for each of the past 10 days of consideration. In total, the 10 feature vectors generated contains $10d$ real

values, where d is the dimension of φ 's codomain. However, suppose \mathbf{x} contains mean and standard deviation of ESS scores, each \mathbf{x} is only built from a few news arrive within one single day and information contained in \mathbf{x} could be biased. In contrast, the feature vector constructed using the gathering-first approach is built from much more news articles and can reflect the state of world more accurately as a result. The drawback of using the gathering-first formulation is that it generates too few features: no matter how large the scope ℓ is, the gathering-first approach only constructs one feature vector \mathbf{x} , which contains d real values.

To fully leverage information from the news articles, predictive models used in this paper are based on features constructed using both paradigms. That is, the model uses both ℓ daily summary feature vectors and one aggregate summary feature vector to forecast the future return r_{t+1} .

4.2.6 Intertemporal Dependency

The state of world is changing from ω_t to ω_{t+1} between two consecutive trading days, and ω_{t+1} depends on ω_t . Moreover, \mathbf{x}_t affects \mathbf{x}_{t+1} as well since certain type of events are reported continuously for more than one day, and the news sentiments exhibits intertemporal correlation. For example, export restrictions by OPEC countries is a negative news for crude oil prices. An article about this OPEC meeting is reported on the first day and an analysis report of this restriction is published on the second day.

Lastly, some news on day $t + 1$ are simply reporting the return on the previous day so that r_t impacts \mathbf{x}_{t+1} as well.

4.3 Empirical Model

The empirical models to be estimated in this paper are focusing on the dynamics of sentiments $\{\mathbf{x}_t\}_t$ and returns $\{r_t\}_t$.

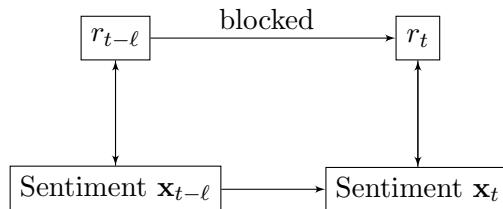
As examined in the section 3.2, the intertemporal correlation among returns is weak, hence, predicting current r_t using lagged value of returns is far too challenging for simple models. The framework proposed by this paper aims to use series of sentiments $\{\mathbf{x}_t\}_t$

constructed, which have stronger intertemporal correlation, to bridge the gap.

Specifically, in Figure 4.1, each pair of \mathbf{x}_{t-1} and \mathbf{x}_t are correlated through the chain $\mathbf{x}_{t-1} \rightarrow \omega_{t-1} \rightarrow \omega_t \rightarrow \mathbf{x}_t$. Therefore, in order to model the dynamic from \mathbf{x}_{t-1} to \mathbf{x}_t , we have to construct models estimating edges P , E , and R in the graph first. Note that each of these edges is a probability distribution.

Figure 4.3 illustrates the idea of forecasting returns via sentiments. Suppose we wish to predict returns r_t on day t using only information at time $t - \ell$ (i.e., both $r_{t-\ell}$ and $\mathbf{x}_{t-\ell}$) for some integer $\ell > 1$. The autocorrelation function (ACF) and partial autocorrelation function (PACF) of return series have only a few significant lags, so that the arrow from $r_{t-\ell}$ to r_t is blocked by this weak correlation. This prevents one from forecasting r_t using $r_{t-\ell}$ with directly a simple model like autoregression integrated moving average (ARIMA). However, the strong intertemporal correlation in sentiment series allows one to predict \mathbf{x}_t using $\mathbf{x}_{t-\ell}$. Secondly, the correlation between sentiment and return enables the model to estimate r_t from the prediction of \mathbf{x}_t . The composite of two steps above provides an indirect approach of forecasting r_t using information at time $t - \ell$.

Figure 4.3: Framework



Literature have been using hidden Markov models (HMMs) for time series forecasting. As mentioned before, \mathbf{x}_{t+1} are determined by the collection of news on day $t + 1$ (the information flow) via a characteristic function. However, many of news articles in IF_{t+1} are simply reporting past price movements of crude oil, that is, r_t . Therefore, the proposed framework extends the hidden Markov framework by allowing the directed edge from r_t to \mathbf{x}_{t+1} , the edge R , to explicitly model the impact of historical price movements on future news sentiment.

We model $\{\mathbf{x}_t\}_t$ as a stochastic process whose dynamics is governed by the **transition**

probability, P , and the **reporting probability**, R , plus random noises:

$$\mathbf{x}_t = \mathbf{x}_t^A + \mathbf{x}_t^B + \varepsilon_t \quad (4.3.1)$$

$$\text{where } \mathbf{x}_t^A \sim P(\mathbf{x}_t^A | \mathbf{x}_{t-1}) \quad (4.3.2)$$

$$\mathbf{x}_t^B \sim R(\mathbf{x}_t^B | r_{t-1}) \quad (4.3.3)$$

The transition probability P models the impact of past news sentiments on the future news sentiment, and \mathbf{x}_t^A is the portion of sentiment \mathbf{x}_t solely determined by the intertemporal correlation among \mathbf{x} . In addition, another reporting probability R models the impact of past returns on future news sentiment. Hence, \mathbf{x}_t^B is the part of \mathbf{x}_t responses to past price movement. For example, \mathbf{x}_t^B could be the average of sentiment scores assigned to articles simply reporting the return on the previous day. More generally, two parts of news sentiments can be merged using one joint distribution PR :

$$\mathbf{x}_t \sim PR(\mathbf{x}_t | \mathbf{x}_{t-1}, r_{t-1}) \quad (4.3.4)$$

Expanding equation (4.3.4) recursively shows that \mathbf{x}_t is in fact impacted by all historical values of \mathbf{x}_t and r_t . Therefore, the entire history of $\{\mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \dots, \mathbf{x}_0\}$ and $\{r_{t-1}, r_{t-2}, \dots, r_0\}$ contribute to the distribution of \mathbf{x}_t

$$\mathbf{x}_t \sim F(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \dots, \mathbf{x}_0, r_{t-1}, r_{t-2}, \dots, r_0) \quad (4.3.5)$$

The **order** of a Markov model determines the length of its memory, a Markov model has order ℓ if the distribution of an arbitrary random variable Y_t only depends on the past ℓ values, that is, for every $t > \ell$,

$$P(Y_t | Y_{t-1}, Y_{t-2}, \dots, Y_0) = P(Y_t | Y_{t-1}, Y_{t-2}, \dots, Y_{t-\ell}) \quad (4.3.6)$$

In most cases, the impact of observations in the distant past, say \mathbf{x}_{t-1000} , on the current observation \mathbf{x}_t is negligible. Therefore, for simplicity, we assume the chain in equation

(4.3.5) is assumed to have a finite order ℓ . Hence,

$$\mathbf{x}_t \sim F(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \dots, \mathbf{x}_{t-\ell}, r_{t-1}, r_{t-2}, \dots, r_{t-\ell}) \quad (4.3.7)$$

As mentioned before, the actual return r_t is determined by multiple factors. Firstly, all those events happening on day t , that is, ω_t affects r_t . Moreover, traders are often trade according to news articles (\mathbf{x}_t) they have read, and traders' behaviours affect the market and therefore r_t . Hence, \mathbf{x}_t is affecting r_t indirectly via traders' behaviours as well. The distribution of r_t follows distribution E termed **emission probability**.³ In particular, E maps the current state of world ω_t and current news sentiment \mathbf{x}_t to a distribution of realized returns r_t . Hence, the distribution of r_t depends on both the true state of world and the news sentiment.

$$r_t \sim E(r | \omega_t, \mathbf{x}_t, r_{t-1}, r_{t-2}, \dots, r_{t-\ell}) \quad (4.3.8)$$

For generality, even not shown in Figure 4.1, we assume the distribution of r_t depends on historical returns too. Therefore, the distribution in equation (4.3.8) includes the lagged values of returns as well. However the impact of adding historical returns should be insignificant given previous analysis on autocorrelations. Moreover, since we assume the Markov chain of returns has order ℓ , so that historical return values before $r - \ell$ are discarded.

Note that the series of $\{\omega_t\}$ is latent, the model has only observations on $\{\mathbf{x}_t\}$ and $\{r_t\}$. For each day t , we can now construct two predictors of return, r_t ,

$$\hat{r}_t^{\text{raw}} = \mathbb{E}[r_t | \Omega_{\text{partial}}] \quad (4.3.9)$$

$$= \mathbb{E}[r_t | r_{t-1}, r_{t-2}, \dots, r_{t-\ell}] \quad (4.3.10)$$

$$= \mathcal{M}^{\text{raw}}(r_{t-1}, r_{t-2}, \dots, r_{t-\ell}) \quad (4.3.11)$$

³In this paper, E is a specific distribution and \mathbb{E} denotes the expectation operator.

$$\hat{r}_t^{\text{senti}} = \mathbb{E}[r_t | \Omega_{\text{complete}}] \quad (4.3.12)$$

$$= \mathbb{E}[r_t | \mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \dots, \mathbf{x}_{t-\ell}, r_{t-1}, r_{t-2}, \dots, r_{t-\ell}] \quad (4.3.13)$$

$$= \mathcal{M}^{\text{senti}}(\mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \dots, \mathbf{x}_{t-\ell}, r_{t-1}, r_{t-2}, \dots, r_{t-\ell}) \quad (4.3.14)$$

The first model \mathcal{M}^{raw} is predicting the return following a classical auto-regressive manner, that is, using lagged values of return only. In contrast, the second model $\mathcal{M}^{\text{senti}}$ incorporates the sentiment series as well.

Using this framework, we may formulate the original question of interest as whether \hat{r}_t^{senti} is a significantly better prediction of r_{t+1} than \hat{r}_t^{raw} , in terms of prediction accuracy. Recall the workflow in Figure 4.2, since the closing price p_t is realized near the end of day and the return r_t can be computed at the same time. Based on Figure 3.13, most information within IF_t will arrive before r_t is realized, if the market is efficient, a great portion of IF_t should have already been reflected in r_t . In this case, IF_t , and therefore \mathbf{x}_t , will not provide meaningful information to the prediction of r_t . Therefore, if the efficient market hypothesis holds true, the performance of two above-mentioned predictions should be similar. And we can conclude that adding the news sentiment series does not help predict returns. In contrast, if $\mathcal{M}^{\text{senti}}$ outperformed \mathcal{M}^{raw} a lot, then we can claim that the news sentiment series is capable of improving return predictions.

In experiment section of this paper, this paper uses different types of statistical learning models to estimate \hat{r}_t^{raw} and \hat{r}_t^{senti} , then compare their performances.

Chapter 5

Experiments

5.1 Procedures

The empirical model in section 4.3 gives a brief description on how to assess the predictive power of models with and without sentiment dataset. We need to choose a specific characteristic function φ and predictive model \mathcal{M} in order to generate quantitative metrics and compare performances.

5.1.1 Feature Constructions

The previous section described the rough idea of using a characteristic function to quantify an information flow. Before implementing any statistical model, the first step is to choose a specific characteristic function which extracts quantitative summary statistics from a given information flow. In order to maximize the number of features extracted, the proposed characteristic function utilizes features from both the gathering first and summarizing first paradigms.

