Financial Analysis Using News Data

Wenbin Zhang, Steven Skiena
Department of Computer Science, Stony Brook University
Stony Brook, NY 11794-4400 USA
{wbzhang, skiena}@cs.sunysb.edu

Contents

Abstract

This paper surveys the field of financial analysis using news data, and also our own considerable work regarding movie gross analysis to show the predictive power of news data. Since the 1990s, linguistic sources such as news have been continuously proved to carry extra and meaningful information beyond traditional quantitative finance data, and thus they can be used as predictive indicators in finance. In the first half of this paper, we analyze and compare important previous research in terms of targets, methodologies, and results. In addition, we discuss financial news analysis with *Lydia*, a system for large-scale news analysis. In the second half of this paper, based on the news data generated by *Lydia*, we build three different models (regression, piecewise, and k-nearest neighbor models) and achieve good prediction results for movie grosses. Therefore, we believe that *Lydia* data could be used in forecasting other financial variables as well, for example, stock returns, volatilities, trading volumes, or firm earnings.

1 Introduction

1.1 What is the problem?

Nowadays, financial markets are playing a more and more important role in world business and our day-to-day lives. Financial markets provide a mechanism that allows people to easily buy and sell financial securities, commodities, or any other financial products at reasonable transaction costs and at prices that reflect the efficient market hypothesis. The financial securities could be stocks or bonds. The commodities we mentioned could be precious metals or agricultural goods. In order to invest wisely in financial markets, financial analysis techniques are increasingly used in this field, which aim to assess the viability, stability and profitability of a firm or its financial products.

There is a large body of literature studying how financial market prices incorporate quantitative data. The traditional finance model assumes investors are rational and act in their own self-interest, which always force capital market prices to converge to the rational present prices of these specific products. Therefore, their analysis and prediction are usually based on some hard-facts, e.g., operating and investing like mergers, acquisitions or IPOs, or financial variables, e.g., historical price data. However, a few researchers study the use of qualitative verbal information in financial analysis, such as news. Actually qualitative verbal information may come from diverse sources, like news, blogs, analysts reports, chat rooms or even small talk.

In fact, most finance investors do not have much chance to access firms' production or operational activities, therefore they make their investment decisions by some collected information, which is usually secondhand. There are three major sources for these kinds of information: analysts' forecast reports, publicly disclosed accounting variables, and linguistic information that contains firms' current and future information. We cannot always expect to get accurate predictions from analysts' reports and accounting variables alone, that's why people still need as much linguistic input as possible, which could be act as an incremental power to predict the fluctuation of financial

assessments, like earnings, returns, volatilities or future prices if they are interpreted properly.

Our objective is to analyze those linguistic sources with certain techniques, determine the underlying indicators in the financial market, and thus help to predict any financial parameters that we are interested in. Among all kinds of linguistic sources, news is the most important and widely used and therefore becomes our major source in this paper. Research on financial analysis from news is an interdisciplinary topic, which involves finance, econometrics, statistics, and computer science, including techniques such as natural language processing, data mining, and machine learning.

1.2 Why is it interesting?

News-based financial analysis is interesting because it can tell something that traditional financial analysis techniques cannot. There are compelling theoretical and empirical underlying reasons to do so. Theoretically, researchers in behavioral finance argue that investors are subject to sentiment. The existence of investor sentiment is widely believed, and it is quite reasonable that future cash flows or market movements are not purely justified by hard facts. In other words, investors are subjective rather than objective. Empirically, the movements of stock prices are not always consistent with the quantitative measures of firms' fundamentals (e.g., [Shi81], [Rol88], [CPS89] and [TSTM07]), which make people re-think the fluctuation of stock prices and seek other evidence to explain it. Eventually, people find that quantified linguistic information may have extra power to explain financial variables and predict futures.

The efficient market hypothesis asserts that financial markets are "informationally efficient", which means the current prices on all tradable finance products or assets have already reflected all known information and all occurred facts. Moreover, any prices in finance market are unbiased and contain all the wisdom or future forecasts from investors. Therefore, investors cannot make profits from the market if their trading strategies are based on known information because market prices are efficiently collecting and aggregating various information and keep changing without delay.

However, as we said, information is unbiased but investors are biased. We cannot assume all information has the same exposure to all investors, let alone informational asymmetry and "buyer-seller" asymmetry. All these reasons make the real finance market not really theoretically efficient. Some encouraging results also prove the conditional usage of the efficient market hypothesis. Particularly, [Cha03] shows that stock prices appear to drift after important corporate events for up to several months. This suggests that some of the drift is caused by the price's underreaction to information.

Another observation is that Tetlock07 [TSTM07] investigates whether the occurence of negative words in firm-specific news articles can help us predict firms' cash flows and whether firms' stock market prices incorporate linguistic information efficiently. The result of his experiments proves firms' stock prices underreact to the underlying negative information of news articles. More specifically, negative information in news articles are reflected in stock market prices with a roughly one-day delay. This is a good sign for investors. Based on this, people should be able to design trading strategies and potentially make profits as long as they can analyze the daily or intraday linguistic

information properly.

Generally speaking, linguistic sources play an important and effective supplementary role in financial forecasting, which cannot be replaced by any other financial indicators. Moreover, a financial news analysis system will also help investors to understand voluminous linguistic information and design trading strategies in an efficient, prompt, and intelligent way. In addition, the research in this area requires researchers to have extensive knowledge in finance, mathematics, statistics and computer science. All those factors illustrate that financial analysis with news is a promising and rather interesting research topic.

1.3 What are the main ideas?

The basic idea of the proposed research is to quantify linguistic information with text mining techniques, get the predefined set of features of the training data, and then build prediction models with classical statistical approaches or statistical learning algorithms. After this, we will be able to use target linguistic data as input, get the corresponding features, and apply them in the prediction model to obtain the forecasting result for the desired financial variables.

In terms of text mining, the 3-category model [MK06b] is widely used to label documents or words. The first category consists of some news articles or words that make the associated financial variables increase to a certain degree in a certain time period, for example, a news event makes the price of the single stock "IBM" increase 0.5% in the following day. By contrast, the second category consists of some news articles or words that make the associated financial variables decrease to a certain degree in a certain time period. The third category includes the other news articles or words. We say that the first category has a positive sentiment while the second category has a negative sentiment. The third category is termed of neutral sentiment.

The most common implementation of the 3-category model is the "Bag-of-Words" scheme [TSTM07], which generates a document-term matrix based on the input news articles. That is, the rows represent all the meaningful words appeared in this article, and the columns are words that show some sentiment tendency, either positive or negative, like "good", "happy", "lose", or "bankrupt". The columns words are some standard word list applicable for any news article. One way to get this list is to use a professional psychosocial dictionary. After the document-term matrix is generated, the challenge in post processing is to translate the matrix into some meaningful, quantitative, and conceptual variables which convey the positiveness or negativeness of the input text data and are expected to be used to correlate with the targeted finance variables.

In terms of modeling techniques, the most efficient and commonly used approaches include multiple linear regression, Approximate Nearest Neighbor, naïve Bayes classification, Neural Networks, Support Vector Machines, and so on. I will discuss these in more detail in the following sections.

2 Previous Work

Financial analysis research with linguistic information has been gaining popularity since early nineties last century. Different people work on this topic in different ways. In this section, firstly we will classify previous research work into different subcategories across 8 different dimensions. Then we will review in a more detail the most relevant work that has been done so far in the areas of finance, computer science, and industry respectively.

2.1 Taxonomies of Financial Analysis with News

The previous work are classified in 8 different dimensions according to their specific attributes like targets, market examined, input sources, forecasting horizon, and modeling methodologies, etc. They are explained in the following sections.

2.1.1 Target Disciplines

Researchers work on this topic because it can target their specifically interested disciplines. Generally speaking, the target disciplines include:

- Econometrics, Business, or Finance: Most research in this category is done by people from econometrics or quantitative finance because they want to use the linguistic power to study financial phenomena or predict the trend of finance variables. Some people even from behavioral finance area and they hope to study the interaction between markets and people's words.
- Applied Mathematics and Statistics: Financial analysis with news is also closely coupled with applied mathematics and statistics. However, only very few researchers are from this area. The basic reason is that researchers from applied mathematics don't have much background in natural language processing.
- Computer Sciences: Many researchers target the computer science discipline. Computer science people have comprehensive understanding regarding NLP, and they can apply computer science approaches to solve modeling problems, such as artificial intelligence or machine learning.
- Commerce and Industry: There are also many relevant industry projects in this area because they can guide money investment in the financial market.

2.1.2 Aims to Forecast

People analyze news in order to do prediction. Usually the following finance variables are expected to be forecasted:

- Market Trends: General trends of financial markets, like Dow Jones Industrial Average or S&P 500 Index.
- Stock Prices or Returns: This category contributes to the biggest portion of the previous work.

- Earnings: Firms' quarterly earnings or any other book values.
- Volatilities: Volatility is considered an accurate measure of risk. The prediction of volatilities can thus help people to invest. In some cases, predicting volatility is more meaningful than predicting prices or returns, for example, in option trading.
- Trading Volumes: Volumes of trade for any financial product.
- Currency Exchange Rate: The currency market is more stable and less risky than the stock market, but is still an important market.
- Futures, Options or other Derivatives: Each derivative may have different behavior, but all of them can be predicted to some extent using news.

2.1.3 Financial Market Examined

The financial markets people have examined include NYSE, NASDAQ, S&P 500, and markets outside US like those in Europe, Japan, or Hong Kong. Despite slight differences, they are all good research targets as long as they are influenced by the efficient market theory.

2.1.4 Text Sources

The quality of input text sources impacts the result of forecasting greatly. Some researchers use general linguistic information such as news, while others use some special information such as analysts' reports. We summarize them as follows.

- News: Public news like the New York Times, the Wall Street Journal, and private news like the Dow Jones News Services.
- Blogs: Regarded to have more targeted text information, and thus sometimes they are more meaningful than news.
- Annual Reports: Distributed by firms, similar sources also include firms' earning press.
- Analysts' Reports: Analysis reports from experts, more consistent and intensive than general news.
- Chat Rooms or Small Talks: Internet chat rooms or small talks usually can record different opinions from different people, which is called collected wisdom. They are quite useful as predictors in some cases.

2.1.5 Price Frequency of Input Data

The price data we examined varies from very high frequency to very low frequency. The high frequency data are intraday prices, which may gathered every minute or several minutes. The low frequency data are daily, monthly or yearly prices. The different price frequency requires the corresponding linguistic information to be collected and processed at different speeds.

2.1.6 Forecasting Horizon

According to the different price frequency of input data, previous work focuses on several different time scales for the prediction, particularly, the forecast of intraday, next-day, or monthly prices.

2.1.7 Natural Language Processing Analysis Level

At the natural language processing analysis level, people need to process the text information and output some specific signs or indicators in order to find out how the text information correlates with prices and how to use them for forecasting. These indicators include:

- Specific Events: Events identified by expects, which may impact the prices of particular financial products. These events are like *merging*, *acquisition*, *lawsuit*, *layoff*, *elect executive*, *debt issue*, etc.
- Reference Counts: The number of occurences of specific entites.
- Entity Sentiment: The positiveness or negativeness collected from the linguistic information.

2.1.8 Statistical Analysis Techniques

Previous work involves many techniques in the wide area of statistics or machine learning. Some important methods are listed below:

- Linear Regression: The most basic technique to describe the relationship between response variables and explanatory variables.
- Approximate Nearest Neighbor: One of the simplest learning algorithms. Objects are classified based on the closest training examples in the feature space.
- Bayes Learning: A learning algorithm based on Bayes' theorem regarding the conditional and marginal probabilities. Bayes learning is one of the most successful algorithms for classifying text documents and has shown great success in many previous applications. Bayes learning method is simple but still robust.
- Artificial Neural Networks: ANNs have also been extensively used as nonlinear mapping for statistical modeling. They have been applied in many areas, especially in applications for classification, clustering or prediction.
- Support Vector Machines: Support Vector Machines (SVMs) are a set of related supervised learning methods used for classification and regression. Usually SVMs perform better than multiple linear regression models because sometimes the real world can not be described with a linear model.
- Time Series Analysis: Clearly, both linguistic data and price data are time series. Some analysis is needed to understand the underlying rules of these data points, then we will be able to set up models to forecast future data points. There have been plenty of time series models built over the years, but the most successful models are ARMA, ARCH, and GARCH.

Papers	Aims to Forecast	Financial Market Examined	Period	Text Sources	Price Frequency	Forecasting Horizon	NLP Analysis Level	Statistical Analysis Techniques	Other Principle Models Used
[Cha03]	Stock Price Trend, Abnormal Return	CRSP stocks	1980s-1990s	News	Monthly	1 Month	Events and Articles	N/A	CAPM , 3-Factor, Size and Book-to-Market Models
[TSTM07]	Firm Earnings, Stock Returns	S&P 500	1980-2004	WSJ & DJNS	Daily	1 Day	Words	Regression	N/A
[BW07]	Stock Returns	N/A	1966-2005	Misc Sources	Monthly	1 Month	Articles	Regression	N/A
[AF06]	Stock Market Overraction to news	NYSE, AMEX, Nasdaq	1973-2001	WSJ	Daily	N/A	Events	Naive Bayes	N/A
[AF04]	Stock Returns, Trading Volume, Volatility	NYSE, Nasdaq	2000	Internet Message Boards	15 mins, 1 hour and 1 day	15 mins, 1 hour and 1 day	Messages	Naive Bayes, Regression	GARCH

Table 1: Works with a Target Discipline in Finance.

2.2 Works with a Target Discipline in Finance

This category contribute one of the largest sector of the previous work. Researchers in this category usually are from Department of Finance, Department of Economics, or School of Business.

Table 1 shows the comparison table of previous works. Some more works ([BSV98], [BSV98], [HLS00], etc) will not be described in detail here.

2.2.1 Chan [Cha03]

The author examines monthly returns to a subset of stocks after public news about them is released as well as compares them to stocks with similar returns, but no identifiable public news. There is a difference between the two sets. The author finds that investors react slowly to information, especially after bad news. Another important finding is that stocks tend to reverse in the subsequent month after extreme price movements unaccompanied by public news. These patterns are statistically significant, even after excluding earnings announcements, controlling for size, book-to-market, risk exposure and other effects.

To verify the results, the author uses a database of stories about companies from major news sources, and looks at monthly stock returns after two sources of stimuli. The first is public news, which is identifiable from headlines and extreme concurrent monthly returns. The second is large price movements unaccompanied by notable news. For each month, portfolios of stocks by each source are formed, and momentum trading strategies are examined. The results are proved by analyzing the stocks with news or no-news using several different kinds of the cumulative abnormal returns (CARs), such as CAPM Model, 3-Factor Model, Size and Book-to-Market Model.

