Sentiment Analysis and Stock Prices: A Novel Predictive Approach using Markov Models

Pratim Chowdhary, Karun Ram, Allison Zhuang

Abstract

This paper aims to use Markov models as a means to modeling financial trends based on sentiment analysis on tweets and news. The process of gathering data and performing sentiment analysis is given with the majority of the paper focusing on using this data to evaluate how these matrices can be used to forecast future returns. An implementation of Markov prediction using a maximum *a posteriori* (MAP) estimate, the Viterbi model, is also given along with results of both an a priori and a posteriori approach. The results are very promising, with the model performing above expectations and the majority of the models obtaining above a 90% accuracy.

Table of Contents

Introduction

Financial modeling– that is, predicting the impact of stock or options trades on price and volume– is at the forefront of research in applied mathematics and computer science. There are a variety of techniques used to predict the prices of stocks, ranging from qualitative analysis to large-scale neural networks. If one can successfully predict the outcomes of a stock exchange with certainty, the financial winnings they stand to receive are practically limitless. Successful trading on the stock market on a reactionary basis (often called day trading) is arguably one of the highest-risk, yet highest-reward financial pursuits.

Undoubtedly, the value of stocks on the market is strongly linked to the opinions of buyers and sellers on the market; a stock's price rises if there are more buyers than sellers, and participants in the market believe that the company will do well in the future. This may be if the company recently had impressive returns, or if buyers agree with the company's recent business decisions, have faith in the company's leaders, or any variety of reasons. Alternatively, a stock's price will typically fall if the company has not had impressive returns, the company or any of its leaders have recently been involved in a scandal, or even general panic over the state of the economy or world.

Our research aims to consider the specific impact of media sentiment across forms of media on market outcomes of various stocks, such as price and trade volume, in order to aid financial decision-making.

Data

Data Sources:

The data required for this modeling problem was procured from historical data using Twitter API and web-scrapers. These datasets were readily compiled and available; this paper uses the following compiled datasets:

- **Tweet Dataset:** "Tweets about the Top Companies from 2015 to 2020," compiled by Kaggle users Mustafa Doğan and Ömer Metin
- **News Dataset:** "Historical financial news archive," compiled by Kaggle user GennadiyR
- **Stock Prices Dataset:** "Stock Market Data (NASDAQ, NYSE, S&P500)," compiled by Kaggle user Paul Mooney

Data Preprocessing:

The data used for this paper spanned five years (January 1, 2015 to December 31, 2019). Five stocks (AAPL, GOOG, AMZN, TSLA, MSFT) were chosen from the datasets since they contained sufficient data across tweets and news datasets. The Natural Language Toolkit's Valence Aware Dictionary and sEntiment Reasoner (VADER) was selected as a sentiment analyzer since its training data allows it to perform well on short bodies of text such as tweets and news headlines. NTLK-VADER provides the text with a sentiment score in the range [-1, 1]. The tweets were then separated by ticker and binned individually (each ticker was binned separately) by quintile.

Data Summary:

Presented below are summaries of the datasets, after filtering based on the date range and stock list and computing sentiments:

Tweet Dataset, N = 3.72m tweets					
Distribution by Stock		**Distribution by Year**		**Distribution by Sentiment**	
Stock Ticker	# Tweets (millions)	Year	# Tweets (millions)	Sentiment Score	# Tweets (millions)
AAPL	1.42	2015	0.73	[-1.0, -0.6)	0.12
AMZN	0.72	2016	0.84	[-0.6, -0.2)	0.43
GOOG	0.72	2017	0.62	[-0.2, +0.2)	1.91
TSLA	1.10	2018	0.77	[+0.2, +0.6)	0.82
MSFT	0.37	2019	0.76	[+0.6, +1.0]	0.44

News Dataset, N = 39.7k headlines					
Distribution by Stock		**Distribution by Year**		**Distribution by Sentiment**	
Stock Ticker	# Headlines (thousands)	Year	# Headlines (thousands)	Sentiment Score	# Headlines (thousands)
AAPL	18.1	2015	2.83	[-1.0, -0.6)	0.95
AMZN	5.84	2016	4.81	[-0.6, -0.2)	5.03
GOOG	4.60	2017	10.1	[-0.2, +0.2)	21.9
TSLA	3.82	2018	9.74	[+0.2, +0.6)	9.73
MSFT	7.23	2019	12.1	[+0.6, +1.0]	2.08

Method

Summary:

Two related models were developed in order to enable this model to predict returns for stock prices. The first is an *a posteriori* method, using sentiment data sequences to estimate stock returns over the timescale that the sentiment data sequence originates from. The second is a predictive, or *a priori*, method, designed to predict future stock returns based on prior stock data. The foundation of these models lies in Markov models and Hidden Markov Models.