Let $\ell \in \mathbb{Z}_+$ denote the length of model's memory. That is, while predicting r_t , the model can only use information from day $t - \ell$ to day $t - 1$. This paper chooses $\ell = 31$ so that the model uses information within a whole month to predict one return.

Firstly, for each day from day $t - \ell$ to day $t - 1$, the characteristic function computes all variables in Table 5.1. Even though `NUM_EVENTS` can be deduced from `ESS_MEAN` and

`ESS_TOTAL`, we decided to include it as a proxy of the volatility of news networks.

Table 5.1: Daily Summary Statistics from Summarizing-First Paradigm (1)

Code Name	Variable	Code Name	Variable
<code>ESS_MEAN</code>	Average ESS	<code>WESSION_MEAN</code>	Average WESSION
<code>ESS_TOTAL</code>	Sum of ESS	<code>WESSION_TOTAL</code>	Sum of WESSION
<code>NUM_EVENTS</code>	Number of Events		

Moreover, following the methodology in section 3.5, the characteristic function classifies positive (/ negative / neutral) news using their ESS (WESSION) scores and a predefined threshold r . The characteristic function added the number of news in each class to the daily summary.

Table 5.2: Daily Summary Statistics from Summarizing-First Paradigm (2)

Code Name	Variable	Code Name	Variable
<code>NUM_POSITIVE_ESS</code>	# news s.t. $\text{ESS} > r$	<code>NUM_POSITIVE_WESSION</code>	# news s.t. $\text{WESSION} > r$
<code>NUM_NEGATIVE_ESS</code>	# news s.t. $\text{ESS} < -r$	<code>NUM_NEGATIVE_WESSION</code>	# news s.t. $\text{WESSION} < -r$
<code>NUM_NEUTRAL_ESS</code>	# news s.t. $\text{ESS} \in [-r, r]$	<code>NUM_NEUTRAL_WESSION</code>	# news s.t. $\text{WESSION} \in [-r, r]$

Table 5.1 and Table 5.2 together provide the daily summary for the sentiment dataset from day $t - \ell$ to day $t - 1$, and concatenating them gives 11ℓ features in total. We denote the characteristic function computing daily summary as φ_{daily} , for a given information flow on day t , $\varphi_{\text{daily}}(\text{IF}_t)$ calculates 11 summary statistics of IF_t .

Studies on the gold future market suggests that negative news sentiments tend to invoke greater responses from the market (Smales 2014). It is likely for this observation to be true in crude oil market as well since gold market and crude oil market share many similar features. One way to separate impacts from positive and negative news is to split all news into a positive and a negative group (neutral news are dropped). Then, applying φ_{daily} on these two subsets of information flow gives two copies of summaries, one for positive news and one for negative news. However, distinguishing positive and negative news

while constructing daily summary doubles the number of features generated (from 11ℓ to 22ℓ), and can potentially lead to the curse of dimensionality especially when ℓ is large (Friedman 1997). Therefore, for the daily summary, the proposed characteristic function only counts the number of news in each class but does not calculate detailed summary statistics (e.g., standard deviation and percentiles).

In contrast, while processing the aggregated information flow in the period of consideration, $\text{IF}_{[t-\ell,t)}$ (i.e., all news from day $t - \ell$ to day $t - 1$), instead of creating ℓ copies of daily summaries for ℓ days, the number of features constructed no longer depends on ℓ . This allows us to choose more complicated characteristic function for the aggregate information flow. Therefore, we may choose a characteristic function distinguishing positive and negative news and computes more detailed summary statistics.

Let $\varphi_{\text{aggregate}}$ denote the second characteristic function extracting features from $\text{IF}_{[t-\ell,t)}$.

Firstly, $\varphi_{\text{aggregate}}$ computes the 8 statistics in Table 5.3 for ESS and WEES of all news in $\text{IF}_{[t-\ell,t)}$ ($2 \times 8 = 16$ features in total).

Table 5.3: Summary Statistics from Gathering-First Paradigm (1)

Code Name	Variable
<code>x_count</code>	Number of Samples X
<code>x_mean</code>	Average of X
<code>x_std</code>	Standard Deviation
<code>x_min</code>	Minimum
<code>x_25%</code>	25^{th} Percentile
<code>x_50%</code>	Median
<code>x_75%</code>	75^{th} Percentile
<code>x_max</code>	Maximum

Secondly, to emphasize extreme events more, we then include 8 summary statistics in Table 5.3 for the ESS* and WEES* scores defined in equation (5.1.1) as well ($2 \times 8 = 16$

features in total).

$$\begin{aligned} \text{ESS}^* &:= \text{sign}(\text{ESS}) \times \text{ESS}^2 \\ \text{WEES}^* &:= \text{sign}(\text{WEES}) \times \text{WEES}^2 \end{aligned} \quad (5.1.1)$$

Thirdly, we split $\text{IF}_{[t-\ell,t)}$ into the positive group, $\text{IF}_{[t-\ell,t)}^+$, and the negative group, $\text{IF}_{[t-\ell,t)}^-$, according to the sign of each news' ESS score (news with zero ESS score are discarded). Note that by the definition of WEES, signs of ESS and WEES are always the same, hence, splitting $\text{IF}_{[t-\ell,t)}$ based on ESS and WEES always gives the same outcome. Then, $\varphi_{\text{aggregate}}$ includes (i) the number of news, (ii) the average ESS (iii) and the average WEES for each of $\text{IF}_{[t-\ell,t)}^+$ and $\text{IF}_{[t-\ell,t)}^-$ (6 features in total).

Lastly, as we noticed in the data exploration, there are two clusters of ESS scores (at -15 and 18). Moreover, news article assigned with these two values of ESS scores are in general reports of past price movements and carry little information about future price changes. To address this issue, we define $\text{IF}_{[t-\ell,t)}^{++}$ (extremely positive news) to be the subset of $\text{IF}_{[t-\ell,t)}$ with ESS strictly greater than 18, and $\text{IF}_{[t-\ell,t)}^{--}$ (extremely negative news) to be these news with ESS strictly less than -15. Afterwards, $\varphi_{\text{aggregate}}$ summarizes the number of extremely positive (negative), the average ESS and the average WEES of news in $\text{IF}_{[t-\ell,t)}^{++}$ and $\text{IF}_{[t-\ell,t)}^{--}$ (6 features in total).

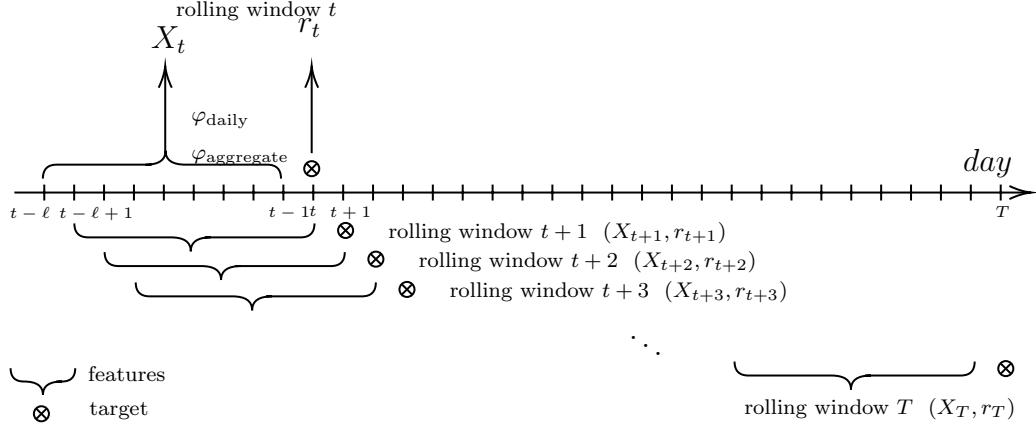
Overall, $\varphi_{\text{aggregate}}$ constructs $16+16+6+6 = 44$ features to summarize each information flow $\text{IF}_{[t-\ell,t)}$. Altogether with the 11ℓ features from φ_{daily} , $11\ell + 44$ features are used to predict the return r_t . For example, there would be 385 features if one chose ℓ to be 31. Given there are around 5,000 daily returns to train the model, using approximate 400 independent variable is reasonable.

5.1.2 Rolling-Window Method

The second step of the pipeline is to construct a batch of feature-target pairs (called a sample), (X_t, r_t) , so that we can evaluate model \mathcal{M} based on how close $\mathcal{M}(X_t)$ and r_t are. Let $\ell = 31$ for now, returns in the first month are discarded from the dataset since we do not have sufficient number of days to construct these features required. Afterwards,

a rolling-window method generates a training set from the series of returns and news dataset as illustrated in Figure 5.1.

Figure 5.1: Using Rolling Window to Construct (X_t, r_t) pairs



For each day t with valid return r_t (those days with missing returns are discarded), the set of features X_t consists of 31 daily summary from φ_{daily} , one aggregate summary from $\varphi_{\text{aggregate}}$ and 31 lagged values of returns.

$$X_t^{\ell=31} := \{\varphi_{\text{daily}}(\text{IF}_{t-31}), \dots, \varphi_{\text{daily}}(\text{IF}_{t-1}), \varphi_{\text{aggregate}}(\text{IF}_{[t-31,t]}), r_{t-31}, \dots, r_{t-1}\} \quad (5.1.2)$$

Therefore, $X_t^{\ell=31}$ consists of 416 real-valued features used to predict r_t . Afterwards, the rolling window constructor move to $t + 1$ (if available, otherwise move to the next day with valid returns) and generate another pair of feature and target (X_{t+1}, r_{t+1}) .

Finally, the rolling window generates 4,934 pairs of (X_t, r_t) , in which t ranges from January 1, 2000 to September 30, 2019. Among the 4,934 samples, each X_t is a 416 dimensional real-valued vector and r_t is a real-valued scalar. This paper uses samples (X_t, r_t) with t before January 1, 2019 as training set (4,747 samples) and the rest of samples are taken as test set (187 samples).

$$\mathcal{D}^{\text{train}} := \{(X_t, r_t) : t \leq \text{December 31, 2018}\} \quad (5.1.3)$$

$$\mathcal{D}^{\text{test}} := \{(X_t, r_t) : t \geq \text{January 1, 2019}\} \quad (5.1.4)$$

After assessing models' performances on the test set, $\mathcal{D}^{\text{train}}$ and $\mathcal{D}^{\text{test}}$ are further split

into 5 subsets ¹ to explore the effectiveness of models on each day of the week.

5.1.3 Performance Metrics

In order to quantify the performance of a predictive model, we have to specify a performance metric measuring the proximity between predictions and actual values. Let \hat{r}_t denote the predicted value of r_t , the performance metric should reflect the proximity between predicted and true returns. The primary performance metric used in this paper is the mean squared error (MSE). The MSE of a model aiming to predict $\{r_1, r_2, \dots, r_T\}$ is defined as

$$MSE := \frac{1}{T} \sum_{t=1}^T (r_t - \hat{r}_t)^2 \quad (5.1.5)$$

One advantage of MSE metric is that it is differentiable with respect to each \hat{r}_t , this differentiability allows us to train models on this dataset using back-propagation algorithm (Hecht-Nielsen 1989). Even though not all predictive models in this paper are based on back-propagation or require differentiable objective functions, we use MSE as the primary metric to select and evaluate models for consistency.