One drawback of this paper is the date granularity. It examines the monthly return of each interesting stock and the relevant news within this month. However, in some case the monthly returns are neutralized if there were both good news and bad news

in this month and we end up catching no signal in terms of prices. This makes the conclusion less convincing.

2.2.2 Tetlock et al. [TSTM07]

This paper examines whether quantitative measures of linguistic information can be used to predict the earnings and stock returns of individual firms. There are three major findings in this paper. Firstly, negative words in firm-specific news stories will decrease firm earnings, which means the words contained in news stories are not redundant information, but instead, they expose some hard-to-quantify aspects of firms' fundamentals. Secondly, stock prices basically underreact to negative sentiment of linguistic information, say, the stock market prices respond to the information embedded in negative words with a small delay, roughly one day. Finally, the earnings and return predictability from negative words largely comes from the information regarding firms' fundamentals. That is, the negative words about firms' fundamentals are particularly useful predictors for both earnings and returns.

This paper is not saying quantitative measures of linguistic information are more useful than the traditional accounting measures of firms' fundamentals. Instead, the author investigates whether the negative words in firm-specific news stories can give us better understanding of firms' cash flows and whether the corresponding stock market prices of firms incorporate linguistic information efficiently. This paper also describes that positive words actually are much weaker predicting signs than negative words, especially after controlling for negative words. The reason is that negative words have a much stronger correlation with stock returns than other any words, including positive words.

This paper verifies its claims by using negative words to predict earnings and stock returns. His analysis focuses on the fraction of negative words in DJNS and WSJ stories about S&P 500 firms from 1980 through 2004 inclusive. He uses the positive and negative word categories in the Harvard-IV-4 psychosocial dictionary and makes the simplifying assumption that all negative words in this dictionary are equally informative, and other words are uninformative. The main control variable is standardize fraction of negative words. Other control variables include firms' lagged earnings, size, book-to-market ratio, trading volume, three measures of recent stock returns, analysts' earnings forecast revisions, and analysts' forecast dispersion.

2.2.3 Baker et al. [BW07]

This paper applies the "top down" approach to behavioral finance, which focuses on the measurement of reduced form, aggregate sentiment and traces its impacts to stock returns. The author explains what kind of stocks are likely to be affected by investor sentiment based on two assumptions in behavioral finance: sentiment and limits to arbitrage. This paper concludes stocks of low capitalization, younger, unprofitable, high volatility, non-dividend paying, growth companies, or stocks of firms in financial distress, are likely to be disproportionately sensitive to investor sentiment. The authors also review the theoretical and empirical evidence for these predictions.

Theoretically, the authors measure investor sentiment via a sentiment index, which combines multiple factors, including: *Investor Surveys*, *Investor Mode*, *Retail Investor Trades*, *Mutual Fund Flows*, *Trading volume*, *Dividend Premium*, *Closed-end Fund Discount*, *IPO First-Day Returns*, *IPO Volume*, *Equity Issumes Over Total New Issues*, *Insider Trading*, etc. Empirically, the author investigates the stock market from January 1966 to December 2005. Based on six measures of sentiment: the closed-end fund discount (CEFD), detrended log turnover (TURN), the number of IPOs (NIPO), the first-day return on IPOs (RIPO), the dividend premium (PDND), and the equity share in new issues (S), the below experiential formula is given:

$SENT = -0.23CEFD + 0.23TURN + 0.24NIPO + 0.29RIPO - 0.32PDND + 0.23S$

One noticeable finding is that when sentiment is low, the average future returns of speculative stocks exceed those of bond-like stocks. On the contrary, when sentiment is high, the average future returns of speculative stocks are lower than those of bond-like stocks. The result indicates that the fact that riskier stocks sometimes have lower expected returns is inconsistent with the classical asset pricing method in which investors bear risk because they expect higher returns.

This paper gives a more comprehensive way to evaluate investors' sentiment. However, it is more theoretical rather than practical in the real world because some factors in this sentiment index model are hard to collect.

2.2.4 Antweiler et al. [AF06]

The purpose of the author is to examine the correctness of the "effective market hypothesis" on public news, which traditionally believes that the stock market could digest public information and reflect the fluctuation of stock price at a very short time, say, in one or two days at most. Moreover, it is well known that the news may leak before it goes to public, which leads the result that theoretically we could catch no subsequent cumulative abnormal returns of stocks after the news is released. However, the author's experiments challenge the effectiveness of the stock market.

In order to study this problem, the authors collect Wall Street Journal corporate news stories from 1973 to 2001, classifies them and identifies all the significant topics (events) with computational linguistics methods based on Naïve Bayes classifier. After tracking the day-by-day stock returns around these topics' release time for up to more than one month in before-release and after-release directions, the result shows there is a reversal after the release of the specific news event, which means the stock market usually overreacts to news stories. In other words, on average the pre-event and post-event abnormal returns have the opposite sign. The stock may have a prompt response to the news, and then follow by a gradual and lengthy reversal, which could be observed for from 1 or 2 weeks up to several months. More importantly, the reversal has a much longer period during recessions than expansions. Clearly, the event window is much larger than that we have thought before.

Papers	Aims to Forecast	Financial Market Examined	Period	Text Sources	Price Frequency	Forecasting Horizon	NLP Analysis Level	Statistical Analysis Techniques	Other Principle Models Used
[WCe98] [Cho99] [CWZ99]	Equity Index Trend	DJIA, Nikkei, FTSE, HS, ST	1997-1998	News	Daily	24 hours	Words	Naive Bayes	k-NN, Neural Net
[LSL+00a] [LSL+00b]	Stock Price Trend	USA stocks	1999-2000	News	10 Min.	1 hour	Terms	Naive Bayes	N/A
[SGS05] [SGS04] [Tho03]	Volatilities	Russell 3000	2001-2002	News	Daily	N/A	Terms	Decision Rules	N/A
[Gid01] [GE03]	Stock Price Trend	DJIA	2001-2002	News	10 Min.	1 hour	Words	Naïve Bayes, Regression	N/A
[PW02]	Exchange Rate	USD/DEM and USD/JPY	1993	News headlines	1 hour	3 hours	Words	Decision Rules	N/A
[FYL02] [FYL03]	Stock Price Trend	Hong Kong Exchange Market	2002-2003	news	Intraday	1 hour	Words	Linear SVM	N/A
[Mit04] [MK06a]	Stock Price Trend	S&P 500	2002	news	15 sec.	15 minutes	Terms	Non-linear SVM	k-NN
[YJDS06]	Stock Price Trend	N/A	1998-2005	Company Announce	Daily	N/A	Terms	SVM	N/A

Table 2: Works with a Target Discipline in Computer Science.

2.2.5 Antweiler et al. [AF04]

This paper intensively studies whether the content in the Internet Stock Message Boards can predict the market. The three topics in this paper are stock returns, trading volume and volatilities. To achieve this goal, the authors downloaded more than 1.5 million messages from Yahoo! Finance and Raging Bull, which are the two most popular Internet Stock Message Boards, and used Naïve Bayes and Support Vector Machine as classifiers to assess the content of these stock messages. The result shows these message boards are quite informative. In terms of returns, the paper defines bullishness and shows that greater bullishness is positively and significantly associated with returns. The authors also find that higher message postings predict negative subsequent returns, which has not previously been reported. In terms of trading volume, the paper shows controversial opinions are associated with more trades. In terms of volatility, the paper also shows the message postings help to predict volatility for both daily and intraday tradings.

2.3 Works with a Target Discipline in Computer Science

In this section, we will review the previous works with a target discipline in Computer Science.

Table 2 shows some of the related works in this sector.

2.3.1 Wuthrich et al. [WCe98]

This paper attempts to forecast daily stock price trends by predicting the five major equity indices in Asian stock markets, i.e. DJIA, Nikkei, FTSE, HS, ST. Other papers [Cho99] and [CWZ99] are also related to this work. The authors use a Naïve Bayes, a Nearest Neighbor and a Neural Net classifier respectively, to classify the input documents into 3 categories, in which it will make a particular index such as the Dow increase (at least 0.5%), or decrease (at least 0.5%) or remain steady(change less than 0.5% from the previous day). The author uses "accuracy" to evaluation the performance of this model, which means what percentage of the prediction are correct. According to the author's experiments, all the accuracy of the five indices are more than 40%. By contrast, the accuracy can only be 33.3% based on traders' random guesses because this is a 3-category model. However, the author only considers the daily closing price and assumes that the next day's open price is the same as the previous day's close price. That is not true in most cases, which makes the conclusion of this paper have less credibility.

2.3.2 Lavrenko et al. [LSL$^+$00a] and [LSL$^+$00b]

The author designs and implements a system named Ænalyst, which aims to predict single stock price trends by identified news stories that are highly correlated with the market trend. Firstly, the Ænalyst helps to select interesting news articles from Biz Yahoo!. The author also identifies financial time series trends by using piecewise linear regression, assigns different labels for different trends, and thus a 5-category model is created. Particularly, the segments with trend slope greater than or equal to 0.75 are labeled *SURGE*; the segments with trend slope between 0.5 and 0.75 are labeled *SLIGHT+*; the labels *No Recommendation*, *SLIGHT-* and *PLUNGE* are also defined accordingly. Next, the selected input news articles are trained with Naïve Bayes classifier. Finally, the future stock price trend could be predicted with the selected news and trained model, and therefore a long or short trading decision could be made. With a 40-days market simulation during 2000, the model achieves a profit gain of 23 bps per transaction after performing 12,000 transactions. However, this model is trying to design intra-day trading strategies and indeed the number of trades are significant on each day, so transaction fees should be considered in this model. Otherwise, the model is not practical in the real world.

2.3.3 Thomas et al. [SGS05], [SGS04] and [Tho03]

These papers described a rule-based trading strategy developed at the Robotics Institute of Carnegie Mellon University. Differing from the previous widely used ["Increase", "Decrease" and "Keep steady"] 3-category model, it uses a rule-based classifier with 39 categories, e.g., acquisition, earnings outlook, IPO/spinoff, or lawsuit. For each category, there are some regular expressions to define it. Moreover, some trading rules are associated with each category. This strategy pays attention to volatilities more than stock price trends. More specifically, if some particular events are identified, which may increase the market volatility, the trading rules require the investors to exit the

market temporarily. The time to re-enter the market also depends on the followed technical indicators. One weakness of this model is that it cannot react to some sudden events promptly because it only deals with daily data. Moreover, some important performance data are missed in these papers, and thus it is hard to give a reasonable evaluation for it.

2.3.4 Gidófalvi [Gid01] and [GE03]

In terms of approach, the work done by Gidófalvi is identical to most of the other research in this section. More specifically, the author also classified stock price movements as "increase", "decrease" and "unchanged", and labeled the corresponding news published around then as "up", "down" or "unchanged". Bayes learning is also used here. However, the author focuses on short period price trends and intraday trading strategies. Moreover, with careful experiments, the author finds significant predictive power for stock price trends during 20 minutes before and 20 minutes after the publication of news. News starts to influence the market before it is published because some market players have ways to access this information even before it becomes publicly available.

2.3.5 Peramunetilleke et al. [PW02]

This paper differs from the above papers in that it aims to forecast not stock prices, but the currency exchange rates from news headlines. Because the headlines contain not only the effect, e.g., "the dollar rises against the Deutschmark", but also the reasons behind it, e.g., "because of a weak German bond market", it is believed that the headline-based model should provide a good indication for currency exchange rate. The authors use a classical 3-category classification("dollar up", "dollar down" and "dollar steady") based on rules, which are generated from a handcrafted dictionary provided by currency experts. The author simulates this model with data from 1993. The result shows this model has a prediction accuracy of around 50%, while a random trader's accuracy can only reach 33.3%.

2.3.6 Fung et al. [FYL02]

The approach described in this paper is similar to that in paper 2.3.2. It uses a statistics based piecewise segmentation model to identify segmental trends in financial time series, i.e., "Rise" and "Drop" two categories according to their slope and coefficient of determination. However, instead of building up everything from scratch, the authors use IBM Intelligent Miner^{TM} for feature extraction and the SVM^{light} package for training and prediction of classifiers. The authors compare the performance of this model versus that of a "fixed period approach", which assumes that every piece of information only has an impact on the market within a certain fixed time duration. Generally speaking, the authors' model performs better because it examines all the news articles instead of partial ones. However, the "fixed period approach" works better while a stock has too many news articles because there maybe some noise among them if we examine all the articles.

2.3.7 Mittermayer et al. [Mit04] and [MK06a]

The model proposed by Mittermayer follows the traditional 3-category model with categories "Buy", "Short" and "No Recommendation". However, there are some differences from previous models. Firstly, Mittermayer's work handles very high-frequency data, say, every 15 seconds. Secondly, only company-specific news articles that have a high distribution probability are considered and all articles from editorial newswires are simply neglected because basically they are regarded as redundant information. Thirdly, several different classification algorithms are applied in this model: k-NN and several non-linear SVM algorithms(Gauss, sigmoid and polynomial kernel respectively), while most previous models only apply Naïve Bayes or linear SVM algorithms. The simulation result is impressive, which produces a profit as high as 23 bps per roundtrip(a buy plus a short action). Particularly, the best result comes from the SVM algorithm based on polynomial kernel.

2.3.8 Yu et al. [YJDS06]

The purpose of this paper is to classify the impact of company specific news to stock prices, say, "Downward impact", "Upward impact", or "Neutral impact" to the stock prices. SVM is also applied in the classification algorithm. The most important difference from previous classification methods is that the authors incorporate "financial domain knowledge" into the SVM algorithm. "Financial domain knowledge" contains two elements - extreme volatilities and changing points, which are computed from stock price and net return time series respectively. Especially, the changing points in the stock price time series are detected by piecewise linear fitting. Simulating a single stock from 06/15/1998 to 03/16/2005, the authors claim that the accuracy of the 3-category classification is 65.73%, which is much higher than some previous published algorithms.

However, generally speaking, it is hard to tell the overall performance of an algorithm with just one single simulation. Clearly, the authors need to provide more evidence. Moreover, the ultimate goal of news classification is to forecast future price trend or volatilities. All the important forecast data are missed in this paper, which makes the paper less persuasive.

2.4 Commercial and Industrial Projects

There are more and more financial analysis projects that use news data deployed in industry in recent years. Two kinds of companies are interested in doing this. The first category is some big financial information firms like Dow Jones or Thomson. The second category is some small firms, like PredictWallStreet or Ivontu, who try to sell their financial prediction methods to investors. Table 3 shows the previous works in this sector.