Assumptions and Simplifications:

A list of assumptions and simplifications made in the model presented is listed below:

(1) Markov Assumption for Sentiments– the transitions between sentiment states are probabilistically independent of time, solely dependent on the previous state.

(2) Markov Assumption for Returns– the transitions between return states are probabilistically independent of time, solely dependent on the previous state.

(3) Markov Assumption for Emissions– the probability that a given sentiment occurs is an emission function that depends on the given stock return state, not on time.

(4) Assumption of Association– there is an association between sentiments and stock returns (heuristically confirmed)

(5) Discretization– the data collected on sentiments and stock returns were binned into discrete states based on quintiles, because discrete data is more conducive to Markov analysis

Keeping these assumptions in mind, we build our model.

Model Construction:

To construct the models, we first define the following:

States	Let $S = \{s_i \mid i \in [0..4]\}$ be the set of sentiment states. Let $R = \{r_i \mid i \in [0..4]\}$ be the set of return states.
Data (binned, daily average, time series)	Let $A = (a_1, a_2, a_3, \ldots, a_n)$ be a sequence of states, where $a_i \in S \ \forall \ i$. Let $B = (b_1, b_2, b_3, \ldots, b_n)$ be a sequence of states, where $b_i \in R \ \forall \ i$.

We then build the following matrices, using the sequences a_n and b_n:

$$M_S\text{: } (M_S)_{ij} = \mathrm{P}(a_{n+1} = s_j \mid a_n = s_i)$$

$$M_R\text{: } (M_R)_{ij} = \mathrm{P}(b_{n+1} = r_j \mid b_n = r_i)$$

$$E\text{: } \ (E)_{ij} = \mathrm{P}(b_n = \ r_j \mid a_n = s_i)$$

These matrices are sufficient to define a pair of associated models:

1. Let M be a Markov chain with transition probabilities defined by M_S.
2. Let H be a hidden Markov model defined such that the return states, whose transition probabilities are governed by M_R, are the hidden states; and the emission probabilities, from return to sentiment states, are governed by E.

See Appendix A for the numerical matrices for the model computed from the data provided.

***A Posteriori* Prediction:**

We propose the following procedure for an *a posteriori* estimate of returns in the time interval [n, n+k-1]. We define the function

$$f_{M,\,H} : (a_n, a_{n+1}, \ldots, a_{n+k-1}) \rightarrow \mathbb{R}$$

to represent the following procedure:

1. Use a maximum a posteriori (MAP) estimate (e.g. the Viterbi algorithm[1]) to compute the highest-likelihood sequence V of hidden states $(v_n, v_{n+1}, \ldots, v_{n+k-1})$ corresponding to the inputted sequence of observed states $(a_n, a_{n+1}, \ldots, a_{n+k-1})$ in H, where $v_i \in R \ \forall \ i$.
2. Compute the expected return that arises from V by multiplying the midpoint of the bin for each state (an estimate for the expected return) of each state in V.

A Priori Prediction:

We propose a similar procedure for an *a priori* prediction (forward-looking) estimate of returns in the time interval $[n+1, n+k]$ given sentiment data in the time interval $[n-4k+1, n]$. We define the function

$$g_{M,H} : (a_{n-4k+1}, a_{n-4k+2}, \ldots, a_n) \rightarrow \mathbb{R}$$

to represent the following procedure:

1. Compute the probability vector Ψ where $\Psi_i = \mathrm{P}(a_k = S_i \mid k \in [n-4k+1, n])$.
2. Generate a list L of random walks on the Markov chain M of length k, using the starting state distribution Ψ.
3. For each random walk l_i in L, use a maximum a posteriori (MAP) estimate (e.g. the Viterbi algorithm) to compute the highest-likelihood sequence V_i of hidden states $(v_n, v_{n+1}, \ldots, v_{n+k-1})$ corresponding to the inputted sequence of observed states $(a_n, a_{n+1}, \ldots, a_{n+k-1})$ in H, where $v_i \in R \ \forall \ i$.
4. For each sequence V_i, compute the expected return that arises from V_i by multiplying the midpoint of the bin for each state (an estimate for the expected return) of each state in V_i.
5. Compute the average expected return over all V_i.