Unfortunately, the MSE is not naturally interpretable, and MSE changes when the unit of returns switches to percentage returns. We introduce another widely used error metric, directional accuracy (DA) defined as following:

$$DA := \frac{1}{T} \sum_{t=1}^T \mathbb{1}\{\text{sign}(r_t) = \text{sign}(\hat{r}_t)\} \quad (5.1.6)$$

The directional accuracy measures the frequency that the model predict the sign of return correctly. The directional accuracy can be interpreted easily, but it is not differentiable.

¹ (X_t, r_t) are split based on which day of the week t is. There are only 5 groups since r_t are always missing when t is weekend.

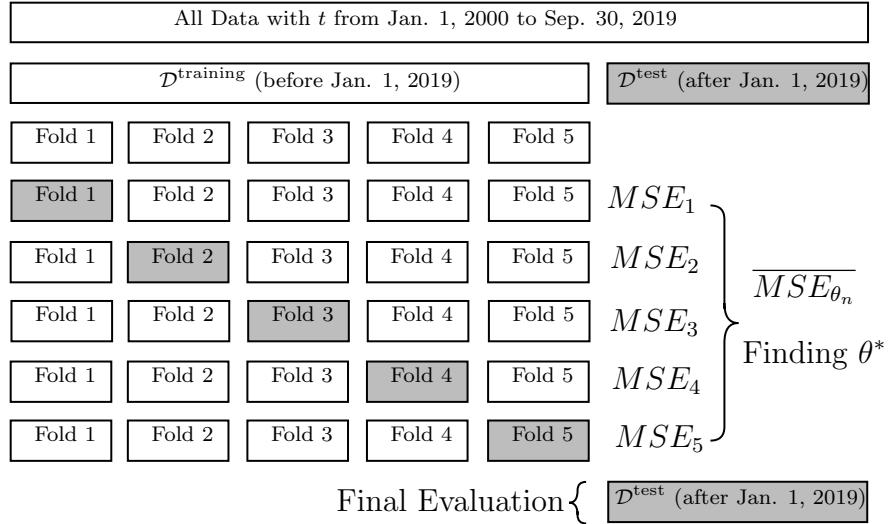
5.1.4 Searching Technology (Model Selection) and Randomized Cross Validation

After choosing a class of predictive models, one still needs to select a set of hyper-parameters (i.e., model configurations), $\theta \in \Theta$. In subsequent discussions, we use subscript \mathcal{M}_θ to denote a model with configuration θ from class \mathcal{M} . For example, suppose \mathcal{M} is the class of all polynomial regressions, then one hyper-parameter is the maximum degree in the regression equation. In this case, the set of possible hyper-parameters, Θ , is all positive integers. One has to choose the optimal maximum degree θ^* from Θ so that \mathcal{M}_{θ^*} has the best (test-time) performance.

Choosing the optimal θ^* is crucial for building effective predictive algorithms, simply choosing a super complicated model would over-fit the training set and leads to poor test-time performance (Claeskens and Hjort 2008).

For each predictor class \mathcal{M} and corresponding space of hyper-parameters Θ , we firstly determine the optimal hyper-parameter $\theta^* \in \Theta$ using a 5-fold randomized cross validation algorithm (5-fold RCV).

Figure 5.2: 5-fold Randomized Cross Validation



Firstly, N candidates of hyper-parameters $\{\theta_1, \theta_2, \dots, \theta_N\}$ are independently sampled from a uniform distribution on Θ (i.e., the randomized part in RCV). Then, for each $\theta_n \in \{\theta_1, \theta_2, \dots, \theta_N\}$, θ_n defines a predictive model \mathcal{M}_{θ_n} . Figure 5.2 illustrates the cross

validation procedure for one hyper-parameter set θ_n . The entire $\mathcal{D}^{\text{train}}$ are split into 5 equal consecutive subsets (called folds): $\mathcal{D}_i^{\text{train}}$ for $i \in \{1, 2, 3, 4, 5\}$. Then, \mathcal{M}_{θ_n} is fitted on $\cup_{i \in \{1, 2, 3, 4\}} \mathcal{D}_i^{\text{train}}$ (training set) and evaluated on $\mathcal{D}_5^{\text{train}}$ (validation set), let \widehat{MSE}_5 denote mean squared error of this model on $\mathcal{D}_5^{\text{train}}$. Afterwards, the same model is fitted again on $\cup_{i \in \{1, 2, 3, 5\}} \mathcal{D}_i^{\text{train}}$, evaluated on $\mathcal{D}_4^{\text{train}}$ and leads to another error metric \widehat{MSE}_4 . The same procedure can be repeated for five times with different validation set and creates five mean squared error metrics. Let $\overline{MSE}_{\theta_n}$ denote the average of \widehat{MSE}_1 to \widehat{MSE}_5 using hyper-parameter θ_n . Then, $\overline{MSE}_{\theta_n}$ constitutes an estimated test-time performance of model \mathcal{M}_{θ_n} on $\mathcal{D}^{\text{test}}$ when it is fitted on $\mathcal{D}^{\text{train}}$. Among the N candidates of hyper-parameters for model class \mathcal{M} , we choose θ_n with the smallest $\overline{MSE}_{\theta_n}$ to be the best-performing parameter, denoted as θ^* . This θ^* is not necessarily the truly best one among all θ in Θ , θ^* is only the best-performing configuration within N samples. However, since we sampled the N candidates uniformly from Θ , performance of the selected θ^* should be close to the truly optimal configuration especially when N is sufficiently large.

While using 5-fold RCV and N sampled θ , we need to fit $5N$ models from class \mathcal{M} in total to determine the optimal configuration θ^* for this model class. Clearly, the larger N is the more likely for us to include the truly optimal configuration in our sampled configurations. Specifically, we choose N to be 500 in this paper.

Since this paper works with two performance metrics: mean-squared-error (MSE) and directional accuracy (DA), there are at least two criterions to identify the optimal model.

MSE-Optimal The first selection criterion identifies the optimal model from a given model class based on models' validation MSEs. Given a class of models, one model is the MSE-optimal of this model class if

- this model achieves at least a DA of 50% on the validation set;
- and, it achieves the lowest MSE on the validation set among all models satisfy the first condition.

If there are more than one models with the same lowest validation MSE and DA greater than 50%, we will choose the one with the best DA to be the optimal model. This

type of model identifying method based on validation MSE is called **randomized cross validation with mean squared error** (RCV-MSE) in this paper.

DA-Optimal The second criterion selects the best model from candidates based on their validation DAs. Given a class of models, one model is the DA-optimal of this model class if

- this model achieves the highest DA on the validation set.

If there are multiple models with the same highest accuracy, the model with the lowest validation MSE is chosen to be the DA-optimal. This kind of searching technology that identifies the optimal model based on validation DA is referred to as **randomized cross validation with directional accuracy** (RCV-DA).

Let $\mathcal{M}_{\text{model class}}^{\text{MSE}}$ and $\mathcal{M}_{\text{model class}}^{\text{DA}}$ denote the MSE-optimal and DA-optimal models from a particular model class. After identifying optimal models, the performances of $\mathcal{M}_{\text{model class}}^{\text{MSE}}$ and $\mathcal{M}_{\text{model class}}^{\text{DA}}$ on the test set will be proxies of the performance of the specified model class on the prediction task. Note that these optimal models can be identified completely without the test set, therefore, evaluating them on the test set mimics real-world environment: someone builds and trains a model at time t using all information available up to time t , then the model is implemented on company's server and used in production after day t . What traders really care about is the predictive model's performance after day t , called test time performance, since the test time performance is directly related to the profitability of any trading algorithms built on this predictive model. Therefore, one model's performance on the test set is a fair proxy for the business value of this model in real-world.

5.2 Baseline Models: The Naive Predictor and Moving Average Predictors

To answer the first research question, whether crude oil returns are predictable, we need to firstly define several dummy models for benchmarking purpose. Recall that this

paper defines two separate information sets:

- Ω_{partial} denotes the information set containing historical returns only.
- Ω_{complete} denotes the information set consisting of both historical returns and news sentiments.

Predictive models based on Ω_{partial} only use 31 lagged returns to predict future returns, in contrast, models based on Ω_{complete} utilizes all 416 features including lagged returns and features extracted from news sentiments.

If a model \mathcal{M} based on $\Omega \in \{\Omega_{\text{partial}}, \Omega_{\text{complete}}\}$ fails to out-perform baseline models on the test set, then we may conclude the crude oil market is efficient (i.e., unpredictable) with respect to this model and information set Ω .

The simplest model is a **naive predictor**, $\mathcal{M}_{\text{naive}}$, based on the martingale assumption on crude oil prices. This model assumes the close spot price p_t on day t to be exactly the close price on the previous day. Therefore, $\mathcal{M}_{\text{naive}}$ is predicting zero returns all the time.

$$\hat{r}_t = \mathcal{M}_{\text{naive}} = 0 \quad (5.2.1)$$

In addition, we define other **moving-average predictors**² denoted as $\mathcal{M}_{\text{MA}(k)}$, where k is a positive integer representing the scope of this model. To predict return r_t , the model $\mathcal{M}_{\text{MA}(k)}$ looks into the past k trading days and predicts the return to be the average return of them.

$$\hat{r}_t = \mathcal{M}_{\text{MA}(k)} = \frac{1}{k} \sum_{\tau=t-k}^{t-1} r_\tau \quad (5.2.2)$$

As mentioned before, this paper uses all data before December 31, 2018 as the training set (4,746 trading days) and data from January 1, 2019 to October 31, 2019 as the test set (187 trading days). Table 5.4 presents performances of benchmark models in terms of mean-squared-error (MSE) and directional accuracy (DA).

²The moving-average predictors is not related to ARIMA models introduced later.

Table 5.4: Performances of Benchmark Models

Model	Training MSE	Training DA	Testing MSE	Testing DA
$\mathcal{M}_{\text{naive}}$	4.655	0.716%	4.057	0.538%
$\mathcal{M}_{\text{MA}(5)}$	5.612	50.274%	4.693	50.000%
$\mathcal{M}_{\text{MA}(25)}$	4.811	50.295%	4.248	50.000%
$\mathcal{M}_{\text{MA}(50)}$	4.725	49.536%	4.261	50.000%
$\mathcal{M}_{\text{MA}(100)}$	4.706	49.241%	4.226	44.624%
$\mathcal{M}_{\text{MA}(300)}$	4.676	47.977%	4.060	48.925%

Note that the directional accuracy of $\mathcal{M}_{\text{naive}}$ model is nearly zero on both training and testing sets because returns are rarely exactly zero in the dataset. As we expected, those benchmark models are too simple to achieve better performances than random guessing (i.e., 50% directional accuracy). This preliminary analysis on benchmark models' performances leads to Conclusion 5.2.1.