2.4.1 Thomson One (*http://www.thomsononeim.com*)

Thomson Financial claims to be the No. 1 provider in the wide area of marketing research and analysis. Thomson One [Tho] is Thomson Financial's web based applica-

Projects	Aims to Forecast	Financial Market Examined	Period	Text Sources	Price Frequency	Forecasting Horizon	NLP Analysis Level	Statistical Analysis Techniques	Other Principle Models Used
ThomsonOne	Stock Prices, Trading Volume, Earnings	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Dow Jones News Analytics	Stock Price Trend	N/A	N/A	News	N/A	N/A	Sentiment	Bayes et al.	N/A
PredictWallStreet	Stock Price Trend	AMEX,OTC, NASDAQ, NYSE	N/A	Collective Sentiment	N/A	1 day	Pos/Neg Sentiment	N/A	N/A
Ivontu	Stock Price Trend	S&P 500	N/A	Any Texts	N/A	1 day	Pos/Neg Sentiment	Machine Learning	N/A

Table 3: Commercial and Industrial Projects

tion that provides research information and financial forecasts for various companies and markets, regarding their stock prices, volumes traded as well as earnings estimates. ThomsonOne's biggest advantage is that it can centralize tens of thousands of financial news, and research documents as well as analysts' reports. It is unsurprising that ThomsonOne's forecasts are based on these information it collects. However, some important questions are left unclear, e.g., how largely the forecasts depend on the automatic linguistic analysis, and what are the techniques behind the forecasting system? Thomson One is like a black box for us, but we just put it here to indicate that it is a successful commercial product.

2.4.2 Dow Jones News Analytics (*http://www.djnewsanalytics.com*)

Dow Jones News Analytics [Dow] aims to provide sentiment analytics for algorithmic and quantitative trading based on real time news coverage including all editions of The Wall Street Journal and Barron's. This system is powered by RavenPack, which is a commercial product that conducts news sentiment analysis with comprehensive computational linguistic methodologies including Bayes training, vector classification, word/phrase lists, pattern detection and market response-based analysis. The biggest advantage is that Dow Jones News Analytics offers a Developer's kit with a Java API that allows the user to access news and sentiment data easily. Moreover, it provides embedded statistical and visualization tools.

2.4.3 PredictWallStreet (*http://www.predictwallstreet.com*)

PredictWallStreet.com [Pre] provides continually updated forecasts of stock prices based on the collective wisdom of its users. The idea is simple. User tells the website whether he thinks a stock price will go up or down. Based on proprietary algorithms, the website combines the user's prediction with the predictions of everyone else who has predicted for that stock and then it show you what others think. Since the stock market moves based on the collective thinking of thousands of individuals, the more people who predict on PredictWallStreet.com, the more potentially useful the information on the site becomes.

The idea sounds reasonable, but the weaknesses are obvious. Firstly, it only tells you what percentage of evaluators think the stock price will go up and what percentage of them think it will go down, but it fails to tell you what the price will be and how accurate it is. Secondly, it only forecasts the trends of closing prices of the next day, which are not adequate enough to make a successful trading strategy. Finally, a group of malicious evaluators can easily mislead investors by voting the wrong opinions, especially if the website has a small population of users.

2.4.4 Ivontu (*http://www.ivontu.com*)

The website [Ivo] allow the user to create his own model, which will use some machine learning techniques to predict the trend of specific S&P 500 stocks with their fed news. The system uses two measures, robustness and accuracy, to measure how good or bad your model is. This system has some advantages in that the user can track one stock with different models, and it allows users possibly to share their models. However, there are more disadvantages. 1) The user must create separate models for each stock. 2) It is strongly suggested that the user only use one single source(for example, news from a specific analyst) for one model. 3) The user needs to manually copy and paste news to the text-box then click the button to analyze it. 4) The learning/prediction algorithms have strong biases because they largely depended on the news selected by users. In fact, this website is in its very early stages of development.

3 Text Analysis with Lydia

The *Lydia* system ([LKS05], *http://www.textmap.com*) is a project that developed in the Algorithms Lab at Stony Brook University. The goal of *Lydia* is to do thorough high-speed analysis of online daily newspapers. The input of *Lydia* includes the coverage of more than 800 nationwide newspapers like the *New York Times*, and the *Washington Post*, or local newspapers like *The New York Observer*, or *Newsday* in Long Island. We have a spider program to collect and download online news every day, before which most online newspapers have updated their information to that of the current day. Then the spider program translates the downloaded news to files in XML format and feeds them to *Lydia*. After that, *Lydia* uses its NLP techniques to do the following: Named entity recognition, Juxtaposition analysis, Synonym set identification, Temporal and spatial analysis.

Lydia generates an entity database, with which we can build news reference and sentiment time series in a scale of one day. Specifically, the data include: all the entities appeared in the fed news, the synonymic sets of entities, entity juxtaposition analysis, and all the entities' corresponding daily reference counts, sentence counts, and article counts. Lydia is also capable of reporting, aggregating and scoring the positive or negative sentiment to each entity on a daily basis. Lydia reports the sentiment data with 7 identified categories, including: *General*, *Business*, *Crime*, *Health*, *Politics*, *Sports* and *Media*. Figure 1 shows an example of sentiment time series using the data created by *Lydia*. *Lydia* is capable of processing any online or offline text information as well other than news. More information about *Lydia* can be found from papers

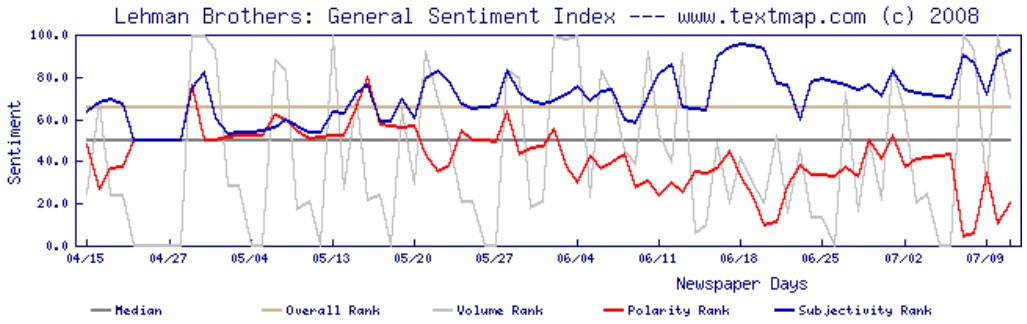

Figure 1: Sentiment Time series output by *Lydia* shows Lehman Brother's general sentiment index from mid April to mid July 2008. There is a slight increase of subjectivity rank and a significant decrease of polarity rank during this time period. The decrease of polarity means the decrease of positive sentiment, which is a negative sign for a firm.

Figure 2: The stock price movement of Lehman Brothers from middle April to middle July in 2008. There is a gradual yet significant decrease of stock price during this time period.

[GSS07], [MBL+06] and [LKS06].

Based on the reference counts and sentiment time series, we will be able to predict something we are interested in for the relevant entities. Figures 1 and 2 show the news sentiment time series and stock price time series of Lehman Brothers from mid April to mid July 2008. There is a significant sign that the movement of the news sentiment index is consistent with the that of stock prices. Therefore, *Lydia* is very suitable for our financial analysis with news. Currently the time series data are on a daily basis, but we may make it finer granularity later in order to do intraday forecasts.

For example, *Lydia* can be used to forecast movie grosses based on the news references and sentiment references of movie entities because we believe more successful movies will get more media exposure or positive occurence even before release. However, one difficulty for this problem is movie title matching. For example, one movie's name in IMDB is "Mr. And Mrs. Smith", but, in some cases it may be written as "Mr. & Mrs. Smith". This situation will give *Lydia* some difficulties to identify that

they are exactly the same movie. *Lydia* may even mistakenly believe the entity's name is "Mrs. Smith" instead of "Mr. & Mrs. Smith". Some movies, like "11th Hour",or "15 Minutes", they are very easily regarded as time periods instead of movies' names. More movies' names are just some daily common used words, e.g., "Pride", "Next", "Sunshine", "Interview", and "Fallen", which will make *Lydia*'s processing even harder. All these cases will cause lots of false positives or false negatives during entity identification phase, which will make that *Lydia* generate less accurate result and hence weaken its predictive power. One solution is to filter out these "bad" data before our analysis. Nevertheless, *Lydia* has proved its predictive power in some problems in the real world, such as social science analysis and movie gross analysis. Moreover, we are also continuously improving the quality of *Lydia*.

4 Movie Gross Forecasting

The movie industry is of intense interest to movie studios, economists and the public because of its high profits and entertainment nature. In 2007, the total revenue of U.S. movie market was $8.74 billion and the market continuous to grow. Now, an interesting question is, can we predict the market? In other words, can we forecast the movie gross even before the movie is released? Obviously, investors, movie studios, and movie distributors or retailers would be greatly interested in accurate predictions of gross because they want to make wise decisions to spend money. Furthermore, this topic is theoretically interesting, especially to researchers in economics, mathematics, computer science, or business schools because this is an interdisciplinary subject.

However, no simple or uniform solutions can work perfectly under all circumstances for movie gross prediction. Traditionally, people predict gross based on historical data analysis regarding some movie specific characteristics, e.g., the movie's genre, MPAA rating, budget, director, number of first-week theaters, etc., but with somewhat limited success. It is unsurprising that some small budget movies gain big success while some huge budget movies fail. Other issues that are not intrinsic to the quality of the movie, such as distribution and promotion, also factor in the gross. Nevertheless, some works are published, and they hope to give meaningful and correct movie gross forecasts in a large sense.

The aim of our work is also to predict movie grosses before the movies' release. The significant difference between our and previous work is that, other than traditional movie indicators, we also take movie relevant media publicity into consideration. It is quite straightforward that commercially successful movies, actors, directors, or distributors are always accompanied by more media exposure. If we analyze daily media coverage carefully, we will be able to identify which movies the public is interested in, and then we can tell which ones will be blockbusters and which ones will be losers.

In fact, to use linguistic information to help forecast something is not a new idea. People began to use news to forecast finance market like stock prices, volatilities, or earnings as early as 1990s. Several publications ([TSTM07], [AF06], [Cha03], [FYL03], et al.) have already shown the media's power on financial forecasting to some extent, and there is a growing body of work in this area. Considering the encouraging results on financial analysis using news, it is reasonable to infer that the news has predictive power for movie grosses as well. However, even though there is much work regarding financial analysis using news, no one has tried to apply linguistic analysis to movie gross prediction till now. This is the main reason why we make it our research topic here.

Our primary goal is to prove that we can give better prediction of movie grosses if we use news data. More specifically, we are trying to answer the following questions:

1. How is movie news data correlated with movie grosses?
2. Can we predict movie grosses merely with news data?
3. Does the combination of movie indicators and news data give a better prediction than either of them alone?

4. What kind of models can be used, and how can we set up those models for prediction?

The contents of this paper are organized as follows: Firstly, we will review related work. Secondly, we will describe our movie data sources, both traditional movie data and news data. Thirdly, we will give a correlation analysis and related modeling methodologies used by us, like regression and k-nearest neighbors. Finally, we will set up different models with traditional movie data, movie news data and their combination respectively as well as evaluate their performance.

4.1 Related Works

Different people work on this topic from different perspectives. Chen ([Che02]) builds a simple statistical model. The author identifies important movie variables like genre, release date, MPAA rating, actors and so on. He fits these variables in a simple linear regression model and then computes the predicted gross. The lower and upper bounds of the prediction are also given. For example, the actual gross of *Bridget Jone's Diary* (released in April 13, 2001) is \$71.50 million, but the author's predicted gross is \$28.22 million, and the predicted lower and upper bounds are \$6.62 million and \$120.40 million respectively. However, the author only provides the predictions for 4 movies and he fails to give a serious evaluation method for his model. Moreover, the lower and upper bounds of prediction indicates such a big range, which makes the prediction less meaningful.

Simonoff ([SS00]) follows a similar method to Chen, but he gives a more thorough analysis regarding the impact on movie grosses of important movie variables. Three models are built in the paper: Pre-release model gives the prediction of revenues prior to release, the first weekend model gives prediction of revenues after the first weekend of release, and the Oscar model gives prediction of revenues with consideration of the academy awards of movies. The simulation results show that the pre-release model works poorly, while the first weekend model and the Oscar model give much more accurate predictions. For example, *The Horse Whisperer* had an actual gross of \$74.37 million. Its predicted grosses for the pre-release, first weekend and Oscar models are \$1.405 million, \$63.932 million and \$59.391 million respectively. However, both the first weekend model and Oscar model are post-release models. Although the post-release models are also useful in some situations, pre-release models are of more practical use.

Differing from the above methods, Sawhney ([SE96]) predicts movie grosses with a two-step stochastic model under a queuing theory framework. The two steps are based on consumer's movie adoption processes - the *time to decide* to see the movie, and the *time to act* to act on the adoption decision. In fact, the objective of Sawhney's research is to forecast the later stage revenue in a movie's life cycle based on its early box office data. In practice, to predict later stage revenue is still meaningful because it could be an important investment guideline for practitioners in this industry, such as, movie exhibitors, home video makers, or even book or CD-ROM publishers. The author builds a parsimonious model based on the two-step process and estimates its parameters. Furthermore, the author predicts the movie's week-by-week gross and accumulated gross with more or less early stage box-office data. The author claims that

the model works pretty well by taking at most the first three weeks of data as input. However, the author also admits that it is much more difficult to give shape estimation for either model parameters or gross if we don't have any early stage movie gross data. One problem is that the author uses a small set of movies to verify his model. Most importantly, the work does not address the problem of pre-release prediction.

Another notable work in this area is done by Sharda ([SM00] and [SD06]). Sharda's work is also to predict a movie's performance before it is released. The difference is that the author transforms the movie gross prediction problem into a classification problem. That is, a movie's gross will be classified into one of nine categories, ranging from flop to blockbuster. The author then use two approaches, a neural network and rough sets, to classify grosses based on collected variables or computed variables of movies, like release date, rating, intensity of competition, star power, genre, technical effects, sequel, screens at opening, etc. The performance of these models is evaluated with accuracy, the percent correct classification rate. By running data for 1997, both the neural network model and rough sets model give the same accuracy. If a prediction within 1 or 2 categories are regarded as correct, both models' accuracy can reach 70%. However, the weakness of these models is also obvious: the models output a category rather than some specific number, which is not acceptable in some cases. Moreover, the author needs more experiments to verify his models.