[1] https://www.sciencedirect.com/topics/mathematics/viterbi-algorithm

Results

The most meaningful way to evaluate the effectiveness of the models for future return forecasting is to compare predicted returns (i.e. using both $f_{M,H}$ and $g_{M,H}$) to the actual returns using historical data. The data was tested on the dataset provided. For this paper, the *a posteriori* model (i.e. $f_{M,H}$) used the parameter $k = 7$. The *a priori* model (i.e. $g_{M,H}$) also used the parameter $k = 7$, which implied that it sampled from a prior period of 28 days. The following tables contain the accuracy of the models for each stock evaluated.

***A posteriori* model**		
Stock	Tweet Model Accuracy (%)	News Model Accuracy (%)
AAPL	95.67	79.90
AMZN	96.24	96.42
GOOG	97.03	97.11
MSFT	96.91	97.09
TSLA	91.66	90.90

***A priori* model**		
Stock	Tweet Model Accuracy (%)	News Model Accuracy (%)
AAPL	95.72	80.90
AMZN	96.35	96.40
GOOG	97.10	97.12
MSFT	96.93	97.10
TSLA	91.79	90.94

Conclusion and Evaluation

Evaluating accuracy rates across the *a posteriori* and *a priori* models, across the five chosen stocks, and across news and tweets, it can be seen that the predictive model works very well on changes in the price of AMZN, GOOG, and MSFT, over both news and tweets. Predictions are slightly less accurate for TSLA, but still over 90% accurate over both news and tweets.

Oddly, the model has very high accuracy for changes in AAPL stock price when using sentiment from tweets, but much lower accuracy when using news headlines. This could be because the dataset for headlines simply contains far more for AAPL than any other stock; AAPL has the most, with about 18 thousand, whereas the stock with the second-most data, MSFT, has only 7.23 thousand. The excess of headlines on AAPL may have had a confounding influence on the prediction accuracy.

Overall, sentiment analysis of tweet and headline data provides promisingly accurate predictions of stock price; however, investigating certain aspects of the prediction process could improve accuracy. Firstly, sentiments were measured on the scale of days; the Markov property may not hold at that time scale. If, instead, sentiments could be measured based on hourly data, accuracy could improve.

Furthermore, discretization of the changes in stock price and sentiment, which was necessary to create separate states to use the Markov models, introduced error. Whenever stock price predictions were made, expected returns were computed based on the middle of the bin representing the predicted state. Though necessary to apply the Markov model, this created inexactness in predictions of changes in price.

Finally, it is possible to use different MAP estimation models to try for greater accuracy. For instance, it would theoretically be most accurate to compute the expected returns for all possible changes in stock price, weighted by their probability, based on the given change in sentiments; the Viterbi algorithm only used the most likely change in stock price. However, this would be too computationally expensive to implement over any reasonably large dataset. It would be possible to instead compare the results from the Viterbi algorithm with k-best paths, random forest expectation, or comparable algorithms.

Given the accuracy of the featured model, however, it is possible to profit off of it by buying the stock before prices are predicted to rise, and selling before prices are predicted to fall. The results of this investigation indicate that it is indeed possible to predict changes in stock price based on external, observable factors. For instance, it would also be worthwhile to model a company's predicted change in stock price relative to its financials–earnings and profits, past prices, or prices of stocks or comparable companies.

It is also possible to try to model other attributes of a stock as the hidden state: for instance, as a more socially responsible application of HMMs to the stock market, one could use HMMs to predict companies' social impact or environmental friendliness through some other external, measurable metric–for instance, donation activity. This information would then allow investors to support more pro-social companies. As Warren Buffet wrote, "in the short run, the stock market is a voting machine." Investors could then express their moral values by investing in companies that most invest in the future. Mathematics, then, could be harnessed as a powerful source for social good in the financial world.

References

1. Gupta, Aditya, and Bhuwan Dhingra. “Stock Market Prediction Using Hidden Markov Models.” Duke Computer Science Department, 2012, https://users.cs.duke.edu/~bdhingra/papers/stock_hmm.pdf.
2. Prasetio, Barlian Henryranu, and Tamura, Hiroki, and Tanno, Koichi. “Deep time-delay Markov network for prediction and modeling the stress and emotions state transition.” Nature Publishing Group, 22 October 2020, https://www.nature.com/articles/s41598-020-75155-w
3. Metin, Ömer, et al. “Tweets about the Top Companies from 2015 to 2020.” Kaggle, 2020, https://www.kaggle.com/omermetinn/tweets-about-the-top-companies-from-2015-to-2020
4. Gennadiyr, “Historical financial news archive.” Kaggle, 2020, https://www.kaggle.com/gennadiyr/us-equities-news-data