Conclusion 5.2.1. These results suggest that the crude oil market is efficient (i.e., unpredictable) with respect to

- (i) the information set Ω_{partial} containing historical returns
- (ii) and both native predictor and moving-average predictors.

Then we are going to examine whether other more sophisticated models based on Ω_{partial} can attain significantly better test time performances than those above-mentioned benchmark models.

After evaluating performances of these benchmark models, we can answer the two research questions using the following Rule 5.2.1 and Rule 5.2.2.

Rule 5.2.1 (Research Question 1). Given an information set $\Omega \in \{\Omega_{\text{partial}}, \Omega_{\text{complete}}\}$, a class of models and a model searching technology, let \mathcal{M}^* denote the optimal model identified. Then the crude oil market is claimed to be predictable (i.e., inefficient) if (i) the testing MSE of \mathcal{M}^* is lower than MSE values of benchmark models and (ii) the testing DA of \mathcal{M}^* is higher than 50%.

Rule 5.2.2 (Research Question 2). Given a class of models and searching technology, let $\mathcal{M}_{\Omega_{\text{partial}}}^*$ and $\mathcal{M}_{\Omega_{\text{complete}}}^*$ denote optimal models on two information sets identified by the searching technology. Then this paper concludes that we can better predict returns by incorporating news sentiments if (i) $\mathcal{M}_{\Omega_{\text{complete}}}^*$ has better testing MSE and testing DA than $\mathcal{M}_{\Omega_{\text{partial}}}^*$ and (ii) testing MSE of $\mathcal{M}_{\Omega_{\text{complete}}}^*$ is lower than benchmark MSEs and testing DA of $\mathcal{M}_{\Omega_{\text{complete}}}^*$ is higher than 50%.

In order to claim news sentiment are helpful, Rule 5.2.2 requires $\mathcal{M}_{\Omega_{\text{complete}}}^*$ to be an useful model.

5.3 Linear Models: Autoregressive Integrated Moving Average

One classical model used for time series forecasting is the autoregressive integrated moving average (ARIMA) model.

The autoregressive moving average (ARMA) models the return at time step t , r_t , as a function of the series itself and a series of error terms. In particular, an ARMA(p, q) process identifies r_t as a linear combination of p lagged variation of r_t and q noise terms:

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} \quad (5.3.1)$$

More concisely, let L denote the lag operator, an ARMA(p, q) process can be written with the more compact notion

$$\Phi_p(L)r_t = \Theta_q(L)\varepsilon_t \quad (5.3.2)$$

where $\Phi_p(\cdot)$ and $\Theta_q(\cdot)$ are polynomials up to degree p and q respectively.

One crucial assumption on $\{r_t\}$ is that it has to be stationary. To handle non-stationary processes, one can apply the differencing operator, $1 - L$, iteratively until the differenced series becomes stationary. If the stationarity is achieved after d iterations of differencing, the original series is said to be integrated of order d and can be modelled using an

ARIMA(p, d, q) model:

$$\Phi_p(L)(1 - L)^d X_t = \Theta_q(L) \varepsilon_t \quad (5.3.3)$$

ARIMA models the intertemporal dependencies naturally, however, ARIMA can only handle linear relationships. In later sections, non-linear models such as support vector machines will be introduced.

Since ARIMA models work on univariate time series only, we are only assessing whether the market is efficient with respect to ARIMA models and the partial information set Ω_{partial} .

Instead of cross validation, the optimal model is identified based on Akaike's information criterion (AIC). For a given ARIMA model, let k denote the number of parameters in this model, the value of k is positively correlated with the model's complexity. Let L denote the maximum value of likelihood function of this model on the training set. The AIC of this model is defined in equation (5.3.4).

$$AIC = 2k - \ln(L) \quad (5.3.4)$$

Since simpler models are less likely to overfit the training set and can generalize better. The AIC penalizes model complexity and rewards log-likelihood, therefore, minimizing AIC seeks for a balance between the model's complexity and training set performance.

One ARIMA model can be uniquely specified by a set of orders (p, d, q) . This paper trains all ARIMA models with $6 \times 3 \times 6 = 72$ different combinations of (p, d, q) specified in Table 5.5. The model attains the lowest AIC is identified to be the optimal model. For generality, this paper reports the best three models. It turns out that ARIMA(5,0,5), ARIMA(4,0,3) and ARIMA(5,0,4) are the three models attain the lowest three AIC values on training set. Table 5.6 summarizes the performances of these three optimal models identified.

Table 5.5: Scope of Orders for ARIMA

Parameter	Scope
p	$\{0,1,2,3,4,5\}$
d	$\{0,1,2\}$
q	$\{0,1,2,3,4,5\}$

Table 5.6: Performances of Linear Models

Model	Testing MSE	Testing DA
$\mathcal{M}_{\text{ARIMA}(5,0,5)}$	4.074	50.763 %
$\mathcal{M}_{\text{ARIMA}(4,0,3)}$	4.070	51.156 %
$\mathcal{M}_{\text{ARIMA}(5,0,4)}$	4.073	50.567 %

Performances of these models suggest that ARIMA models can hardly achieve a better-than-guessing accuracy. Moreover, in terms of test time MSE, all three ARIMA models preform poorly compared with naive predictor, which achieves testing MSE of 4.057. Such unsatisfactory performances suggest that the crude oil return is essentially unpredictable using ARIMA models based on the partial information set. Therefore, we can extend our previous finding to Conclusion 5.3.1.

Conclusion 5.3.1. The crude oil market is efficient with respect to

- (i) Information set: the information set Ω_{partial} containing historical returns.
- (ii) Model Class: ARIMA models.
- (iii) Searching Technology: the model attaining the lowest AIC on training set is identified to be the optimal model.

5.4 Support Vector Regression

The previous section shows that the crude oil market is efficient with respect to linear models. In contrast, this section is devoted to non-linear models and answers whether

the market is efficient against non-linear models.

Support vector machines (SVM) was first proposed by Boser, Guyon and Vapnik as a classification method for hand-written digit recognition (1992). Over the past three decades, SVM has been believed to be the best off-the-shelf classification algorithm.

As mentioned in section 5.1.1, characteristic functions generate 416 predictors in total for each one target r_t , so that the prediction task is in fact a high dimensional problem. SVM models only focus a few training samples termed **support vectors**, therefore, SVMs often produce promising results on high dimensional problems.

Moreover, by using different **kernel functions**, SVMs are capable of transforming these raw features to an even higher dimensional space. For example, if one wishes to classify points in \mathbb{R}^2 , other algorithms like logistic regressions would classify the point based on (x_1, x_2) directly and maps (x_1, x_2) to class labels. Instead, a SVM with polynomial kernel of degree two will classify the point based on all polynomials of (x_1, x_2) up to degree two, that is, $(x_1, x_2, x_1^2, x_1x_2, x_2^2) \in \mathbb{R}^5$. In this case, the original 2-dimensional input space is transformed into a 5-dimensional feature space by the kernel function. While a SVM is using the Radial Basis Function (RBF) kernel, the original input space is transformed into an infinite-dimensional feature space. This implicit feature engineering enables SVM to explore more complex patterns in the dataset. Smola and Scholkorf provide a detailed review of training SVMs and theories behind kernel functions in their work (2004).

A few years after the SVM was proposed as a classifier, Drucker and others proposed an extension to original SVM called support vector regression machines (SVR) (Drucker et al. 1997). As the name suggests, SVR is designed for regression problems. It has been shown that SVRs perform reasonably well on high dimensional regression problems by focusing on a few support vectors and engineering features implicitly using kernel functions.

As mentioned in Smola and Scholkorf's work, performance of support vector machines is mainly determined by several hyper-parameters listed in Table 5.7. To choose the optimal configuration of SVR in this paper's prediction task, a RCV algorithm samples 500 configurations from the scope of hyper-parameters in Table 5.7 and evaluates each

configuration based on their MSE and DA on the validation set.

Table 5.7: Scope of Hyper-parameters for Support Vector Regression Machines

Hyper-parameter	Scope
Kernel Type	{Radial Basis Function (RBF) kernel}
γ	$\{10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10\}$
Tolerance	$\{10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10\}$
ε	$\{10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10\}$
C	$\{10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10\}$

Table 5.8 and Table 5.9 presents the optimal models under both criterions and their respective performances.

Table 5.8: Optimal Hyper-parameters for Support Vector Regression Machines

Model	Kernel	γ	Tolerance	ε	C
For Ω_{partial}					
$\mathcal{M}_{\text{SVR}}^{\text{MSE}}$	RBF	0.1	10^{-2}	1	1
$\mathcal{M}_{\text{SVR}}^{\text{DA}}$	RBF	0.1	10^{-7}	10^{-6}	1
For Ω_{complete}					
$\mathcal{M}_{\text{SVR}}^{\text{MSE}}$	RBF	10^{-6}	1	10^{-3}	10^{-9}
$\mathcal{M}_{\text{SVR}}^{\text{DA}}$	RBF	10^{-7}	10^{-5}	10^{-7}	10

Table 5.9: Performances of Support Vector Regression Machines

Model	Validation MSE	Validation DA	Testing MSE	Testing DA
Trained on Ω_{partial}				
$\mathcal{M}_{\text{SVR}}^{\text{MSE}}$	4.654	51.033%	4.036	52.941%
$\mathcal{M}_{\text{SVR}}^{\text{DA}}$	4.655	51.833%	4.040	53.476%
Trained on Ω_{complete}				
$\mathcal{M}_{\text{SVR}}^{\text{MSE}}$	4.655	51.433%	4.055	54.301%
$\mathcal{M}_{\text{SVR}}^{\text{DA}}$	4.830	51.665%	4.399	49.462%

The MSE-optimal SVR, $\mathcal{M}_{\text{SVR}}^{\text{MSE}}$, outperforms benchmark models and ARIMA models examined before. Using either Ω_{partial} or Ω_{complete} , $\mathcal{M}_{\text{SVR}}^{\text{MSE}}$ achieves better test time DA compared with the random-guessing-accuracy. Hence, this paper concludes that the market is essentially predictable with respect to both information sets and SVR-optimal support vector regression machines. In contrast, test time performance of the DA-optimal SVR trained on Ω_{complete} suggests that the market is efficient in this case. Conclusion 5.4.1 summarizes answers to the first research question.

Conclusion 5.4.1.

Model Class	Information Set	Searching Technology	Market Efficiency
SVR	Ω_{partial}	RCV-MSE	Inefficient
SVR	Ω_{partial}	RCV-DA	Inefficient
SVR	Ω_{complete}	RCV-MSE	Inefficient
SVR	Ω_{complete}	RCV-DA	Efficient

As for the second research question, even though both $\mathcal{M}_{\text{SVR}}^{\text{MSE}}$ on Ω_{partial} and on Ω_{complete} attain lower test time MSE than benchmark models and ARIMA models, there is no improvement by utilizing the complete information set (4.055 MSE) instead of the partial information set (4.036 MSE). Moreover, the DA-optimal SVR trained on Ω_{complete} per-

forms even worse than the DA-optimal SVR trained on Ω_{partial} . Above observations lead to Conclusion 5.4.2 and Conclusion 5.4.3, which answer the second research question.