Finally, the Hollywood Stock Exchange ([Hol] *http://www.hsx.com*) shows an interesting web-based virtual market, in which players can trade "shares" of actors, directors, upcoming films and film-related options. The traded prices of movie stocks could act as predictors of movie grosses because the ultimate value of movie stock indicates the movies' grosses. In short, HSX also achieves the goal of movie gross prediction via collected wisdom.

4.2 Movie Data

There are two kinds of movie data used in this paper, movie specific variables and movie news data. The movie specific variables are collected from traditional movie websites like IMDB, but the movie news data are obtained from *Lydia*, a powerful text processing engine in our Lab. With the combination of movie variables and news data, we will be able to build better movie prediction models.

4.2.1 Traditional Movie Variables

All the traditional movie variables are available at *http://www.imdb.com* and *http://www.the-numbers.com*. We wrote a spider program and downloaded data for all movies released from 1960 to 2008, totalling 10580 movies and 57736 actors.

We are interested in important movie attributes provided by above two websites and import them to our movie database. We call an attribute "important" because the attribute may impact the movie gross. The details of these attributes are:

1. *The Release Date*: The release date is important because a movie released during a holiday or summer season may attract more customers. The holidays considered by us include President's Day, Memorial Day, Independence Day, Labor Day,

Thanksgiving Day, and Christmas season. The summer season usually means the period from Memorial Day through Labor Day.

2. *Movie Name*: Usually different movies have different titles, but this is not always true. In addition to movie name, we also use a unique movie ID to identify a movie in our database.
3. *Distributor*: To choose the right distributor is critical to a movie's success. It's no doubt that historically successful distributors tend to continuously gain success in the future. Actually some distributors like *20th Century Fox* and *Universal* are top players in this market. However, we are not seriously considering the distributor as a factor in our models because lots of movies missed the distributor information in both "*IMDB*" and "*the-numbers*" databases.
4. *Budget*: Movie budgets vary from thousands of dollars to hundreds of millions of dollars. Higher investments expect higher grosses although it does not necessarily become true. But in most cases, budget is always one of the most important factors that influence the grosses of movies.
5. *Movie Gross in USA*: This is a movie's box-office revenue in the United States.
6. *Movie Worldwide Gross*: This is the movie's box-office revenue throughout the world.
7. *Director*: This is the director of a movie. A famous director usually has positive impact on the movie's box-office revenue.
8. *Actors*: For each movie, we take the top 15 actors in credits order. Top dollar actors definitely could increase the movie's gross greatly because of their "*star power*".
9. *Genres*: Genres define the category of a movie. For any specific movie, genres should be one or more categories from the following (19 genres totally according to IMDB): *Action, Adventure, Animation, Biography, Comedy, Crime, Documentary, Drama, Family, Fantasy, Horror, Music, Mystery, Romance, Sci-Fi, Short, Sport, Thriller, War*. For example, "*Forrest Gump*" has the genres of *Comedy*, *Drama* and *Romance*.
10. *MPAA Rating*: MPAA is the Motion Picture Association of America's film-rating system. It is used to rate a film's thematic content suitability for certain audiences. Different rating systems are used in different times. As early as 1960s, many movies are "*unrated*" or just rated as "*Approved*". Ratings "*G/M/R/X*" were used from 1968 to 1970, and ratings "*G/GP/R/X*" were used from 1970 to 1972. Nowadays, the MPAA ratings are:
    - G (General audiences): All ages admitted.
    - PG (Parental Guidance Suggested): Some material may be inappropriate for younger children.
    - PG-13 (Parents Strongly Cautioned): Some material may be inappropriate for children under 13 years old.
    - R (Restricted): No one under 17 admitted without parent or guardian.

- NC-17 (No children 17 or under admitted).

Most movies released after 1972 are rated by the "*G/PG/PG-13/R/NC-17*" system.

11. *IMDB Rating*: IMDB rating is a number with a total score of 10.0 that represents the overall evaluation from the audiences over years. For example, *Forrest Gump*'s score is 8.5 out of 10.0. However, the IMDB rating cannot be used as the movie gross prediction because normally it is evaluated after release.
12. *Origin Country*: The origin country indicates the country in which the movie is released initially. If the origin country of a movie is not the USA, we call it a foreign movie.
13. *Running Time*: The running time indicates the lasted time of a movie.
14. *Source*: The source of a movie could be "*Original Screenplay*", "*Based on TV*", "*Based on Book*", "*Sequel*", etc. However, currently the only character we are interested in is whether a movie is a sequel of a previous movie or not.
15. *Awards*: Awards mean a movie's academic award nominations or wins, e.g., Oscar, etc. It is obvious that awards are positively correlated with a movie's gross. But we don't take awards as the consideration in our models because the award information usually is not available before release.
16. *Movie Weekly Data*: Movie weekly data include the following:
    - Rank of this week
    - Gross of this week
    - Number of theaters of this week

    Actually all the weekly data are unknown before release except the first week's number of theaters.

We should notice that the Actors, MPAA Rating, IMDB Rating, Origin Country, and Awards data are downloaded from the *IMDB* website, while all others are fetched from the *the-numbers* website.

4.2.2 News Data from Lydia

The movie news data are output from Lydia, a high speed text processing system, by analyzing daily news from around 1000 nationwide or local online newspapers, from November 2004 to March 2008. During this time period, roughly 2600 movies were released.

Based on the input news, the output of Lydia is a list of entities, their corresponding number of references as well as number of sentiment references. As we have said before, for any specific entity, the Lydia data include:

1. Daily frequency counts
2. Daily article counts
3. Daily sentiment(both positive and negative) counts in 7 categories: *General*, *Business*, *Crime*, *Health*, *Politics*, *Sports* and *Media*.

The daily sentiment counts are in the basis of daily frequency counts.

4.3 Modeling Methodologies

Three basic modeling methodologies used in this paper are regression, piecewise linear regression, and k-nearest neighbor classifier.

4.3.1 Linear Regression

The first class of models used in this paper is the multilinear regression model. In this model, we use the training set to estimate the coefficients of all the predictors and then calculate the gross of the target movie based on the regression model. For example, $x_1, x_2, ..., x_n$ are movie predictors and they are correlated with gross g, so we can build a equation: $g = c_0 + c_1 * x_1 + c_2 * x_2 + ... + c_n * x_n$. The coefficients $c_0, c_1, c_2, ...c_n$ can be obtained from historical data. Therefore, the gross of a new movie can be computed with the linear equation. The regression model can be built with software like SAS, R, or S-Plus. Because our movie prediction software is written in Perl, here the Perl "Statistics-Regression" module is used.

The regression models are simple and easy to build, and we can give pretty good prediction if the predictors and responsors are highly correlated. However, the limitations of linear regression models are also obvious. They only ascertain the relationship between predictors and responsors, but fail to explain their underlying causal mechanism. For example, even if some factors are strong correlated with movie gross, we still cannot justify that the movie gross is caused by those specific indicators, which weakens our models. Moreover, some variables are not necessarily linearly correlated even though they show some relationship to a certain extent. However, we always assume the predictors and responsors are linearly correlated while we set up linear regression models.

4.3.2 Piecewise Linear Regression

It is hard to fit all the cases into a single regression model, which is why sometimes we need to set up piecewise linear regression models for different situations. For example, if we use a uniform regression model to forecast movie grosses, we always tend to overestimate low-grossing movies and underestimate high-grossing movies. However, the overall performance will be improved if we set up separate linear regression models for low-grossing movies and high-grossing movies respectively. The reason is the grosses of low-grossing movies and high-grossing movies react to predictors in different ways.

4.3.3 K-Nearest Neighbor Classifier

K-nearest neighbor is a simple machine learning algorithm, which builds a multidimensional feature space, calculates the distance between the target movie and all the movies in the training set, and then takes the k movies that have the k least distances from the target movie. In fact, all the movies released before the target movie are qualified to be in the training set. The estimated gross of the target movie will be the mean or median gross of the k-nearest neighbors. Usually k is a small integer like 1, 3, or 5, etc. The multidimensional feature space can be built using some movie parameters, e.g., budget, genres, number of theaters in the first week, or even movie

news data like movie reference counts or sentiment counts. The distance of two movies can be evaluated by Euclidean distance or Manhattan distance.

If n movie parameters are used in the k-NN model, the first movie has the parameters of $p_1, p_2, ..., p_n$ and the second movie has the parameter of $q_1, q_2, ..., q_n$, then the Euclidean distance of the two movies is calculated by: $dis = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$; while the Manhattan distance of the two distance is calculated by: $dis = \sum_{i=1}^{n}(|p_i - q_i|)$. In our experiments, the two methods lead to almost identical results because one distance measure is bigger which usually means the other measure is also bigger.

The correctness of the k-NN model is based on the assumption that "similar" movies should have similar grosses. Therefore, the goal of k-NN model is to identify the most "similar" movie of the target movie from the training set by examining their similarities.

4.4 Correlation Analysis

In this section, we will analyze the correlation between movie grosses and movie variables or news variables. The purpose of the correlation analysis is to guide us to set up the reasonable models for movie gross forecast. We will also be able to have an explicit understanding regarding how the movie variables and news variables influence the movie grosses.

4.4.1 Movie Variables and Movie Grosses

Basically speaking, there are two kinds of movie variables: numerical variables and categorical variables. Numerical movie variables are indicators like budget, opening screens, first-week grosses, and world grosses. Categorical movie variables are indicators like release date, MPAA rating, source, origin country, and genres. For all these movie variables, their correlation coefficient with grosses and some other statistical data are computed, including the number of movies in the corresponding category, the mean gross, median gross, maximum gross, and minimum gross. The detailed analysis is below.

1. Budget and Gross

   Usually the movies with higher investments are expected to yield higher grosses. The relationship between budget and gross can be shown by historical data. We collected all the movies which have the budget value in our database and were released between January 01, 1990 and December 31, 2007, totally 1500 movies. Figure 3 shows how the grosses fluctuate according to the movie budget. The movies are not shown chronologically, instead, they are sorted by their budget value. The trend is clear: more budget generates more gross.

   - *Time Value of Money:* The actual value of budget and gross may differ when we consider the time value of money. The correlation between raw grosses and raw budget is 0.629. But if we covert the value of budget and gross to the standard dollars (here we mean year 2007 dollars) based on the year-by-year interests rate before the correlation is evaluated, the correlation coefficient of them is 0.635. Table 4 shows this. Therefore, to use the time value of money

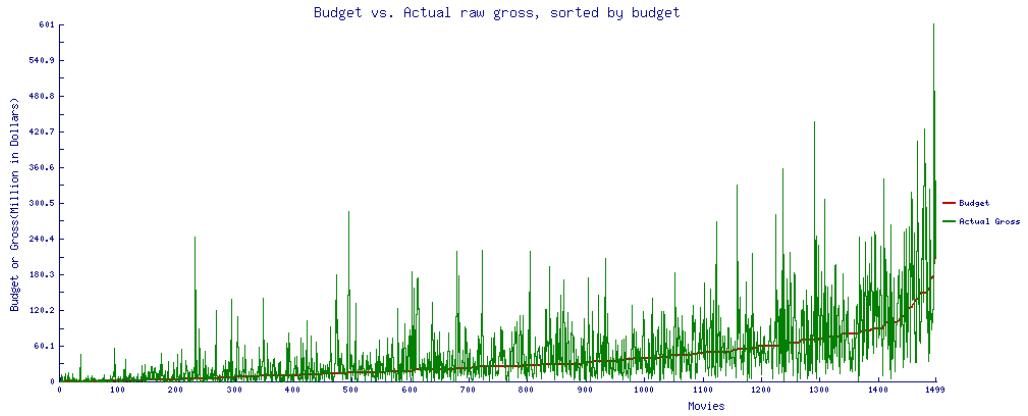

Figure 3: Movie budget vs. gross, sorted by budget (01/01/1990-12/31/2007). The rising line shows movies' budget, while the fluctuated line shows the corresponding grosses of movies.

Budget vs. Gross Correlation	All Movies	All Movies, Time Value	Low-grossing Movies	High-grossing Movies
Raw Value	0.629	0.635	0.405	0.537
Logged Value	0.672	0.680	0.453	0.483

Table 4: Budget versus Gross Correlation

improves the correlation of budget and gross. The time value of money is always used in this paper hereafter.

- *Raw value vs. Logged value:* The correlation coefficient of budget and gross is 0.629, and that of logged budget and logged gross is 0.672. Here we can see the logged values of budget and gross are more highly correlated than those of raw values. Table 4 shows the correlation of raw value and logged value of budget and grosses for all movies, low-grossing movies and high-grossing movies respectively. Figure 4 shows the scatter plot of the logged budget versus logged gross value. We can clearly see they are positive correlated.
- *Low-Grossing vs. High-Grossing Movies:* Traditionally people pay more attention to high-grossing movies, especially blockbusters. We classify the 1400 movies into 2 categories, 557 low-grossing movies with grosses smaller than $15 million, and 943 high-grossing movies with grosses equal to or bigger than $15 million. The correlation coefficients of budget versus gross on low-grossing and high-grossing movies are calculated in table 4. Even though both of them have the smaller coefficients than the total movie set, the grosses of high-gross movies are still more highly correlated with their budgets than those of low-grossing movies.