Appendix A: Model Construction Data

AAPL

Tweets

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.478	0.522	0.0	0.0	0.0
S_1	0.024	0.717	0.258	0.0	0.0
S_2	0.002	0.198	0.665	0.134	0.002
S_3	0.0	0.014	0.336	0.641	0.009
S_4	0.0	0.0	0.0	0.75	0.25

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.143	0.714	0.0	0.143
S_1	0.006	0.283	0.675	0.033	0.003
S_2	0.003	0.25	0.731	0.016	0.0
S_3	0.04	0.44	0.52	0.0	0.0
S_4	0.5	0.0	0.5	0.0	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.571	0.429	0.0	0.0
R_1	0.018	0.432	0.438	0.112	0.0
R_2	0.019	0.327	0.452	0.198	0.004
R_3	0.0	0.44	0.44	0.12	0.0
R_4	0.0	0.5	0.0	0.5	0.0

News

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.036	0.286	0.5	0.179	0.0
S_1	0.023	0.158	0.492	0.299	0.028
S_2	0.023	0.131	0.54	0.286	0.019
S_3	0.022	0.148	0.481	0.312	0.038
S_4	0.0	0.065	0.452	0.484	0.0

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.143	0.714	0.0	0.143
S_1	0.0	0.306	0.657	0.037	0.0
S_2	0.008	0.246	0.732	0.014	0.001
S_3	0.0	0.375	0.625	0.0	0.0
S_4	0.0	0.0	1.0	0.0	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.429	0.571	0.0	0.0
R_1	0.018	0.135	0.544	0.275	0.028
R_2	0.024	0.143	0.496	0.312	0.025
R_3	0.042	0.125	0.625	0.208	0.0
R_4	0.0	0.0	1.0	0.0	0.0

AMZN

Tweets

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.0	0.5	0.5	0.0
S_1	0.0	0.2	0.667	0.133	0.0
S_2	0.004	0.036	0.601	0.358	0.002
S_3	0.0	0.007	0.283	0.694	0.015
S_4	0.0	0.0	0.0	0.647	0.353

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.143	0.429	0.286	0.143
S_1	0.017	0.1	0.483	0.367	0.033
S_2	0.005	0.045	0.801	0.145	0.003
S_3	0.005	0.033	0.737	0.221	0.005
S_4	0.0	0.286	0.571	0.143	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.286	0.429	0.286	0.0
R_1	0.0	0.033	0.667	0.3	0.0
R_2	0.002	0.023	0.421	0.539	0.015
R_3	0.0	0.019	0.376	0.596	0.009
R_4	0.0	0.0	0.429	0.571	0.0

News

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.067	0.267	0.467	0.2	0.0
S_1	0.051	0.121	0.424	0.343	0.061
S_2	0.008	0.085	0.502	0.36	0.045
S_3	0.01	0.094	0.447	0.413	0.036
S_4	0.021	0.146	0.396	0.312	0.125

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.143	0.571	0.286	0.0
S_1	0.02	0.16	0.52	0.3	0.0
S_2	0.005	0.034	0.792	0.162	0.006
S_3	0.006	0.061	0.726	0.196	0.011
S_4	0.143	0.429	0.429	0.0	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.0	0.286	0.714	0.0
R_1	0.02	0.04	0.56	0.38	0.0
R_2	0.015	0.102	0.478	0.355	0.049
R_3	0.011	0.095	0.413	0.43	0.05
R_4	0.0	0.0	0.429	0.571	0.0

GOOGL

Tweets

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.0	0.5	0.5	0.0
S_1	0.0	0.18	0.82	0.0	0.0
S_2	0.002	0.081	0.772	0.141	0.003
S_3	0.004	0.011	0.297	0.681	0.007
S_4	0.0	0.0	0.0	0.8	0.2

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.083	0.333	0.417	0.167
S_1	0.047	0.209	0.5	0.244	0.0
S_2	0.01	0.075	0.713	0.194	0.008
S_3	0.005	0.096	0.69	0.203	0.005
S_4	0.125	0.25	0.5	0.125	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.167	0.583	0.25	0.0
R_1	0.0	0.093	0.593	0.291	0.023
R_2	0.003	0.064	0.643	0.287	0.003
R_3	0.0	0.048	0.615	0.332	0.005
R_4	0.0	0.25	0.375	0.375	0.0