Conclusion 5.4.2. While using support vector machines as predictive models and RCV-MSE searching technology, incorporating news sentiment dataset does not help predict crude oil returns.

Conclusion 5.4.3. While using RCV-DA searching technology, adding news sentiment features even worsens the performance of support vector machines.

Lastly, the two conclusions above suggest that if one wishes to extend current SVR-based predictive models to incorporate news sentiment features, RCV-MSE is a dominant strategy to identify the optimal model.

5.5 Random Forests

Another class of non-linear methods used widely is the random forest. Breiman proposed an ensemble model based on traditional tree methods called random forest (2001). A forest as an ensemble of independently trained trees reduces the variance of prediction and achieves a better performance compared with one single tree predictor. Let k denote the number of independent variables for prediction ($k = 416$ here). Training each tree in the forest using all k features can cause over-fitting problems and lead to poor test time performance. Therefore, each independent tree predictor in the forest is trained only using a random subset of k features, which constitutes the randomness of a random forest. This randomness on feature selection helps random forests to generalize better and achieves even lower loss on the test set. Typically, each tree in the forest is only trained on $\log_2(k)$ features.

Table 5.10 enumerates key hyper-parameters of a random forest predictor and corresponding scopes our cross validation procedure searches over.

Table 5.10: Scope of Hyper-parameters for Random Forests

Hyper-parameter	Scope
n Number of trees	$\{1, 2, 3, \dots, 200\}$
f Max num. of features for each tree	$\{k, \log_2(k)\}$
d Max depth of each tree	$\{10, 14, 19, 24, \dots, 100, 105, 110, \infty\}$
m_1 Min amount of samples required to split an internal node	$\{2, 5, 10\}$
m_2 Minimum number of samples required at each leaf node	$\{1, 2, 4\}$
What dataset is used to construct each tree	{bootstrapped samples, entire training dataset}

Table 5.10 summarizes configurations of $\mathcal{M}_{\text{RF}}^{\text{MSE}}$ and $\mathcal{M}_{\text{RF}}^{\text{DA}}$ identified.

Table 5.11: Optimal Hyper-parameters for Random Forests

Model	n	f	d	m_1	m_2	Training Samples
For Ω_{partial}						
$\mathcal{M}_{\text{RF}}^{\text{MSE}}$	41	$\log_2(k)$	10	5	1	Bootstrapped Samples
$\mathcal{M}_{\text{RF}}^{\text{DA}}$	130	k	10	10	1	Entire Training Set
For Ω_{complete}						
$\mathcal{M}_{\text{RF}}^{\text{MSE}}$	96	$\log_2(k)$	10	10	2	Bootstrapped Samples
$\mathcal{M}_{\text{RF}}^{\text{DA}}$	41	k	14	5	4	Entire Training Set

After identifying optimal models, these models are evaluated using the test set and Table 5.11 reports performances of random forests.

Surprisingly, random forests trained on Ω_{partial} consistently perform better than those trained on Ω_{complete} regardless of the searching technology used. Test time accuracies of models on Ω_{partial} suggest the market is predictable in this setting. Moreover, note that models on Ω_{partial} attain lower testing MSE compared with benchmark models (4.057 MSE), this observation further confirm our previous conclusion that the market is inefficient.

In contrast, random forests defined on Ω_{complete} only achieve an unsatisfactory result in terms of testing MSE. Both optimal models perform worse than a naive predictor, which attains a testing MSE of 4.057.

Table 5.12: Performances of Random Forests

Model	Validation MSE	Validation DA	Testing MSE	Testing DA
Trained on Ω_{partial}				
$\mathcal{M}_{\text{RF}}^{\text{MSE}}$	4.717	50.716%	4.024	53.476%
$\mathcal{M}_{\text{RF}}^{\text{DA}}$	5.325	51.243%	4.053	53.476%
Trained on Ω_{complete}				
$\mathcal{M}_{\text{RF}}^{\text{MSE}}$	4.675	50.464%	4.148	48.387%
$\mathcal{M}_{\text{RF}}^{\text{DA}}$	5.716	51.960%	4.753	53.226%

In terms of the directional accuracy, $\mathcal{M}_{\text{RF}}^{\text{DA}}$ achieves better results than random guessing on both validation and test sets. However, a market is predictable only if a model can consistently achieve better-than-guessing results. The statistics in Table 5.12 are not sufficient to conclude that the market is predictable. It is possible that too many sentiment features are generated in the previous section and most of them are essentially noises. Consequently, the informative content in Ω_{partial} are masked by these additional noisy signals in Ω_{complete} and random forests fail to perform well on Ω_{complete} . Findings on random forests answering the first research question are summarized in Conclusion 5.5.1.

Conclusion 5.5.1.

Model Class	Information Set	Searching Technology	Market Efficiency
RF	Ω_{partial}	RCV-MSE	Inefficient
RF	Ω_{partial}	RCV-DA	Inefficient
RF	Ω_{complete}	RCV-MSE	Efficient
RF	Ω_{complete}	RCV-DA	Efficient

The answer to the second research question is straightforward in the random forest's case.

Conclusion 5.5.2. It is unlikely to improve random forest's performance on crude oil return forecasting by utilizing additional news sentiment features.

5.6 Long Short-Term Memory Recurrent Neural Networks

An ARIMA model captures the intertemporal correlation among independent variables explicitly. However, ARIMA models are not capable of modelling complex non-linearities. The superior performance of SVRs indicates that considering non-linear interactions among independent variables can improve model performance significantly.

Unfortunately, even though SVRs and random forests are capable to model complex non-linearities, they squeeze all independent variables and disregard the sequence structures among independent variables. In fact, SVRs and random forests regard all permutations of input features to be the same. For instance, the two sets of independent variables in equation (5.6.1) and equation (5.6.2) are considered as the same for SVRs and random forests. But clearly, these two sequences encode completely different trend.

$$p_1 = 1, p_2 = 2, p_3 = 3, \dots, p_9 = 9 \quad (5.6.1)$$

$$p_1 = 9, p_2 = 8, p_3 = 7, \dots, p_9 = 1 \quad (5.6.2)$$

In current literature, neural networks have been widely used to capture non-linearity among independent variables. One special type of neural works termed recurrent neural networks (RNN) is designed to model both non-linearities and intertemporal correlations. However, conventional RNNs suffer from vanishing and exploding gradient problems and becomes impotent on longer time series. In section 3.2, the ACF and PACF plots have identified possible seasonality in crude oil returns. Modelling seasonality requires the RNN to pay attention to intertemporal dependencies over a longer period of time.

Hochreiter and Schmidhuber proposed the long short-term memory (LSTM) cell for RNNs, this novel architecture allows RNNs to focus on intertemporal dependencies over both short and long periods (1997). All RNNs trained and evaluated in this paper are based on this LSTM architecture.

Table 5.13 summarizes the scope of hyper-parameters for LSTM RNNs.

Table 5.13: Scope of Hyper-parameters for LSTM RNNS

Hyper-parameter	Scope
Epochs of training	$\{5, 6, 7, 8, \dots, 18, 19, 20, 25, 30, 35, \dots, 200\}$
h Size of RNN hidden layer	$\{32, 64, 128, 256, 512, 1024\}$
ℓ Number of RNN hidden layers	$\{1, 2, 3\}$
p_{rnn} Dropout probability in RNN hidden layers	$\{0, 0.25, 0.5\}$
p_{fc} Dropout probability in the output layer	$\{0, 0.25, 0.5\}$
B Batch size	$\{32, 128, 512\}$
α Learning rate	$\{10^{-5}, 3 \times 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 10^{-3}, 3 \times 10^{-3}, 0.01, 0.03, 0.1, 0.3\}$

Table 5.14 and Table 5.15 present two optimal models and their test time performances. Because LSTM RNNs are capable to capture all of non-linearities, intertemporal dependencies over short time periods (interday transitions) and intertemporal dependencies over longer time periods (seasonalities), both $\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$ and $\mathcal{M}_{\text{LSTM}}^{\text{DA}}$ out-perform all other models implemented earlier in this paper.

Table 5.14: Optimal Hyper-parameters for LSTM RNNs

Model	Epochs	h	ℓ	p_{rnn}	p_{fc}	B	α
For Ω_{partial}							
$\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$	8	512	3	0.5	0.5	32	3×10^{-4}
$\mathcal{M}_{\text{LSTM}}^{\text{DA}}$	155	32	3	0.5	0.0	512	0.1
For Ω_{complete}							
$\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$	40	32	2	0.0	0.5	512	10^{-5}
$\mathcal{M}_{\text{LSTM}}^{\text{DA}}$	12	128	2	0.0	0.25	512	10^{-5}

Table 5.15: Performances of LSTM RNNs

Model	Validation MSE	Validation DA	Testing MSE	Testing DA
Trained on Ω_{partial}				
$\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$	3.992	52.450%	4.044	44.385%
$\mathcal{M}_{\text{LSTM}}^{\text{DA}}$	4.668	53.792%	4.045	54.011%
Trained on Ω_{complete}				
$\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$	4.192	51.609%	4.043	54.012%
$\mathcal{M}_{\text{LSTM}}^{\text{DA}}$	4.888	54.480%	4.041	54.011%

Statistics in Table 5.15 suggest that testing MSE values of LSTM-RNN are always below MSE values of benchmark models. Even though $\mathcal{M}_{\text{LSTM}}^{\text{MSE}}$ on Ω_{partial} does not perform well in terms of testing DA (44.385%), its performance is improved significantly to 54.012% after switching to Ω_{complete} . This observation serves as an evidence implying that including news sentiment can help crude oil forecasting. Overall, Conclusion 5.6.1 and Conclusion 5.6.2 answer the first and the second research question of this thesis.

Conclusion 5.6.1. The efficiencies of crude oil market with respect to LSTM-RNN, various information sets and searching technologies are summarized as:

Model Class	Information Set	Searching Technology	Market Efficiency
LSTM-RNN	Ω_{partial}	RCV-MSE	Efficient
LSTM-RNN	Ω_{partial}	RCV-DA	Inefficient
LSTM-RNN	Ω_{complete}	RCV-MSE	Inefficient
LSTM-RNN	Ω_{complete}	RCV-DA	Inefficient

Conclusion 5.6.2. A LSTM-RNN selected using RCV-MSE can leverage the predictive power of news sentiment. That is, and crude oil market can be better predicted by including news sentiments.

5.7 Taking the Day-of-the-Week Effect into Consideration

In section 3.3, we have shown that crude oil returns experience the day-of-the-week effect. In particular, the empirical distribution of Mondays' returns is significantly different from distributions of other days. Moreover, Mondays are more likely to experience negative returns compared with other days. Therefore, it is reasonable to conjecture that the underlying dynamics of returns on Mondays might be different from the dynamics of returns of other days. Recall that the original dataset built upon Ω_{complete} consists of 4,933 pairs of (X_t, r_t) and each feature vector $X_t \in \mathbb{R}^{416}$. Among the 4,932 feature-target pairs (X_t, r_t) , targets of the first 4,746 pairs are returns before December 31, 2018 and these pairs are used as the training set. In contrast, the last 186 pairs serve as the testing set.