2. First Week Theaters and Gross

   The number of first week theaters is also called opening screens. From table 5 and figure 5 we can clearly see that the opening screens and grosses are positively related. The correlation coefficient of logged opening screens and logged grosses

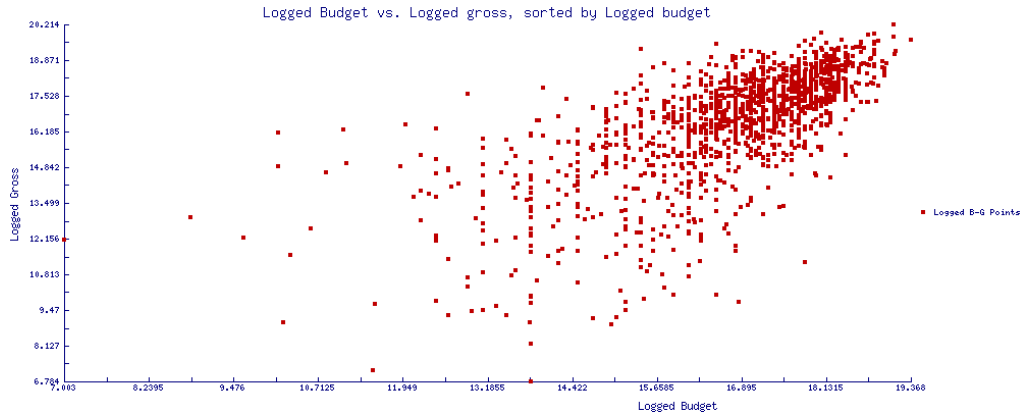

Figure 4: Scatter plot of Logged budget vs. Logged gross (01/01/1990-12/31/2007)

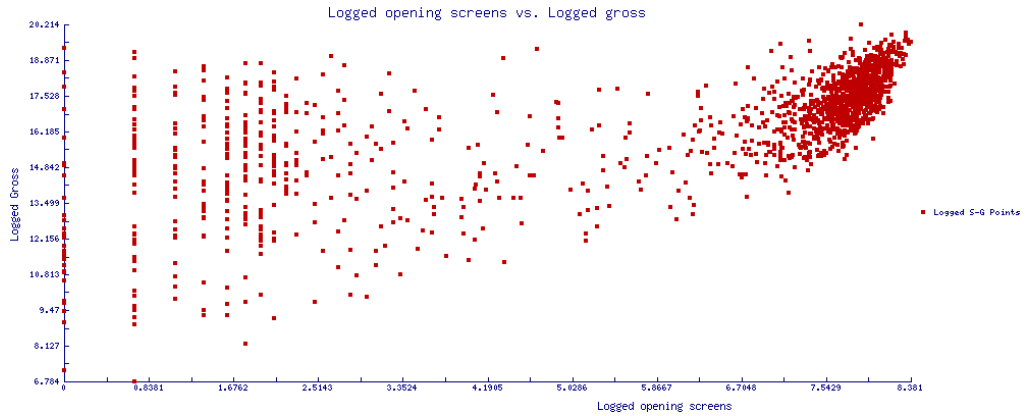

Figure 5: Scatter plot of Logged opening screens vs. Logged gross (01/01/1990-12/31/2007)

is as high as 0.647. If the movie's opening screens are less than 250, we treat it as a low opening-screen movie, otherwise it is a high opening-screen movie. We can find that the logged opening screens and logged grosses are not significantly correlated for low opening-screen movies, but they are strongly correlated for high opening-screen movies.

3. First Week Gross and Total Gross

   A movie's first week gross and its total gross are highly correlated, see figure 6. The correlation coefficient of their logged value is as high as 0.841, which means either a popular movie will be continuously maintaining its popularity afterwards, or most of the gross comes from the first week. Unfortunately, the first week gross is not a pre-release indicator, therefore it is not used in our model even though Simonoff ([SS00]) verified that the first-week gross is a very good indicator for movie total grosses.

4. World Gross and US Gross

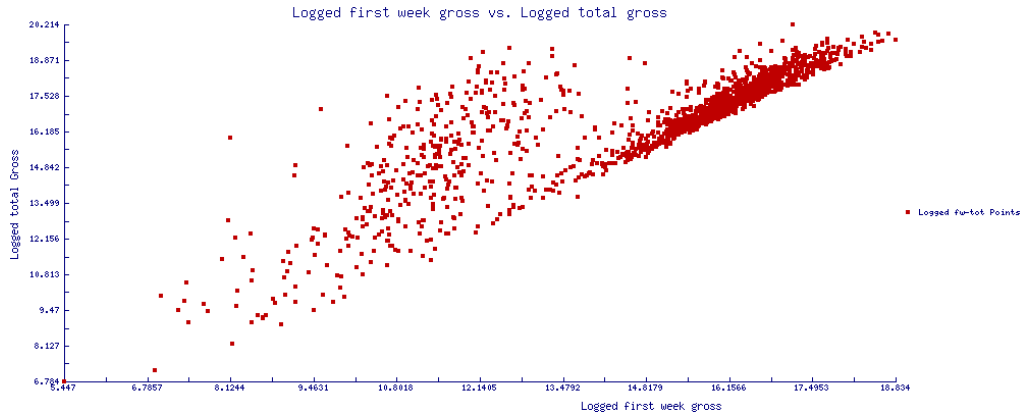

Figure 6: Scatter plot of Logged First Week Gross vs. Logged Total Gross (01/01/1990-12/31/2007)

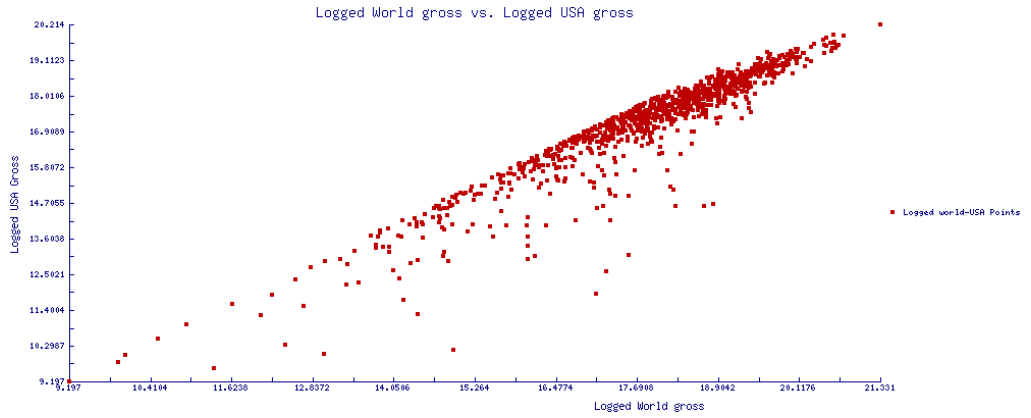

Figure 7: Scatter plot of Logged US Gross vs. Logged World Gross (01/01/1990-12/31/2007)

Similarly, the movie's US Gross and world gross are strongly correlated, with a correlation coefficient of 0.936. Scatter plot 7 illustrated their relationship.

5. Release Date and Gross

   All movies are classified into two groups based on whether their release dates are during holiday period or not. The "holiday" means all the important holidays period plus summer season period. The important holidays include President's Day, Memorial Day, Independence Day, Labor Day, Thanksgiving Day, and Christmas season. The summer season usually means the period from Memorial Day through Labor Day. Table 5 shows that holiday release dates are positively correlated with movies' gross while the non-holiday release dates are negatively correlated with grosses. Moreover, all the statistical data of movies released during holiday period are greater than those of movies released during non-holiday period.

6. MPAA Rating and Gross

   A movie's MPAA rating is G, PG, PG-13, R, or NC-17, from the least "mature"

to the most "mature". As a result, the statistical data show their mean gross, median gross, correlation of MPAA rating and gross are decreasing accordingly, see table 5. Therefore, the MPAA rating is an important indicator of movie grosses.

7. Source and Gross

   There are several different sources for a movie, but here we only consider if a movie is a sequel of a previous movie or not. Table 5 shows "sequel" is positive correlated with a movie's gross. The mean and median grosses of sequel movies are more than double those of non-sequel movies.

8. Origin Country and Gross

   We only differentiate a movie is a USA movie or foreign movie. Again, domestic movies show much better performance than foreign movies, which means "domestic" factor is positively correlated with movie grosses.

9. Genre and Gross

   The last important indicator is genre. Because most movies in IMDB have multiple genres, the first genre is called the primary genre, while the rest of them are called subsidiary genres. For example, movie *Forrest Gump* has three genres, *Comedy* is its primary genre, but *Drama* and *Romance* are its subsidiary genres.

   In Table 5, the numbers outside the parentheses of "*Movies*" column indicate the number of movies whose primary genres are in this category, while the numbers inside the parentheses indicate the number of movies whose either primary genres or subsidiary genres are in this category. The numbers inside the parentheses of "*Correlation*" column have the similar meaning.

   The statistical data in this table show that "*Action*", "*Adventure*", and "*Animation*" are positively correlated with grosses in both cases. By contrast, "*Biography*", "*Documentary*", and "*Drama*" are negatively correlated with grosses in both cases, and "*Crime*" and "*Horror*" have slightly negative correlation with grosses. "*Family*" is a special case: "*Family*" is a bad thing to be a primary genre of movies but it is a good thing to be a subsidiary genre of movies. Other genres have too few movies and their correlations with movie gross are less credible.

4.4.2 News Data and Movie Grosses

In order to study the relationship between movie news references and grosses, it is required that the movies at least have some media exposure. We collected the daily news from November 2004 to March 2008 as the input of *Lydia*. Because we need to examine the news references for a 4-month time periods before movies' release, we only consider the movies released between February 21th, 2005 and January 31th, 2008.

Our daily time series data for news entities are a group of 14 numbers, including:

- Entity article counts: The number of articles that refer the entity.
- Entity frequencies: The number of the occurences of the entity.

Movie Variables	Categories	Movies	Mean	Median	Min	Max	Correlation
Budget	All	1400	N/A	N/A	N/A	N/A	0.672
	Low-grossing	557	N/A	N/A	N/A	N/A	0.453
	High-grossing	943	N/A	N/A	N/A	N/A	0.483
Opening Screens	All	1400	N/A	N/A	N/A	N/A	0.647
	Low-screens	557	N/A	N/A	N/A	N/A	0.115
	High-screens	943	N/A	N/A	N/A	N/A	0.652
First Week Gross	All	1400	N/A	N/A	N/A	N/A	0.841
World Gross	All	1400	N/A	N/A	N/A	N/A	0.936
Release Date	Holiday	640	55.13	32.22	0.008	600.79	0.132
	Non-holiday	860	39.15	21.44	0.0008	436.72	-0.132
MPAA Rating	G	41	82.72	58.40	0.669	339.71	0.103
	PG	201	65.60	42.27	0.119	436.72	0.128
	PG-13	500	59.04	35.28	0.011	436.72	0.154
	R	646	30.68	16.98	0.0008	216.33	-0.221
	NC-17	17	18.18	7.4	0.030	70.10	-0.049
Source	Sequel	127	90.24	64.96	0.146	436.72	0.224
	Not Sequel	1373	41.88	22.73	0.0008	600.79	-0.224
Origin Country	USA	1191	50.28	30.31	0.0008	600.79	0.141
	Not USA	309	29.38	11.55	0.009	317.56	-0.141
Genres	Action	333 (369)	68.12	46.21	0.0008	423.32	0.197 (0.191)
	Adventure	83 (266)	69.35	38.40	0.407	317.56	0.094 (0.276)
	Animation	62 (72)	102.32	68.72	0.469	436.72	0.195 (0.081)
	Biography	51 (62)	26.79	16.01	0.240	125.55	-0.060 (-0.062)
	Comedy	461 (689)	38.65	18.62	0.008	329.69	-0.081 (0.054)
	Crime	89 (266)	35.70	23.09	0.017	165.09	-0.043 (-0.003)
	Documentary	22 (23)	14.33	3.55	0.0014	119.11	-0.064 (-0.067)
	Drama	284 (826)	28.02	14.91	0.008	600.79	-0.144 (-0.162)
	Family	15 (174)	39.99	19.42	3.59	144.16	-0.010 (0.236)
	Fantasy	9 (176)	47.04	42.29	19.29	101.07	0.001 (0.254)
	Horror	37 (139)	30.48	26.42	0.009	155.46	-0.041 (-0.057)
	Music	3 (83)	30.48	4.16	3.08	170.69	0.010 (0.008)
	Mystery	3 (116)	34.12	14.44	11.81	76.12	-0.009 (0.031)
	Romance	11 (345)	25.49	20.04	0.008	78.72	-0.029 (0.002)
	Sci-Fi	6 (154)	23.67	13.76	0.018	58.22	-0.024 (0.090)
	Sport	0 (67)	-	-	-	-	- (-0.007)
	Thriller	20 (454)	32.24	21.34	0.882	116.74	-0.027 (0.072)
	War	1 (46)	1.31	1.31	1.31	1.31	-0.019 (-0.005)

Table 5: Correlation Coefficient of Movie Variables versus Movie Grosses. The given value of Mean, Median, Min, and Max grosses are in terms of million dallars.

- Entity positive (or negative) frequencies: The number of positive (or negative) occurences of the entity in terms of 7 sentiment categories, i.e., *General*, *Business*, *Crime*, *Health*, *Politics*, *Sports* and *Media*.

For each entity, the below variables are measured:

- Accumulated news references for the first week period before the release of the movie.
- Accumulated news references for the second week through the 4th week period before the release of the movie. We call them the "*1-month*" data.
- Accumulated news references for the 5th week through the 16th week period before the release of the movie. We call them the "*4-month*" data.

All three time periods are mutually exclusive excluded each other. Our study focuses on the pre-release data, but sometimes the 1-week, 1-month, and 4-month post-release data are also considered.

In our experiments, a movie could be counted if and only if the below conditions are satisfied:

- The accumulated frequency of the entity is at least 2 during the 1-week period before the movie's release.
- The movie's total USA gross is at least $20,000.
- In terms of the article counts of movie title, the 1-month data should not more than three times the 1-week data, and the 4-month data should not more than seven times the the 1-week data. This approach filters some movies which have wrong news references data because some occurences of entities are taken into account mistakenly. For example, "*Look*", "*Forever*", "*Diggers*", "*Zoo*", and *Bella*" are movies' names, but in many cases, their occurences are not referring to movies.

Furthermore, our news data include the media coverage of four different kinds of entities:

- Movie Title: News references according to the name of the movie. There are 771 movies after filtering by the news data of movie titles.
- Director: News references according to the director's name of the movie. There are 388 movies after filtering by the news data of directors.
- Top 3 Actors: Total news references of the top 3 actors in the cast of the movie. There are 777 movies after filtering by the news data of top 3 actors.
- Top 15 Actors: Total news references of the top 15 actors in the cast of the movie. There are 827 movies after filtering by the news data of top 15 actors.

To build decent models based on news data, we require movies to have some media coverage for both title and top 3 or 15 actors, and thus we get a smaller movie set. If we consider some other conditions, e.g., director coverage, or budget information, we will make the movie set even smaller.

In Table 6, we divide the movies into 2 different classes according to their budget information, in which *Budget-* means their budget information is not provided in our

Scenario	Low-grossing	High-grossing	Total
Budget-	283	215	498
Budget+	87	158	245

Table 6: Number of movies in different classification sets. "Budget-" means that we don't require that the movie has budget information in our database, while "Budget+" means the opposite.

database, while *Budget+* means their budget information is provided. Clearly, *Budget-* is a bigger data set than *Budget+*. Moreover, people usually pay more attention to high-grossing movies, therefore we further divide the movies into two sub-groups: high-grossing movies and low-grossing movies. The number of available movies in different categories are shown in this table. Movie sets $\text{Movie}_{Budget-}$, $\text{Movie}_{Budget+}$, and their corresponding high-grossing movie sets are the major objects investigated by us in the following chapters.