News

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.045	0.182	0.523	0.182	0.068
S_1	0.038	0.177	0.423	0.285	0.077
S_2	0.072	0.152	0.407	0.296	0.072
S_3	0.04	0.173	0.493	0.24	0.053
S_4	0.038	0.096	0.5	0.346	0.019

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.091	0.818	0.091	0.0
S_1	0.014	0.135	0.608	0.23	0.014
S_2	0.011	0.068	0.711	0.197	0.013
S_3	0.019	0.142	0.648	0.191	0.0
S_4	0.125	0.25	0.25	0.375	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.0	0.727	0.273	0.0
R_1	0.027	0.081	0.541	0.297	0.054
R_2	0.063	0.169	0.436	0.266	0.066
R_3	0.037	0.173	0.414	0.309	0.068
R_4	0.125	0.25	0.375	0.25	0.0

MSFT

Tweets

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.0	0.5	0.5	0.0
S_1	0.0	0.053	0.579	0.368	0.0
S_2	0.006	0.024	0.506	0.446	0.018
S_3	0.0	0.012	0.175	0.76	0.054
S_4	0.0	0.0	0.117	0.733	0.15

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.0	0.364	0.545	0.091
S_1	0.008	0.109	0.622	0.244	0.017
S_2	0.007	0.092	0.786	0.113	0.002
S_3	0.012	0.111	0.772	0.105	0.0
S_4	0.2	0.0	0.4	0.4	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.0	0.455	0.545	0.0
R_1	0.0	0.008	0.336	0.613	0.042
R_2	0.002	0.017	0.266	0.664	0.052
R_3	0.0	0.012	0.222	0.735	0.031
R_4	0.0	0.0	0.0	1.0	0.0

News

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.172	0.483	0.31	0.034
S_1	0.028	0.105	0.476	0.343	0.049
S_2	0.028	0.144	0.398	0.367	0.064
S_3	0.023	0.113	0.407	0.4	0.056
S_4	0.032	0.095	0.413	0.444	0.016

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.091	0.182	0.273	0.273	0.182
S_1	0.009	0.135	0.667	0.189	0.0
S_2	0.008	0.091	0.784	0.113	0.003
S_3	0.014	0.101	0.723	0.162	0.0
S_4	0.0	0.2	0.4	0.4	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.182	0.545	0.273	0.0
R_1	0.018	0.18	0.432	0.324	0.045
R_2	0.025	0.116	0.413	0.394	0.052
R_3	0.034	0.142	0.405	0.331	0.088
R_4	0.0	0.0	0.2	0.8	0.0

TSLA

Tweets

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.0	0.0	0.0	1.0	0.0
S_1	0.1	0.3	0.4	0.2	0.0
S_2	0.0	0.005	0.723	0.271	0.002
S_3	0.004	0.007	0.311	0.657	0.021
S_4	0.0	0.0	0.133	0.733	0.133

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.16	0.44	0.32	0.08	0.0
S_1	0.031	0.549	0.393	0.021	0.006
S_2	0.038	0.584	0.368	0.011	0.0
S_3	0.04	0.68	0.28	0.0	0.0
S_4	0.25	0.5	0.0	0.25	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.0	0.0	0.8	0.2	0.0
R_1	0.003	0.006	0.559	0.418	0.014
R_2	0.002	0.011	0.444	0.533	0.011
R_3	0.0	0.04	0.52	0.44	0.0
R_4	0.0	0.0	0.75	0.25	0.0

News

$M_S =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.065	0.194	0.419	0.258	0.065
S_1	0.025	0.191	0.414	0.293	0.076
S_2	0.023	0.169	0.47	0.273	0.065
S_3	0.039	0.142	0.422	0.312	0.085
S_4	0.067	0.107	0.413	0.293	0.12

$M_R =$

	S_0	S_1	S_2	S_3	S_4
S_0	0.071	0.595	0.31	0.024	0.0
S_1	0.04	0.544	0.391	0.02	0.005
S_2	0.047	0.58	0.354	0.017	0.003
S_3	0.0	0.684	0.316	0.0	0.0
S_4	0.0	0.75	0.0	0.25	0.0

$E =$

	S_0	S_1	S_2	S_3	S_4
R_0	0.048	0.286	0.405	0.214	0.048
R_1	0.033	0.167	0.455	0.278	0.067
R_2	0.025	0.135	0.434	0.309	0.097
R_3	0.053	0.158	0.316	0.421	0.053
R_4	0.25	0.25	0.5	0.0	0.0

Appendix B: Python Code

Go to https://github.com/cpratim/Stock-Sentiment-HMM to see the code used for this project.