To examine the potential benefits from building different models for different days of the week, training and testing sets are split into two training and two testing sets.

- Training set (Monday) consists of samples (X_t, r_t) from the original training set such that t is a Monday (889 samples).
- Testing set (Monday) consists of samples (X_t, r_t) from the original testing set such

that t is a Monday (34 samples).

- Training set (other days) consists of samples (X_t, r_t) from the original training set such that t is not a Monday (3,857 samples).
- Testing set (other days) consists of samples (X_t, r_t) from the original testing set such that t is not Monday (152 samples).

The same RCV-MSE and RCV-DA techniques are applied on the two training sets to identify the corresponding MSE-optimal and DA-optimal models for returns on Mondays and other days. Table 5.16, Table 5.17 and Table 5.18 report optimal models for each dataset. Notations for optimal models are similar as before, for example, $\mathcal{M}_{\text{RF, Mondays}}^{\text{MSE}}$ represents the MSE-optimal random forest for predicting returns on Mondays and $\mathcal{M}_{\text{SVR, Other Days}}^{\text{MSE}}$ indicates the MSE-optimal SVR for predicting returns on Tuesdays, Wednesdays, Thursdays and Fridays.

Table 5.16: Optimal Hyper-parameters for Random Forests on Restricted Datasets

Model	n	f	d	m_1	m_2	Training Samples
$\mathcal{M}_{\text{RF, Mondays}}^{\text{MSE}}$	115	$\log_2(k)$	38	10	4	Bootstrapped Samples
$\mathcal{M}_{\text{RF, Mondays}}^{\text{DA}}$	116	$\log_2(k)$	38	2	1	Entire Training Set
$\mathcal{M}_{\text{RF, Other Days}}^{\text{MSE}}$	88	$\log_2(k)$	10	5	4	Bootstrapped Samples
$\mathcal{M}_{\text{RF, Other Days}}^{\text{DA}}$	157	k	10	2	1	Entire Training Set

Table 5.17: Optimal Hyper-parameters for Support Vector Regressions on Restricted Datasets

Model	Kernel	γ	Tolerance	ε	C
$\mathcal{M}_{\text{SVR, Mondays}}^{\text{MSE}}$	RBF	10^{-10}	10^{-3}	10^{-4}	10
$\mathcal{M}_{\text{SVR, Mondays}}^{\text{DA}}$	RBF	10^{-6}	0.1	10^{-6}	1
$\mathcal{M}_{\text{SVR, Other Days}}^{\text{MSE}}$	RBF	0.1	0.1	10^{-4}	10
$\mathcal{M}_{\text{SVR, Other Days}}^{\text{DA}}$	RBF	10^{-9}	0.1	10^{-7}	1

Table 5.18: Optimal Hyper-parameters for LSTM RNNs on Restricted Dataset

Model	Epochs	h	ℓ	p_{rnn}	p_{fc}	B	α
$\mathcal{M}_{\text{LSTM, Mondays}}^{\text{MSE}}$	45	128	2	0.5	0.0	128	10^{-4}
$\mathcal{M}_{\text{LSTM, Mondays}}^{\text{DA}}$	75	1024	3	0.0	0.25	128	0.01
$\mathcal{M}_{\text{LSTM, Other Days}}^{\text{MSE}}$	14	512	3	0.0	0.5	32	0.001
$\mathcal{M}_{\text{LSTM, Other Days}}^{\text{DA}}$	12	512	3	0.0	0.0	32	0.001

After optimal models for each dataset are identified, they are evaluated on the corresponding testing sets. Performances of models are reported in Table 5.19, best performances are in bold font. As we have expected, optimal models for both datasets are from LSTM RNN class since LSTM-RNNs are capable of capturing both key factors in time series forecasting task: non-linearities and intertemporal correlations. In particular, $\mathcal{M}_{\text{LSTM, Mondays}}^{\text{MSE}}$ achieves a superior performance in terms of the directional accuracy in the test set.

The test-time performance of using two separate models can be estimated by taking the weighted average of loss/accuracy on two testing sets using equation (5.7.1). Let \mathcal{M}_A and \mathcal{M}_B denote models used for Mondays and other days in the joint model. The joint model uses \mathcal{M}_A to make prediction every Monday and uses \mathcal{M}_B on other days.

$$\text{Performance} \approx \% \text{ of Mondays} \times \mathcal{M}_A \text{'s test time performance}$$

$$\begin{aligned}
 & + \% \text{ of other days} \times \mathcal{M}_B \text{'s test time performance} \\
 & = \frac{34}{186} \times \mathcal{M}_A \text{'s test time performance} + \frac{152}{186} \times \mathcal{M}_B \text{'s test time performance}
 \end{aligned} \tag{5.7.1}$$

all three types of models used are data-intensive but there are only 889 training samples of Mondays. Therefore, none of them delivers a significantly better result compared with models previously trained using the entire dataset.

Table 5.19: Performances of Models on Restricted Datasets

Model	Validation MSE	Validation DA	Testing MSE	Testing DA
$\mathcal{M}_{\text{SVR, Mondays}}^{\text{MSE}}$	0.659	52.984%	0.943	41.176%
$\mathcal{M}_{\text{SVR, Mondays}}^{\text{DA}}$	0.681	54.887%	1.042	44.118%
$\mathcal{M}_{\text{RF, Mondays}}^{\text{MSE}}$	0.655	55.574%	0.919	47.059%
$\mathcal{M}_{\text{RF, Mondays}}^{\text{DA}}$	0.672	56.461%	0.939	55.882%
$\mathcal{M}_{\text{LSTM, Mondays}}^{\text{MSE}}$	0.484	51.479%	0.971	57.143%
$\mathcal{M}_{\text{LSTM, Mondays}}^{\text{DA}}$	0.673	56.819%	1.080	42.857%
$\mathcal{M}_{\text{SVR, Other Days}}^{\text{MSE}}$	5.575	52.605%	4.762	53.289%
$\mathcal{M}_{\text{SVR, Other Days}}^{\text{DA}}$	5.583	52.631%	4.776	53.289%
$\mathcal{M}_{\text{RF, Other Days}}^{\text{MSE}}$	5.606	50.997%	4.799	46.711%
$\mathcal{M}_{\text{RF, Other Days}}^{\text{DA}}$	6.844	52.449%	5.499	51.974%
$\mathcal{M}_{\text{LSTM, Other Days}}^{\text{MSE}}$	4.947	51.255%	4.749	53.290%
$\mathcal{M}_{\text{LSTM, Other Days}}^{\text{DA}}$	5.761	54.836%	4.760	53.290%

Chapter 6

Conclusions

Concluding Remarks Main Findings Before answering the two central research questions, this paper analyzes crude oil returns and shows that the series of returns only exhibits weak intertemporal correlation. As a result, it is tough to forecast crude oil returns using historical returns only. In contrast, the series of news sentiments shows significant intertemporal correlation and predictable patterns. Altogether the significant relationship between these two time series, it is reasonable to conjecture that one can forecast crude oil returns by using sentiments as a bridge.

In the experiment section, this paper firstly examines the market efficiency hypothesis (MEH) on the crude oil market with respect to different combinations of information sets, predictive models, and search technologies. In general, the market shows efficiency (i.e., unpredictability) with respect to simple models such as moving average predictors and ARIMA models. Still, returns are predictable using more sophisticated models such as LSTM-RNNs. As for the second research question, it turns out that news sentiments help predict returns only in one single case in which LSTM-RNN is used. In most cases, incorporating news sentiments, even worsen models' test time performances. Such poor performance is not a surprising result. Firstly, most of the news of day t arrives before the market is close, and if the price adjusts based on news quickly and accurately, the close spot price of the day t has already reflected a great percentage of news on the day t . At the beginning of day $t + 1$ when we gather news on day t (the information flow of day t) and predict r_{t+1} (equivalently, predict p_{t+1}), only the small portion of news

that arrive after market closes on day t is the novel content in the information flow. Consequently, adding daily information flow and news sentiments is too slow for the fast-evolving market and does not help much on forecasting daily returns. Secondly, recall that the models presented in this paper use information from the past 31 trading days to forecast the return on the upcoming trading day. In total, there are 385 sentiment features, but only 31 lagged values of returns used to generate one prediction. It could be the case that there are far too many sentiment features than necessary so that these noisy predictors obscure any information from historical returns. These two conjectures explain why adding news sentiments lead to an even worse test-time performance.

Limitations and Extensions Even though this paper answers its research questions, the analysis in this paper can still be extended in many ways. At the beginning of this paper, we define a market to be efficient with respect to an information set, a class of models, and a searching technology if no trading strategy based on the optimal model can attain positive (expected) economic profit. Throughout this paper, we claim the market to be inefficient whenever we found the market is predictable using some models. However, accurate predictions can only guarantee profit only if transaction costs are zero, which is a strong assumption. Incorporating a non-negligible transaction cost could flip some of the conclusions in this paper: the market could be efficient even if it is predictable. More robust conclusions can be drawn by using back-testing techniques. After choosing an optimal predictive model, we can design some trading strategies based on the model's prediction. Then the back-testing system simulates real-world trading based on the planned trading strategy and calculates the profit. Gaining positive profit in multiple rounds of simulation provides more robust evidence on market inefficiency than just showing there exists models can attain accuracy higher than 50

Another limitation is that this paper uses intraday high-frequency trading data and concludes the market can react to news quickly so that adding news sentiments does not produce an advantage in forecasting daily returns. Experimental results in this paper only imply that the market's reaction time is faster than 24 hours; further analysis using intraday oil returns can produce more meaningful results. Potentially, one can build a profitable forecasting system for the crude oil market on an intraday frequency.

The characteristic function is an essential component in the proposed framework. In this paper, characteristic functions only summarize information flows using simple statistics such as mean and standard deviation of sentiment scores. It turns out that too many features are generated, and these constructed sentiment features are too noisy. As a result, adding these sentiment data does not help improve models' performances. Future works can include more sophisticated and meaningful technical indicators built on news sentiments such as the moving average of daily sentiments. Informative content, if any, within news sentiments can be fully captured using only a few technical indicators and help handle the curse of dimensionality.

To analyze the usefulness of news sentiments, this paper looks at two information sets only (historical returns v.s. both historical returns and news sentiments). Including other macroeconomic indicators as additional features improve performances of models implemented in this paper. For example, crude oil and gold are substitutes as commodities for investments. As a result, changes in gold prices would affect crude oil prices and hence contain useful information to forecast crude oil returns. It is worthy of devoting more time to design models incorporating other exogenous features.

The performance of a time series forecasting model depends on whether it can successfully grasp two key factors: intertemporal correlations and non-linearities. This paper examines ARIMA models which can capture intertemporal relationship but not non-linearities, random forests and support vector machines that can model non-linear patterns but not intertemporal correlation, and LSTM-RNNs capturing both factors. It turns out that only LSTM-RNNs attain a satisfactory and potentially profitable result on the test set, which suggests that both elements are necessary for a successful forecasting system. However, LSTM-RNN is not the only model to handle two factors at the same time. Other models, such as CNN-RNN, can capture these two factors as well and performs much better on high dimensional problems.