1. Movie Grosses versus News Reference Counts

   Table 7 shows the correlation analysis of logged pre-release news reference counts versus logged gross or budget under different scenarios. The rows indicate what kind of entities are examined in terms of what kind of duration, i.e., 1 week, 1 month, or 4 months. The columns indicate the correlation is for gross or budget in terms of pre-release(or post-release) article counts. We are using article counts here instead of frequencies because the article counts show better correlation with both budgets and grosses during our experiments.

   From the correlation table, we can see the below significant observations.

   (a) All the correlation coefficient are positive numbers, which means article counts of all movie entities during any specific time period we examined are positively correlated with both movie grosses and movie budgets.

   (b) Raw correlation vs. Logged correlation: Logarithm operation generates higher correlations for news reference counts and grosses (or budget). Table 8 shows this.

   (c) Grosses vs. Budget: News references are more highly correlated with grosses than budgets.

   (d) Pre-release and post-release references: Post-release news references are more strongly correlated with grosses (or budgets) than pre-release news references. Figures 8 and 9 show the sorted raw counts plot, and the corresponding logged scatter plots of the pre-release first week article counts of movies versus movie grosses, which has a correlation coefficient of 0.462. By contrast, figures 10 shows the relevant post-release data of the first week, which has a correlation coefficient of 0.542. Clearly, the post-release data correlate with grosses better.

   (e) Time periods: The 1-week data have the strongest correlation, and the correlations of 1-month data and 4-month data decrease accordingly. Figure 11 shows the 1-month pre-release data has a lower correlation coefficient (0.329)

Entities	Duration	Gross (Pre-rel)	Gross (Post-rel)	Budget (Pre-rel)	Budget (Post-rel)
Movie	1 week	0.707	0.781	0.497	0.480
	1 month	0.672	0.779	0.463	0.474
	4 months	0.629	0.749	0.437	0.455
Director	1 week	0.494	0.602	0.311	0.389
	1 month	0.371	0.495	0.218	0.389
	4 months	0.192	0.317	0.117	0.078
Top 3 Actors	1 week	0.640	0.726	0.476	0.528
	1 month	0.569	0.683	0.448	0.477
	4 months	0.493	0.618	0.413	0.424
Top 15 Actors	1 week	0.646	0.725	0.533	0.595
	1 month	0.575	0.686	0.477	0.530
	4 months	0.511	0.618	0.415	0.433

Table 7: Correlation Coefficient of Logged Pre-release News References vesus Logged Grosses under various scenarios

Correlation	Article Counts			Frequencies		
	1 week	1 month	4 months	1 week	1 month	4 months
Raw Counts	0.462	0.329	0.238	0.465	0.388	0.278
Logged Counts	0.707	0.672	0.629	0.697	0.669	0.625

Table 8: The correlation of pre-release movie article counts and frequencies versus grosses, for raw counts and logged counts respectively.

with grosses, and figure 12 shows that the 4-month pre-release data has a even lower coefficient (0.238) with grosses.

(f) News entities: Directors have the least correlation, movie titles have better correlation, and top actors have the best correlation with grosses (or budget). Moreover, the reference counts for the top 15 actors perform better than those of the top 3 actors in terms of the correlation with grosses.

(g) 7 sentiment categories: Table 9 shows that "*General*" and "*Media*" sentiment counts have the highest correlation with grosses among all the 7 sentiment categories. In the following chapters, we will always use the "*General*" sentiment counts for our analyses.

(h) Negative references vs. positive references: From Table 9, We can also see that positive references are better correlated with grosses than negative ones for all sentiment categories except "*Crime*" and "*Health*". The possible reason is that a movie will be more attractive if it contains the plot that some people are executed or killed.

2. Pairwise Correlation of Various News Statistical Measures

As an example, we create the pairwise correlation table 10 for various 1-week pre-release measures of movie titles and top 15 actors. From this table, it is clear that article counts, frequencies, positive frequencies, and negative frequencies are strongly correlated. Moreover, the news references for movie titles and top 15

Scenarios		General	Business	Crime	Health	Politics	Sports	Media
1 week	Positive	0.692	0.666	**0.418**	**0.520**	0.615	0.684	0.695
	Negative	0.665	0.564	**0.594**	**0.624**	0.565	0.444	0.513
1 month	Positive	0.665	0.651	0.401	0.520	0.603	0.669	0.675
	Negative	0.650	0.579	0.580	0.616	0.564	0.466	0.507
4 months	Positive	0.625	0.626	0.370	0.497	0.561	0.635	0.643
	Negative	0.608	0.544	0.541	0.557	0.531	0.438	0.490

Table 9: Logged Movie Grosses versus Logged Pre-release Positive and Negative Sentiment Counts in Seven Sentiment Categories, in terms of movie title coverage.

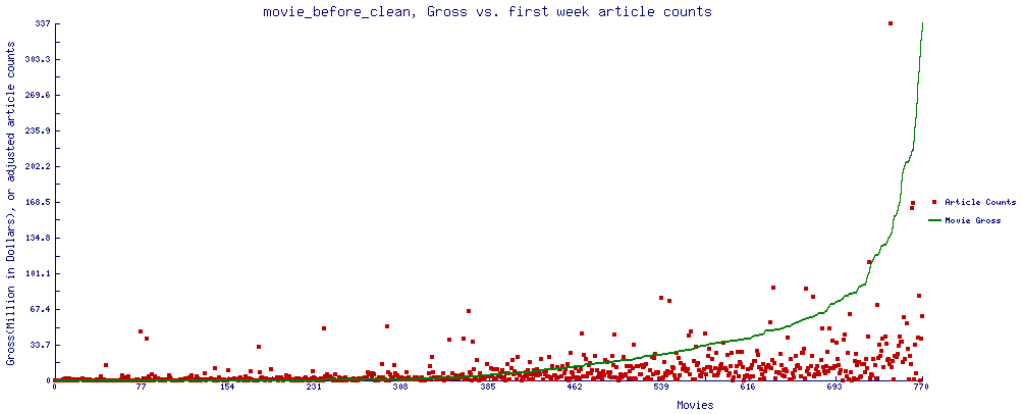

Figure 8: Movie news article counts(the first week before release) vs. grosses, sorted by grosses. Correlation coefficient is 0.462.

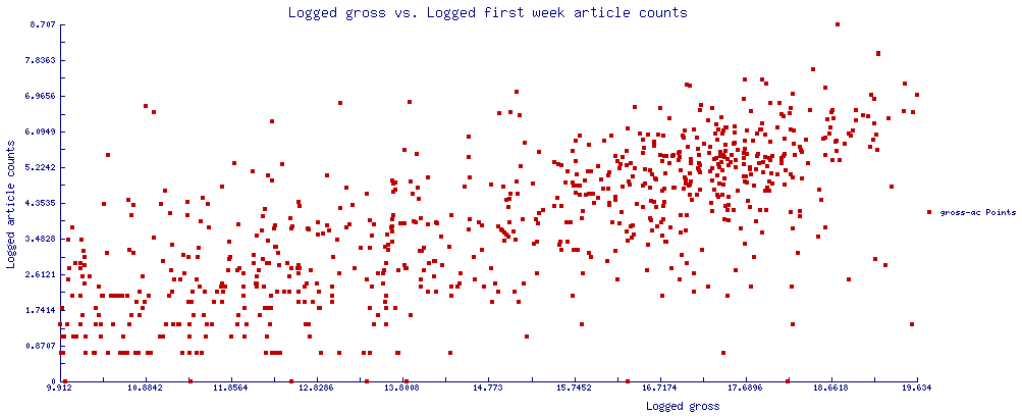

Figure 9: Scatter plot of Logged movie news article counts(the first week before release) vs. Logged grosses. Correlation coefficient is 0.462.

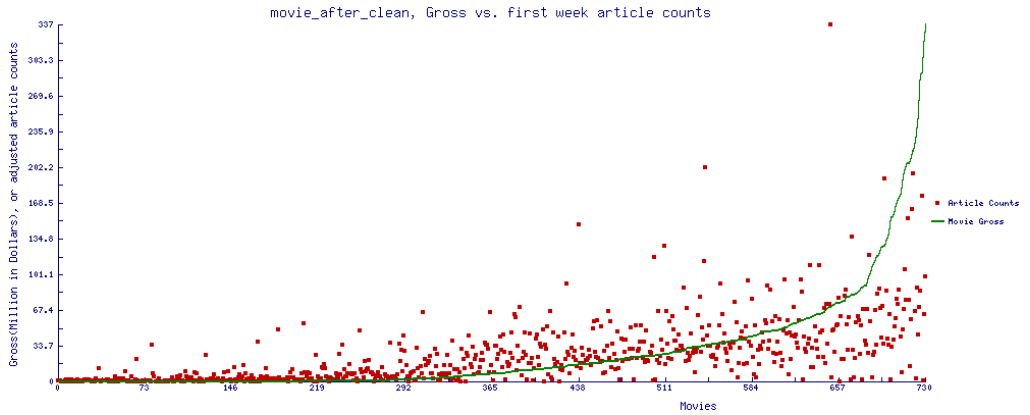

Figure 10: Movie news article counts(the first week after release) vs. grosses, sorted by grosses. Correlation coefficient is 0.542

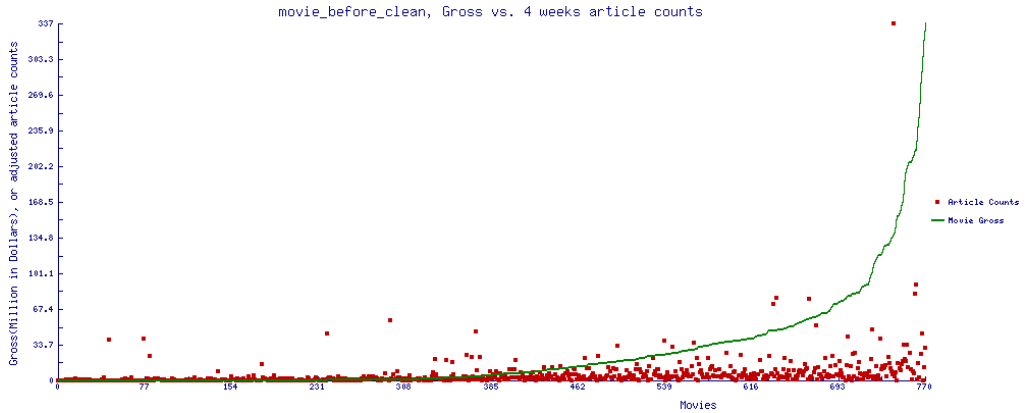

Figure 11: Movie news article counts(the four weeks before release) vs. grosses, sorted by grosses. Correlation coefficient is 0.329.

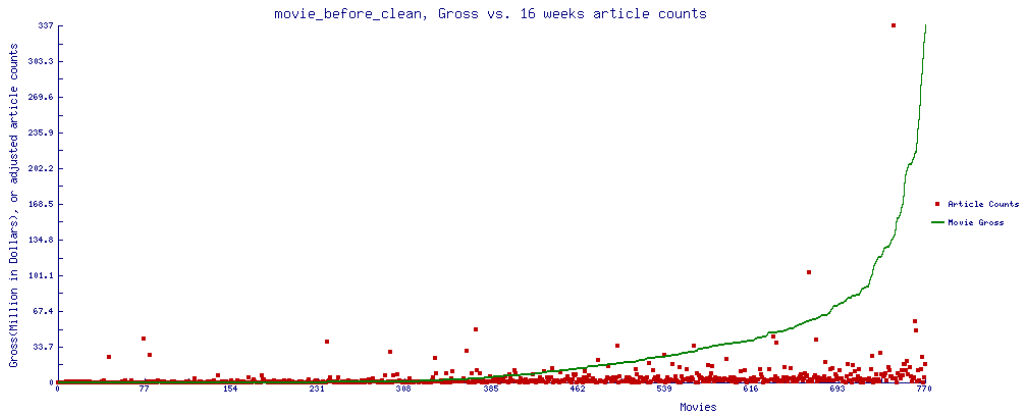

Figure 12: Movie news article counts(the 16 weeks before release) vs. grosses, sorted by grosses. Correlation coefficient is 0.238.

Entities	References	Movie				Top 15 Actors			
		Freqs	Articles	Positive	Negative	Freqs	Articles	Positive	Negative
Movie	Freqs	1.000	0.905	0.969	0.883	0.425	0.499	0.417	0.365
	Articles	0.905	1.000	0.862	0.793	0.527	0.587	0.509	0.459
	Positive	0.969	0.862	1.000	0.861	0.435	0.514	0.440	0.377
	Negative	0.883	0.793	0.861	1.000	0.339	0.411	0.335	0.326
Top 15 Actors	Freqs	0.425	0.527	0.435	0.339	1.000	0.924	0.981	0.949
	Articles	0.499	0.587	0.514	0.411	0.924	1.000	0.892	0.859
	Positive	0.417	0.509	0.440	0.335	0.981	0.892	1.000	0.950
	Negative	0.365	0.459	0.377	0.326	0.949	0.859	0.950	1.000

Table 10: Pairwise Correlation Analysis for news references (1 week pre-release data).

Entities	Duration	Set $\text{Movie}_{Budget-}$			Set $\text{Movie}_{Budget+}$		
		Total	Low	High	Total	Low	High
Movie	1 week	0.747	0.611	0.323	0.622	0.602	0.326
	1 month	0.570	0.479	0.200	0.400	0.405	0.205
	4 months	0.490	0.431	0.209	0.349	0.339	0.198
Top 3 Actors	1 week	0.599	0.451	0.222	0.449	0.499	0.249
	1 month	0.452	0.330	0.195	0.255	0.247	0.204
	4 months	0.386	0.247	0.243	0.237	0.144	0.240
Top 15 Actors	1 week	0.645	0.437	0.430	0.539	0.439	0.464
	1 month	0.459	0.303	0.309	0.246	0.142	0.323
	4 months	0.417	0.251	0.331	0.229	0.085	0.325

Table 11: Pre-release Correlation Analysis (Logged article counts vs. Logged grosses) for low-grossing and high-gross movies.

actors are also positively correlated.

3. Low-grossing Movies versus High-grossing Movies

   Table 11 shows the correlation of logged news references and logged grosses for low-grossing and high-gross movies as well as all the movies.

   (a) For low-grossing movies, the news references for top 3 actors are better predictors for grosses than those of the top 15 actors. However, for high-grossing movies, the news references for the top 15 actors are better predictors than those of the top 3 actors. The reason is that there are more famous actors among the casts for high gross movies.

   (b) The references for movie titles matter more for low-grossing movies than high-grossing movies.

4. Sentiment Statistics versus Grosses for Low-grossing and High-grossing Movies

   Figures 12 and 13 show the correlation coefficient of grosses and difference sentiment measures for movie set $\text{Movie}_{Budget-}$ and $\text{Movie}_{Budget+}$ respectively.

   Apart from positive and negative sentiment counts, the sentiment measures also include:

   - $entity\_polarity = \frac{positive\_sentiment\_references}{total\_sentiment\_references}$

Entities	Statistic	All			Low			High		
		1 week	1 mon	4 mons	1 week	1 mon	4 mons	1 week	1 mon	4 mons
Movie	Positive	0.455	0.288	0.204	0.371	0.123	0.125	0.317	0.223	0.127
	Negative	0.371	0.194	0.130	0.319	0.176	0.098	0.218	0.091	0.048
	Polarity	***0.042***	0.097	0.137	-0.080	0.125	-0.011	***0.154***	0.094	0.148
	Subjectivity	***-0.069***	-0.014	0.066	-0.022	0.057	0.095	***-0.214***	-0.151	0.020
	PositivePer	-0.047	0.001	0.095	-0.079	0.055	0.069	-0.093	-0.088	0.077
	NegativePer	***-0.054***	-0.025	0.017	0.060	0.039	0.082	***-0.209***	-0.139	-0.035
	DifferencesPer	0.001	0.021	0.062	-0.103	0.023	-0.014	0.099	0.044	0.093
Top 3 Actors	Positive	0.336	0.143	0.140	0.441	0.338	0.298	0.165	0.054	0.053
	Negative	0.299	0.092	0.084	0.460	0.323	0.247	0.145	0.012	0.007
	Polarity	-0.005	0.085	0.062	-0.031	0.150	0.095	-0.029	0.024	0.077
	Subjectivity	-0.008	-0.015	0.021	0.023	0.098	0.071	-0.083	-0.140	-0.144
	PositivePer	-0.005	0.027	0.031	-0.021	0.134	0.107	-0.047	-0.086	-0.120
	NegativePer	-0.007	-0.052	-0.001	0.068	0.020	0.000	-0.069	-0.132	-0.120
	DifferencesPer	0.001	0.062	0.030	-0.067	0.089	0.093	0.022	0.023	-0.040
Top 15 Actors	Positive	0.266	0.117	0.130	0.085	0.036	0.037	0.250	0.101	0.108
	Negative	0.219	0.062	0.061	0.084	0.008	0.005	0.205	0.052	0.039
	Polarity	0.032	0.050	0.046	-0.084	0.048	0.075	0.059	0.039	0.040
	Subjectivity	0.024	-0.009	0.079	-0.014	0.004	0.059	-0.044	-0.090	-0.054
	PositivePer	0.037	0.033	0.099	-0.071	0.036	0.095	0.005	-0.047	-0.028
	NegativePer	-0.003	-0.047	0.028	0.061	-0.027	0.002	-0.073	-0.096	-0.063
	DifferencesPer	0.034	0.060	0.060	-0.106	0.047	0.074	0.057	0.023	0.025

Table 12: Pre-release correlation analysis for Movie Set Movie$_{Budget-}$: Sentiment statistics vs. grosses. The bold and italic figures show these statistics are significantly correlated with movie grosses. These statistic variables include the polarity, subjectivity, and negative references per reference of a movie's 1-week pre-release references in terms of movie titles.

- $entity\_subjectivity = \frac{total\_sentiment\_references}{total\_references}$
- $positive\_refs\_per\_ref = \frac{positive\_sentiment\_references}{total\_references}$
- $negative\_refs\_per\_ref = \frac{negative\_sentiment\_references}{total\_references}$
- $sentiment\_diffs\_per\_ref = \frac{positive\_sentiment\_references - negative\_sentiment\_references}{total\_references}$

Those factors that have high correlation coefficients with grosses will be used in our prediction model later. For example, in movie set Movie$_{Budget+}$, the 1-week polarity and subjectivity measures of movie titles for high gross movies are 0.173 and -0.266 respectively, which are significant. Other strong indicators are also highlighted in the the two tables as well.

4.5 Evaluation Methods

We are using the "Modeling-Predicting-Evaluating" approach to forecast the movie grosses. "Modeling" is the training phase to give parameter estimations. "Predicting" is the execution phase to give the prediction for movie grosses. "Evaluating" is the phase to evaluate how good or how bad the model is. There are a couple of methods to evaluate the performance of our models as follows.

Here we suppose G is the actual gross and P is the predicted gross. If we forecast the grosses for n movies, $\{G_i | 1 \leq i \leq n\}$ is the actual gross set of the movies, while $\{P_i | 1 \leq i \leq n\}$ is the corresponding predicted gross set for movies.

- Percentile Differences and Absolute Percentile Differences.

Entities	Statistic	All			Low			High		
		1 week	1 mon	4 mons	1 week	1 mon	4 mons	1 week	1 mon	4 mons
Movie	Positive	0.398	0.255	0.191	0.344	0.074	0.073	0.332	0.241	0.181
	Negative	0.279	0.126	0.104	0.238	0.116	0.110	0.209	0.082	0.060
	Polarity	***0.143***	0.096	0.172	-0.098	0.081	-0.074	***0.173***	0.102	0.167
	Subjectivity	***-0.189***	-0.118	0.021	-0.100	-0.096	0.060	***-0.266***	-0.167	0.048
	PositivePer	-0.096	-0.084	0.061	-0.119	-0.080	0.021	-0.131	-0.107	0.099
	NegativePer	***-0.190***	-0.107	-0.026	-0.034	-0.086	0.075	***-0.242***	-0.147	-0.018
	DifferencesPer	0.072	0.011	0.071	-0.078	-0.005	-0.040	0.099	0.023	0.097
Top 3 Actors	Positive	0.242	0.056	0.074	0.452	0.318	0.347	0.166	0.024	0.034
	Negative	0.213	0.022	0.029	0.475	0.325	0.335	0.145	-0.011	-0.012
	Polarity	-0.007	0.047	0.084	-0.144	0.117	0.141	-0.024	0.032	0.100
	Subjectivity	-0.065	-0.131	-0.076	0.100	-0.036	0.018	-0.088	-0.155	-0.149
	PositivePer	-0.027	-0.082	-0.042	-0.019	-0.005	0.100	-0.051	-0.107	-0.121
	NegativePer	-0.067	-0.122	-0.086	0.171	-0.047	-0.093	-0.070	-0.128	-0.120
	DifferencesPer	0.035	0.038	0.019	-0.180	0.047	0.148	0.020	0.007	-0.038
Top 15 Actors	Positive	0.304	0.083	0.079	0.318	0.045	-0.041	0.241	0.074	0.087
	Negative	0.247	0.031	0.012	0.339	-0.024	-0.068	0.189	0.027	0.017
	Polarity	0.083	***0.091***	0.077	-0.217	***0.259***	0.169	0.076	***0.050***	0.048
	Subjectivity	-0.021	***-0.142***	-0.025	0.098	***-0.217***	-0.041	-0.042	***-0.126***	-0.049
	PositivePer	0.035	-0.055	0.022	-0.029	-0.048	0.085	0.008	-0.075	-0.026
	NegativePer	-0.071	***-0.167***	-0.066	0.193	***-0.276***	-0.134	-0.075	***-0.119***	-0.057
	DifferencesPer	0.087	0.075	0.070	-0.239	0.196	0.165	0.059	0.012	0.018

Table 13: Pre-release correlation analysis for Movie Set Movie$_{Budget+}$: Sentiment statistics vs. grosses. The bold and italic figures show these statistics are significantly correlated with movie grosses. These statistic variables include the polarity, subjectivity, and negative references per reference of a movie's 1-week pre-release references in terms of movie titles, plus the polarity, subjectivity, and negative references per reference of a movie's 1-month pre-release references in terms of top 15 actors.

Percentile difference: $PD = \max_{abs}(\frac{P-G}{G}, \frac{P-G}{P})$
Absolute percentile difference: $APD = |PD| = |\max_{abs}(\frac{P-G}{G}, \frac{P-G}{P})|$
"$\max_{abs}$" is an operator, which chooses the element that has the biggest absolute value in the list follows.

Below specific variables are used to evaluate the overall performance of a model that predicts the grosses of the n movies.

- Mean of PD or APD
- Standard deviation of PD or APD
- Median of PD or APD
- Maximum of PD or APD
- Minimum of PD or APD
- Residual mean: $RM = \frac{\sum_{i=1}^{n} |P_i - G_i|}{n}$

The range of PD is $(-\infty, \infty)$, while the range of APD is $[0, \infty)$. Intuitively, the mean or median of PD should be a number around 0 while the mean or median of APD should be a positive number; the minimum of PD should be a negative while the minimum of APD should be around 0.

The absolute value of PD or APD indicate how far the predicted gross is from the actual gross. If there are two predictions and they have the same absolute value of PD or APD, we regard the two predictions equally good. Furthermore, the PD has sign. A negative sign means the gross is underestimated while a positive sign means the gross is overestimated.

For example, a movie's actual gross is \$50 millions. The prediction is \$75 millions, and thus $PD = 0.50$. Another prediction is \$33.3 millions and thus $PD = -0.50$. We will think \$75 millions and \$33.3 millions are two equally good predictions for this movie.

Another example, the movie's actual gross is \$50 millions, then \$135.9 and \$18.4 also give two equally good predictions because the APD is 1.718 in both cases.

- Logarithmic Ratio and Absolute Logarithmic Ratio.

  Logarithmic Ratio: $LR = \ln(\frac{P}{G})$

  Absolute Logarithmic Ratio: $ALR = |\ln(\frac{P}{G})|$

  Similarly, below specific variables are used to evaluate the overall performance of a model that predicts the grosses of the n movies.

  - Mean of LR or ALR
  - Standard deviation of LR or ALR
  - Median of LR or ALR
  - Maximum of LR or ALR
  - Minimum of LR or ALR
  - Residual mean: $RM = \frac{\sum_{i=1}^{n} |P_i - G_i|}{n}$
  - Summary statistics: $SS = \ln(\frac{\sum_{i=1}^{n} P_i}{\sum_{i=1}^{n} G_i})$, ideally should be 0.

  The range of LR is $(-\infty, \infty)$, while the range of ALR is $[0, \infty)$. Intuitively, the mean or median of LR should be a number around 0 while the mean or median of ALR should be a positive number; the minimum of LR should be a negative while the minimum of ALR should be around 0.

  The absolute value of LR or ALR indicate how far the predicted gross is from the actual gross. If there are two predictions and they have the same absolute value of LR or ALR, we regard the two predictions as equally good. Furthermore, the LR has sign. A negative sign means the gross is underestimated while a positive sign means the gross is overestimated.

  Again, if a movie's actual gross is \$50 million. The prediction \$75 million and \$33.3 million are equally good as an estimation because they have $ALR = \ln(\frac{3}{2})$. Also, the prediction \$135.9 millions and \$18.4 million give two equally good predictions because both of them have a ALR value of $\ln(2.718) = 1$.

  In fact, the logarithmic ratio and the percentile difference of actual and predicted grosses are convertible from each other.

  Because $LR = \ln(\frac{P}{G})$, we have $\frac{P}{G} = e^{LR}$,

  Then $PD = \max_{abs}(\frac{P-G}{G}, \frac{P-G}{P}) = \max_{abs}(e^{LR} - 1, 1 - \frac{1}{e^{LR}})$

- Score. Score is defined by:

  $Score = \frac{\sum_{i=1}^{n} (100 - \min(100, APD_i))}{n}$

  According to the formula, the full score is 100 and the minimum score is 0.

- $\alpha\%$ percentage coverage.

  The $\alpha\%$ percentage coverage is defined by:

$PC_{\alpha\%} = \frac{Number\ of\ movies\ whose\ APD \leq \alpha\%}{Total\ number\ of\ movies\ (n)}$

For example, if we predict 100 movies totally and 28 movies have a APD value less than or equal to 50%, the 50% percentage coverage is $PC_{50\%} = \frac{28}{100} = 28\%$.

Obviously, if a prediction needs to be in the 50% percentage coverage, it must have $APD \leq 50\%$, say, $\frac{2}{3} \leq \frac{P}{G} \leq \frac{3}{2}$. If a prediction needs to be in the 100% percentage coverage, it must have $\frac{1}{2} \leq \frac{P}{G} \leq 2$.

More generally, if a prediction need to be in the $\alpha\%$ percentage coverage, it must have $\frac{1}{1+\alpha\%} \leq \frac{P}{G} \leq (1 + \alpha\%)$.

One example is, we assume a movie's actual gross is \$50 million. The prediction has to have a value of \$33.3 $million \leq P \leq$ \$75 $million$ if it wants to be in the 50% percentage coverage, while the prediction has to have a value of \$25 $million \leq P \leq$ \$100 $million$ if it wants to be in the 100% percentage coverage.

In our performance evaluation, the 50%, 100%, 200% and 800% percentage coverage are compared among models.

4.6 Prediction with Traditional Movie Variables

To predict movie grosses with traditional movie variables, we build three kinds of models: regression models, piecewise models, and k-nearest neighbor models. We are following the "training-predicting-evaluating" three steps to set up the models and evaluate their performance. Therefore, our movie data set is divided into two parts, the training data set and the predicting data set. The training data set helps us to build models and tune parameters elaborately, while the predicting data set helps us to verify whether it is a good model.

As we mentioned before, we classify the movies into several different classification sets according to their gross scale, budget number availabilities, and media exposure regarding movie related entities. We can get only a relatively small movie set if we require both movie database coverage and news coverage within a period from November 2004 to March 2008, therefore we will build the general models with a much larger movie set firstly. These general models will be applied to the news movie set (the movie set with news coverage) as well.

Table 14 shows the size of all movie sets, their corresponding training and predicting sets used in this paper. The set S includes all the movies from 2000 to 2007, in which the first five year data is the training data set and the last three year data is the predicting data set. $\text{Movie}_{Budget-}$ and $\text{Movie}_{Budget+}$ represent the news movie sets that have available budget information and do not have available budget information respectively.

As for the performance evaluation, APD and LR are used. APD evaluates how far the predicted grosses are away from the actual grosses, while LR evaluates in which direction the predicted grosses are away from the actual grosses, i.e., whether the grosses are overestimated or underestimated.