Bibliography

- Baker, Malcolm, and Jeffrey Wurgler. 2006. “Investor Sentiment and the Cross-Section of Stock Returns”. *The Journal of Finance* 61 (4): 1645–1680. ISSN: 0022-1082. doi:10.1111/j.1540-6261.2006.00885.x.
- Baumeister, Christiane, and Lutz Kilian. 2016. “Forty Years of Oil Price Fluctuations: Why the Price of Oil May Still Surprise Us”. *Journal of Economic Perspectives* 30 (1): 139–160. ISSN: 0895-3309. doi:10.1257/jep.30.1.139.
- Bodnaruk, Andriy, Tim Loughran, and Bill McDonald. 2015. “Using 10-K Text to Gauge Financial Constraints”. *Journal of Financial and Quantitative Analysis*.
- Boser, Bernhard E, Isabelle M Guyon, and Vladimir N Vapnik. 1992. “A training algorithm for optimal margin classifiers”. *Proceedings of the fifth annual workshop on Computational learning theory*: 144–152. doi:10.1145/130385.130401.
- Brandt, Michael W, and Lin Gao. 2019. “Macro fundamentals or geopolitical events? A textual analysis of news events for crude oil”. *Journal of Empirical Finance* 51 (J. Finance 59 3 2004): 64–94. ISSN: 0927-5398. doi:10.1016/j.jempfin.2019.01.007.
- Breiman, Leo. 2001. “Random Forests”. *Machine Learning* 45 (1): 5–32. ISSN: 0885-6125. doi:10.1023/a:1010933404324.
- Bybee, Leland, et al. 2019. “The Structure of Economic News”. *SSRN Electronic Journal*. doi:10.2139/ssrn.3446225.
- Claeskens, Gerda, and Nils Lid Hjort. 2008. *Model Selection and Model Averaging*. Cambridge Books. Cambridge University Press. ISBN: 9780521852258. <https://ideas.repec.org/b/cup/cbooks/9780521852258.html>.

- Deeney, Peter, et al. 2015. "Sentiment in oil markets". *International Review of Financial Analysis* 39:179–185. ISSN: 1057-5219. doi:10.1016/j.irfa.2015.01.005.
- Drucker et al. 1997. "Support vector regression machines". *Advances in neural information processing systems*: 155–161.
- Fama, Eugene F. 1970. "Efficient Capital Markets: A Review of Theory and Empirical Work". *The Journal of Finance* 25:383–417.
- . 1991. "Efficient Capital Markets: II". *The Journal of Finance* 46 (5): 1575–1617. ISSN: 0022-1082. doi:10.1111/j.1540-6261.1991.tb04636.x.
- Friedman, Jerome H. 1997. "On Bias, Variance, 0/1—Loss, and the Curse-of-Dimensionality". *Data Mining and Knowledge Discovery* 1 (1): 55–77. ISSN: 1384-5810. doi:10.1023/a:1009778005914.
- Gibbons, Michael R, and Patrick Hess. 1981. "Day of the Week Effects and Asset Returns". *The Journal of Business* 54 (4): 579. ISSN: 0021-9398. doi:10.1086/296147.
- Hecht-Nielsen. 1989. "Theory of the backpropagation neural network". *International 1989 Joint Conference on Neural Networks*: 593–605 vol.1. doi:10.1109/ijcnn.1989.118638.
- Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. "Long Short-Term Memory". *Neural Computation* 9 (8): 1735–1780. ISSN: 0899-7667. doi:10.1162/neco.1997.9.8.1735.
- Hodges, J L. 1958. "The significance probability of the smirnov two-sample test". *Arkiv för Matematik* 3 (5): 469–486. ISSN: 0004-2080. doi:10.1007/bf02589501.
- Hu, Ziniu, et al. 2018. "Listening to Chaotic Whispers": 261–269. doi:10.1145/3159652.3159690. eprint: 1712.02136.
- Jensen, Michael C. 1978. "Some anomalous evidence regarding market efficiency". *Journal of Financial Economics* 6 (2-3): 95–101. ISSN: 0304-405X. doi:10.1016/0304-405x(78)90025-9.
- Loughran, Tim, and Bill McDonald. 2016. "Textual Analysis in Accounting and Finance: A Survey". *Journal of Accounting Research* 54 (4): 1187–1230. ISSN: 0021-8456. doi:10.1111/1475-679x.12123.

- . 2011. “When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks”. *The Journal of Finance* 66 (1): 35–65. ISSN: 1540-6261. doi:10.1111/j.1540-6261.2010.01625.x.
- Mann, Janelle, and Peter Sephton. 2016. “Global relationships across crude oil benchmarks”. *Journal of Commodity Markets* 2 (1): 1–5. ISSN: 2405-8513. doi:10.1016/j.jcomm.2016.04.002.
- Mohammadi, Hassan, and Lixian Su. 2010. “International evidence on crude oil price dynamics: Applications of ARIMA-GARCH models”. *Energy Economics* 32 (5): 1001–1008. ISSN: 0140-9883. doi:10.1016/j.eneeco.2010.04.009.
- Mudinas, Andrius, Dell Zhang, and Mark Levene. 2019. “Market Trend Prediction using Sentiment Analysis: Lessons Learned and Paths Forward”. eprint: 1903.05440.
- Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. “Glove: Global Vectors for Word Representation”. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*: 1532–1543. doi:10.3115/v1/d14-1162.
- Roache, Shaun K, and Marco Rossi. 2010. “The effects of economic news on commodity prices”. *The Quarterly Review of Economics and Finance* 50 (3): 377–385. ISSN: 1062-9769. doi:10.1016/j.qref.2010.02.007.
- Smales, Lee A. 2014. “News sentiment in the gold futures market”. *Journal of Banking & Finance* 49:275–286. ISSN: 0378-4266. doi:10.1016/j.jbankfin.2014.09.006.
- Smirnov, Nikolai. 1939. “On the estimation of the discrepancy between empirical curves of distribution for two independent samples”. *Bulletin Moscow University* 2:3–16.
- Smola, Alex J., and Bernhard Schölkopf. 2004. “A tutorial on support vector regression”. *Statistics and Computing* 14 (3): 199–222. ISSN: 0960-3174. doi:10.1023/b:stco.0000035301.49549.88.
- Tetlock, Paul C. 2007. “Giving Content to Investor Sentiment: The Role of Media in the Stock Market”. *The Journal of Finance* 62 (3): 1139–1168. ISSN: 0022-1082. doi:10.1111/j.1540-6261.2007.01232.x.

Timmermann, Allan, and Clive W J Granger. 2004. "Efficient market hypothesis and forecasting". *International Journal of Forecasting* 20 (1): 15–27. ISSN: 0169-2070. doi:10.1016/s0169-2070(03)00012-8.

Ushakov, Nikolai G. 1999. *Selected Topics in Characteristic Functions*. ISBN: 9783110935981.

Chapter 7

Appendix

Table 7.1: All Categories of Positive News

Category	Number of Positive news
commodity-price-gain	22,893
commodity-futures-gain	11,648
supply-decrease-commodity	5,845
imports-up	2,705
commodity-buy-target	1,171
demand-increase-commodity	1,070
exports-down	1,020
spill-commodity	787
commodity-offer-target	429
demand-guidance-increase-commodity	375
price-target-upgrade	332
exports-guidance-down	217
technical-view-bullish	193
supply-guidance-decrease-commodity	186
imports-guidance-up	122
relative-strength-index-oversold	85
embargo	80
piracy	57
pipeline-bombing-attack	32
force-majeure-commodity	26
tanker-accident-commodity	17
export-tax-guidance-decrease	11

Table 7.2: All Categories of Negative News

Category	Number of negative news
commodity-price-loss	26,475
commodity-futures-loss	12,818
supply-increase-commodity	6,629
imports-down	2,017
exports-up	1,308
resource-discovery-commodity	1,179
technical-view-bearish	1,172
demand-decrease-commodity	650
demand-guidance-decrease-commodity	341
commodity-sell-target	303
supply-guidance-increase-commodity	268
price-target-downgrade	261
exports-guidance-up	208
technical-price-level-resistance-bearish	150
force-majeure-lifted-commodity	85
imports-guidance-down	75
export-tax-increase	29
drilling-commodity	27
export-tax-guidance-increase	24
facility-upgrade-output	21
import-tax-increase	18
relative-strength-index-overbought	16
embargo-lifted	12
import-tax-guidance-increase	9
facility-open-output	5
facility-accident-contained-commodity	4
import-tax	3
export-tax	3
facility-sale-output	3
hijacking-released-commodity	1
tax-break-ended	1

Table 7.3: Filtering using Event Sentiment Score

r	Num Negative	Num Neutral	Num Positive
0	50.59% (100.00%)	3.25% (100.00%)	46.15% (100.00%)
1	50.57% (99.96%)	3.29% (101.24%)	46.13% (99.96%)
2	50.55% (99.91%)	3.33% (102.33%)	46.12% (99.93%)
3	50.52% (99.85%)	3.39% (104.08%)	46.09% (99.87%)
4	50.51% (99.83%)	3.82% (117.33%)	45.68% (98.96%)
5	50.20% (99.23%)	5.24% (161.14%)	44.55% (96.54%)
6	50.04% (98.91%)	5.42% (166.77%)	44.54% (96.49%)
7	50.01% (98.85%)	5.47% (168.07%)	44.52% (96.46%)
8	49.99% (98.81%)	5.51% (169.27%)	44.50% (96.42%)
9	48.88% (96.62%)	6.82% (209.80%)	44.29% (95.97%)
10	48.84% (96.53%)	6.91% (212.45%)	44.25% (95.88%)
11	48.82% (96.49%)	6.97% (214.20%)	44.21% (95.80%)
12	48.78% (96.41%)	7.86% (241.51%)	43.37% (93.96%)
13	48.74% (96.33%)	7.92% (243.55%)	43.34% (93.90%)
14	48.72% (96.29%)	7.96% (244.64%)	43.33% (93.87%)
15	11.93% (23.58%)	44.76% (1376.20%)	43.31% (93.83%)
16	11.88% (23.49%)	44.83% (1378.41%)	43.28% (93.78%)
17	11.85% (23.42%)	44.89% (1379.97%)	43.27% (93.74%)
18	11.82% (23.37%)	77.24% (2374.59%)	10.94% (23.71%)
19	11.80% (23.33%)	77.27% (2375.51%)	10.93% (23.68%)
20	11.73% (23.18%)	77.41% (2379.79%)	10.87% (23.55%)
21	11.42% (22.57%)	77.82% (2392.47%)	10.76% (23.32%)
22	5.69% (11.25%)	83.65% (2571.83%)	10.66% (23.09%)
23	5.57% (11.00%)	83.86% (2578.38%)	10.57% (22.90%)
24	5.53% (10.94%)	89.23% (2743.37%)	5.24% (11.34%)
25	5.41% (10.70%)	89.43% (2749.47%)	5.16% (11.17%)
26	5.37% (10.62%)	89.52% (2752.20%)	5.11% (11.07%)
27	5.32% (10.51%)	89.65% (2756.25%)	5.03% (10.91%)
28	4.23% (8.37%)	91.79% (2822.05%)	3.98% (8.62%)
29	4.21% (8.33%)	91.86% (2824.12%)	3.93% (8.51%)
30	4.18% (8.27%)	91.90% (2825.38%)	3.92% (8.49%)

Table 7.4: Filtering using Weighted Event Sentiment Score

r	Num Negative	Num Neutral	Num Positive
1	46.54% (100.00%)	9.06% (100.00%)	44.40% (100.00%)
1	39.25% (84.33%)	20.32% (224.32%)	40.43% (91.06%)
2	33.69% (72.40%)	29.93% (330.42%)	36.37% (81.92%)
3	31.12% (66.87%)	34.62% (382.22%)	34.25% (77.14%)
4	28.35% (60.92%)	40.08% (442.46%)	31.57% (71.10%)
5	25.24% (54.24%)	46.49% (513.20%)	28.27% (63.66%)
6	24.92% (53.54%)	49.89% (550.79%)	25.19% (56.73%)
7	21.78% (46.79%)	53.19% (587.19%)	25.03% (56.38%)
8	21.42% (46.04%)	57.06% (629.93%)	21.51% (48.45%)
9	18.09% (38.87%)	60.76% (670.80%)	21.15% (47.62%)
10	17.51% (37.64%)	61.39% (677.67%)	21.10% (47.52%)
11	17.46% (37.53%)	65.26% (720.44%)	17.28% (38.91%)
12	14.07% (30.23%)	69.17% (763.62%)	16.76% (37.75%)
13	13.25% (28.47%)	70.03% (773.05%)	16.72% (37.66%)
14	13.15% (28.25%)	74.32% (820.49%)	12.53% (28.22%)
15	9.83% (21.13%)	77.68% (857.58%)	12.48% (28.11%)
16	9.66% (20.76%)	77.96% (860.57%)	12.38% (27.88%)
17	8.29% (17.82%)	79.37% (876.19%)	12.34% (27.79%)
18	8.18% (17.57%)	84.22% (929.75%)	7.60% (17.12%)
19	8.06% (17.31%)	84.49% (932.75%)	7.45% (16.78%)
20	7.98% (17.15%)	84.63% (934.20%)	7.39% (16.65%)
21	7.51% (16.14%)	85.37% (942.46%)	7.12% (16.03%)
22	4.77% (10.24%)	88.20% (973.67%)	7.03% (15.84%)
23	4.66% (10.01%)	88.42% (976.11%)	6.92% (15.58%)
24	4.48% (9.63%)	91.35% (1008.46%)	4.17% (9.39%)
25	4.22% (9.06%)	91.95% (1015.06%)	3.83% (8.63%)
26	4.16% (8.95%)	92.06% (1016.33%)	3.77% (8.50%)
27	4.09% (8.79%)	92.24% (1018.25%)	3.67% (8.27%)
28	3.28% (7.04%)	93.86% (1036.13%)	2.86% (6.45%)
29	2.95% (6.34%)	94.23% (1040.22%)	2.82% (6.35%)
30	2.92% (6.27%)	94.31% (1041.09%)	2.77% (6.25%)

Table 7.5: News on Bloomberg June 12, 2019

PALM OIL: Futures Drop to 6-Month Low as Crude Oil Prices Slump
OIL DAYBOOK AMERICAS: Saudis Sell Extra Crude to China; EIA Data
Oil Stocks Fall Most in Europe as Rising Stockpiles Hit Crude
Oil Extends Slide After EIA Reports Surprise U.S. Crude Inventory Increase
U.S. DOE Crude Oil Inventories, Production and Imports (Table)
Canadian Stocks Fall After Crude Oil Plunges as Storage Swells
OIL DAYBOOK ASIA: Oil Demand Shrivels Amid U.S.-China Trade War
DJ News Highlights: Top Energy News of the Day
Enterprise Sees U.S. Oil Exports Topping Saudi Arabia's by 2025
Market Realist: Commodities Are Mixed Early on June 18
China Times Crude Purchases Well During Oil-Price Slump
TOPLive Starts: Analysis of the EIA Crude Oil Inventory Report
Malaysia May Palm Oil Stockpiles -10.3% M/m to 2.45M Tons
RTT News: Crude Oil Futures Plunge Sharply, Settle At 5-month Low
Oil Plunges as U.S. Storage Swells, Demand Outlook Worsens
Copper and Crude Oil 'Do-or-Die' Elevate Macroeconomic Risks
Malaysia May Crude Palm Oil Prices -3.6% M/m at MYR1,947/Ton
German Baltic Port Boosts Crude Imports to Offset Druzhba Loss
Energy Stocks Teeter Along With Oil After Inventory Build
Oil Holds Near Lows on U.S. Stockpile Build, Poor Demand Outlook
Saudis Heed China Request for More Oil as Supplies Squeezed
OIL BRIEF: OPEC Makes First Nod to Curbs in 2020
Action Forex: Crude Oil: Oil Trading Lower, Ahead Of EIA's Weekly Crude Oil Stockpiles Data
Oil & Gas 360: Crude Oil Inventories Up 2.2 Million Barrels
Watch Oil Stocks as Higher Inventories, Trade Jitters Hit Crude
WTI Crude Oil Is Teetering on the Cliff's Edge
Crude oil futures fall on weak global cues
UNIAN: Reuters: Oil falls 1% on weaker oil demand growth, surprise gain in U.S. crude stocks
Times Now: Crude oil prices down more than 2% on US inventories, demand worries
Xinhua: Oil prices decline on increasing U.S. crude inventories
Crude Oil WTI (F) Has Changed Elliott Wave Count
Crude Oil Brent Has Changed Elliott Wave Count
ASIA CRUDE: Chinese Buyer Gets Extra Saudi Oil; Iran July OSPs
World Oil Inventory Days Above 10-Year Average
/Crude Oil Capitulation on Tap
HoustonChronicle: Nation's crude oil supplies jump again as oil prices slip further
S. Korea's Hyundai Oilbank Cuts Oil Processing Rate by About 3%
Bonds Gain as Crude Price Slides Ahead of CPI Data: Inside India
*CRUDE OIL INVENTORIES ROSE 2.21 MLN BARRELS, EIA SAYS
Oil's 2019 Weakness Has Roots in 2018's Strength: Liam Denning
EIA Stockpile Hits 20-Month High Amid Subdued Demand: Lee
OIL MKT DRIVERS: API Reports Build; OPEC Latest; Demand Worries
OIL DAYBOOK EUROPE: OPEC+ 'Very Close' on Cuts Extension; EIA
Bus Day Nigeria: OPEC crude oil output shrinks in May, organisation struggles to avoid \$40 oil
[Delayed] J.P. Morgan 2019 Energy Equity Conference: The Question Bank
Saudi Arabia Economic Report - 2019 - English
[Delayed] MLP 1Q19 PM Handbook: Earnings Defies Fears with Operating Leverage & Optimization on Display; Much Needed Signs of Halyk Finance Daily (ENG) - June 12, 2019
Sunwah Kingsway - Willas-Array (854 HK)
BOBCAPS First Light Monthly Eco Chartbook, Inflation and IIP

Table 7.6: News on Bloomberg June 13, 2019

-
- ✓Market Realist: US Crude Oil Near \$50: Why Denbury Resources Might Be in Trouble
✓Oil-Services Benchmark Rebounds Along With U.S. Crude: Chart
Watch Oil Stocks as Brent Surges After Report of Tanker on Fire
✓WTI Crude Oil \$80 Resistance, Then \$70, Now \$60 Suggests a Trend
PALM OIL: Futures Rally From 6-Month Low as Crude Oil, Soy Jumps
Oil is surging after a suspected torpedo attack on 2 tankers in the Gulf of Oman
Oil Slump Forces Norwegian Producer OKEA to Slash IPO Price
Oil Stocks Rise Following Jump in Crude After Tankers Attacked
✓OPEC May Crude Oil Production in OPEC Countries (Table)
AP NewsAlert: Benchmark Brent Crude Oil Rises 4% in Trading to Over \$62 a Barrel After Oil Tanker Incident in Gulf of Oman
Oil Prices Spike As Navy Rushes To New Attack But OPEC Flashes This Warning
Staring at a Recessionary Abyss, Crude Oil Needs an OPEC+ Jolt
DJ Energy Calendar - 2019 Futures, Options Dates - Jun 13
Oil Stocks Climb With Crude After Tanker Attack: EU Energy Wrap
DJ Oil Jumps After Tanker Attacks in Middle East -- 2nd Update
RUBBER: Futures Reverse Losses in Singapore as Crude Oil Jumps
DJ News Highlights: Top Energy News of the Day
U.S. Stocks Climb With Treasuries; Crude Oil Jumps: Markets Wrap
✓OPEC Allies May Need to Deepen, Extend Crude Oil Supply Cuts
✓OIL BRIEF: Crude Surges as Gulf Tankers Suffer New Attack
Cartagena Oil Refinery Crude Unit Restarted, Spain's Repsol Says
Bus Day Nigeria: Crude oil build up sends price lower
Hormuz Attacks May Raise Oil's Premium Despite Risk of Slowdown
India Infoline: Energy Preview: Crude Melts 3.50% On MCX
RTT News: Oil Futures Settle Sharply Higher After Attacks On Oil Tankers
Europe Stocks Recover From Earlier Losses as Oil Shares Advance
Times Now: Crude oil slumps 4% on US crude build, slowing demand fears
URGENT: The Latest: Crude Over \$62 a Barrel After Mideast Incident
Oil Surges on Mideast Tanker Incident, Stocks Rise
Tanker Attacks Are Threatening the World's Most Important Oil Route
How Tanker Attacks on a Skinny Waterway Could Affect Oil Prices
Tanker Attack in Gulf of Oman Sends Oil Price Soaring
Xinhua: U.S. oil imports, exports down last week: EIA
HoustonChronicle: Crude-oil marketing and transportation firm rides rising tide of oil prices
Investing.com: Oil Prices Recover Slightly After Plunging 4% on Crude Inventories Data
Market Realist: Why Diamondback Energy Has Been an Outperformer This Quarter
DJ ICE Gas Oil/Brent Crude Oil/WTI Open Interest - Jun 13
Persian Gulf Remains Bedrock for VLCC Loadings
\$40 WTI Crude Is More Likely Than \$60
Action Forex: Crude Oil: Oil Trading Lower In The Asian Session
DJ ICE Brent Crude Oil Volume - Jun 13
Oil Jumps From Near Five-Month Low After Report Tanker on Fire
COMMODITIES
Crude Prices Rebound From Five-month Low as Tanker Attack Fuels Fears Over Key Oil Artery
DBH: Nigeria earns \$236bn from petroleum exports in five years
Too much supply uncertainty; new highs point to 465, 472 3/4 soon - HIGHTOWER
[Delayed] Iron ore: Upgrading Chinese steel demand and iron ore prices, but the peak disruption is likely behind us
FW: LatAm Daily