Movie Sets	Set S			Movie$_{Budget-}$		Movie$_{Budget+}$	
	Total	Low	High	Total	High	Total	High
Total	984	371	613	498	215	245	158
Training Set	638	228	410	280	119	151	102
Predicting Set	346	143	203	218	96	94	56

Table 14: Number of Movies in the training and predicting sets of movie set S, Movie$_{Budget-}$, and Movie$_{Budget+}$. Movie$_{Budget-}$ and Movie$_{Budget+}$ represent the news movie sets that have available budget information and do not have available budget information respectively. In this section, our models are based on set S. From the next section, our models are based on sets Movie$_{Budget-}$ and Movie$_{Budget+}$ because we need news coverage.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
Reg$_{base}$	All	9.11	58.48	1.25	996.92	0	0.21	0.07	94.35
Reg$_{budget}$	All	6.65	42.97	0.98	750.17	0.01	0.23	0.06	95.63
Reg$_{genre}$	All	10.51	77.9	1.28	1340.9	0	0.21	0.02	94.31
Reg$_{adv1}$	All	**6.26**	38.46	1.04	669.02	0.01	0.25	0.08	**95.71**
Reg$_{adv2}$	All	**5.96**	31.62	1.03	520.43	0	0.28	0.07	**95.64**
Piece$_{adv1}$	All	**3.3**	20.94	0.73	365.33	0.01	0.11	0.03	**97.53**
Reg$_{base}$	High	3.34	9.16	0.82	64.96	0	-0.54	-0.33	96.66
Reg$_{budget}$	High	1.8	3.91	0.66	39.94	0.01	-0.41	-0.32	98.2
Reg$_{genre}$	High	3.43	9.74	0.8	69.99	0	-0.53	-0.39	96.57
Reg$_{adv1}$	High	**1.77**	3.91	0.68	38.84	0.01	-0.4	-0.34	**98.23**
Reg$_{adv2}$	High	**1.8**	4.04	0.67	42.13	0	-0.36	-0.3	**98.2**
Piece$_{adv1}$	High	**0.82**	0.81	0.59	5.26	0.01	0.00	0.01	**99.18**

Table 15: Performance of Linear Regression Models and Piecewise Linear Models. The performance for high-grossing movies is better than the overall performance. Among regression models, Reg$_{adv1}$ and Reg$_{adv2}$ perform best. Piecewise linear model performs better than regular regression models.

4.6.1 Regression Models

The first model we built is a regression model, in which movie variables are predictors and gross is the responsor. Based on our previous correlation analysis, we can make both numerical and categorical movie variables as predictor variables because they are correlated with grosses. Genres is a special variable because we need to design up to 18 dimensions for the 18 genres, therefore we design separate models with or without genre information. Moreover, budget is another special variable. Budget is an important indicator for grosses, but only less than half of the movies in our database have budget values. Thus we need to set up models for both Movie$_{Budget-}$ and Movie$_{Budget+}$. Based on these, 5 regression models and their performance comparison are shown in table 15.

1. Basic Regression Model (Reg$_{base}$)

   In this model, 8 indicators are included: holiday flag(h) to indicate the movie is released during holiday or summer period or not, four MPAA rating flags(g, pg, $pg13$, and r), sequel flag(h) to indicate the movie is a sequel of previous movie or not, foreign flag(f) to indicate the movie is a foreign movie or not, theater counts(tc) to indicate the opening screens. After regressing data set $S1_T$, the following formula is given(g is the movie gross):

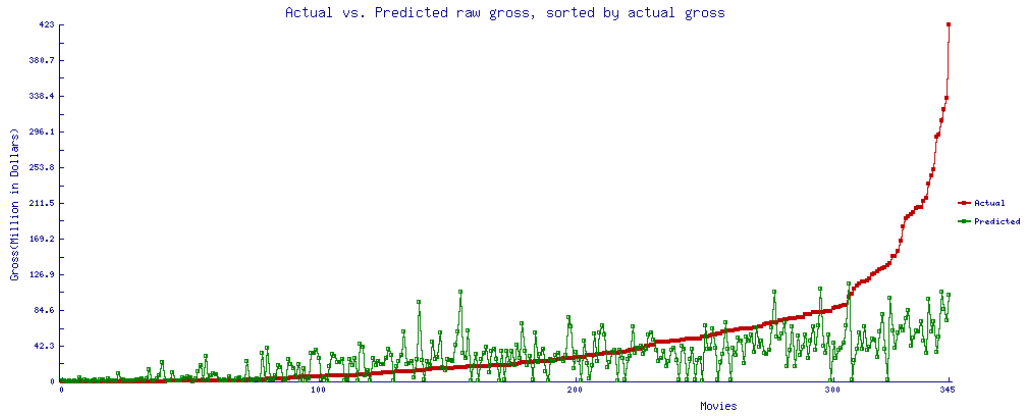

Figure 13: Predicted Grosses vs. Actual Grosses for model Reg_{base}, sorted by grosses. The rising line shows movies' actual grosses, while the fluctuatedline shows the corresponding predicted grosses.

$log(g) = 12.897 + 0.367 * h + 0.693 * g + 0.918 * pg + 0.733 * pg13 + 0.464 * r + 0.503 * s - 0.170 * f + 0.475 * log(tc)$

This formula can be used to predict future movie grosses based on the related movie variables. As we expected, the coefficients of pg, $pg13$, and r are decreasing gradually. Moreover, the coefficients of h, s, and tc are positive numbers, while the coefficient of f is negative numbers. All of these are identical with the previous correlation analysis.

If we forecast the movie grosses for the following three years, the "Actual-Predicted" gross plot can be shown in figure 13. This graph clearly shows that the low gross movies are overestimated while high gross movies are underestimated in a large sense.

Table 15 shows the high gross movies are predicted better because grosses of low gross movies are even harder to predict than high gross movies. We should notice that, the LR mean value is -0.54 for high gross movies using model Reg_{base}, which proves the high gross movies are underestimated because on average the predicted grosses are lower than actual grosses.

2. Basic Budget Regression Model (Reg_{budget})

   This model is the same as previous model, but adds one more indicator - "budget" (b). Accordingly, the relevant formula is changed to:

   $log(g) = 5.1969 + 0.193 * h + 0.637 * g + 0.764 * pg + 0.518 * pg13 + 0.300 * r + 0.430 * s - 0.389 * f + 0.254 * tc + 0.560 * b$

   From performance table 15, we can see that the budget indicator improves the model performance greatly, especially for the high gross movies.

3. Basic Regression Model with Genres (Reg_{genre})

   This model is the same as model Reg_{base}, but adds genre indicators. Only movies' primary genres are considered in this model. According to table 5, the 6 genre indicators we added are: *Action*, *Adventure*, *Animation*, *Biography*, *Documentary*,

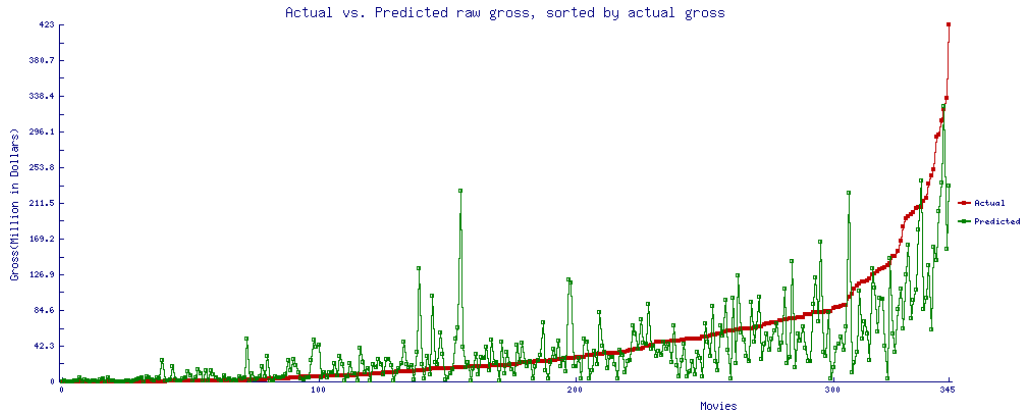

Figure 14: Predicted Grosses vs. Actual Grosses for model Reg_{adv2}, sorted by grosses

Drama. Table 15 shows the performance of Reg_{genre} is similar with Reg_{base} but worse than Reg_{budget}.

4. Advanced Regression Model (1) (Reg_{adv1})

   This model adds genres information based on model Reg_{budget}. Only the primary genre of a movie is considered in this model. The six genre indicators we used are: *Action*, *Adventure*, *Animation*, *Biography*, *Documentary*, and *Drama*. Table 15 indicates the movie grosses are forecast better than all previous models.

5. Advanced Regression Model (2) (Reg_{adv2})

   This model is very similar to model Reg_{adv1}. The only difference is that it uses not only the primary genre, but also the subsidiary genres. For example, because "*Forrest Gump*" has the genres of *Comedy*, *Drama* and *Romance*, flags for all the three genres will be set in this model. 9 genre indicators are used in this model: *Action*, *Adventure*, *Animation*, *Biography*, *Documentary*, *Drama*, *Family*, *Fantasy* and *Sci-Fi*. Therefore, this model has 18 dimensions in total. This model gains comparable performance, but only a little bit worse than model Reg_{adv1}. The "Actual-Predicted" gross plot is shown in figure 14. Compared to graph 13, this model has much better rising curve for predicted grosses.

Figure 15 shows the performance comparison of regression models Reg_{base}, Reg_{budget}, Reg_{genre}, and Reg_{adv1}, both for overall performance and performance of high-grossing movies. From this figure, we can find model Reg_{adv1} has the best performance among all of them, both overall performance and high-grossing performance.

4.6.2 Piecewise Linear Models

All previous models are linear models and they are trying to give the uniform description for all movies. But actually we always overestimate the low gross movies and underestimate the high gross movies. A straightforward improvement method is to divide movies into two subsets, a low-grossing subset and a high-grossing subset, and

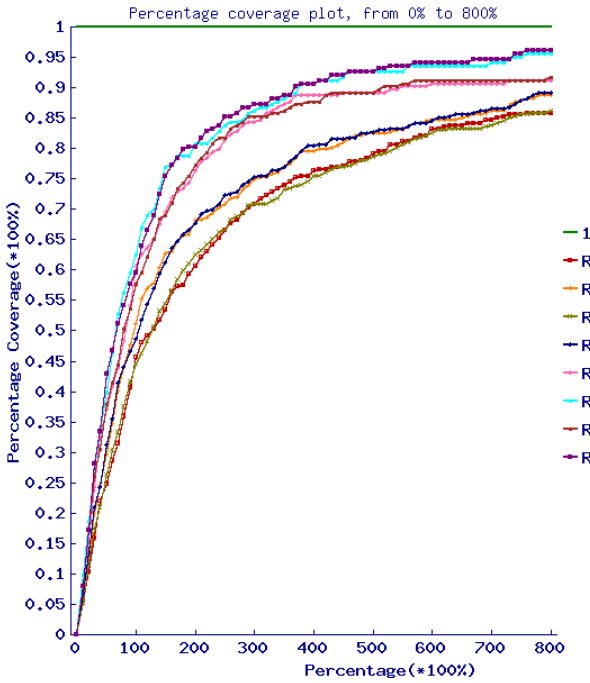

Figure 15: Performance comparison of four regression models: Reg_{base}, Reg_{budget}, Reg_{genre}, and Reg_{adv1}, both for overall performance and performance of high-grossing part. The performance of high-grossing movies is better than the overall performance, and model Reg_{adv1} has the best performance in both cases.

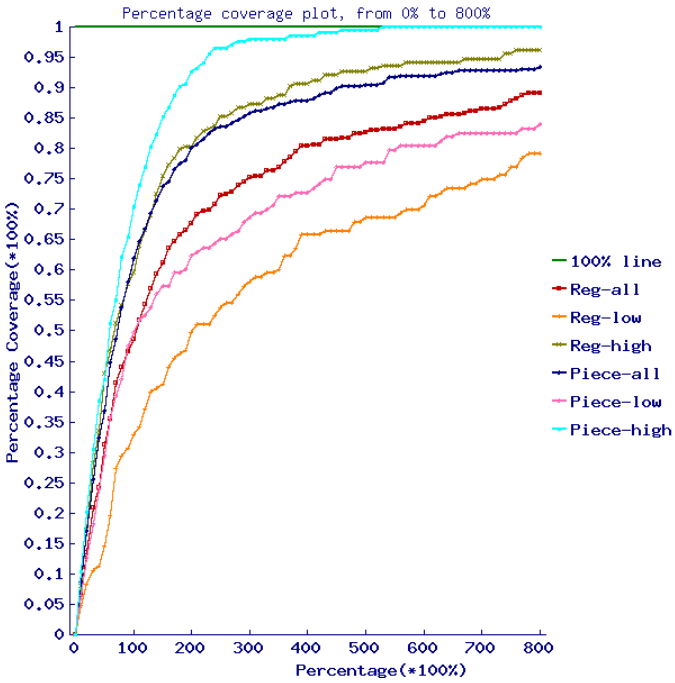

Figure 16: Performance comparison of regression model Reg_{adv1} and piecewise model. The overall performance, high-grossing performance and low-grossing performance of piecewise model are all better than those of model Reg_{adv1}. High-grossing performance is better than low-grossing performance.

design separate models for each subset. Here we make 15 million as the threshold to differentiate low and high gross movies and set up a regression model for each subset using the same variables as model Reg_{adv1}. The piecewise model is called Piece_{adv1}. From the performance data in table 15 and figure 16, we can see both the overall performance, the low-grossing performance, and high-grossing performance are all improved. The 800% percentage coverage for high gross movies are as high as 100%.

4.6.3 K-Nearest Neighbor Models

Different from regression models, another approach to forecast a movie's gross is to find a "similar" movie from previously released movies and make the gross of this "similar" movie as the predicted gross of the target movie. The reason of this method is that "similar" movies usually have similar grosses. The similarity of movies could be measured by "distance", which is further evaluated in a multi-dimensional space. Two methods are used to measure the distance of movies: regression method and normalization method.

In the first method, the distance of each dimension is defined as follows:

- Budget Distance d_b

  The distance of two budget value $budget1$, $budget2$ are calculated as:

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
$k\text{NN}'_{base}(k=1)$	All	32.36	173.23	2.02	2436	0.01	0.31	0.4	87.01
$k\text{NN}_{base}(k=1)$	All	52.52	574.54	1.28	10197	0	0.37	0.25	92.16
$k\text{NN}_{base}(k=3)$	All	38.49	360.92	1.11	4674	0	0.64	0.41	93.61
$k\text{NN}_{base}(k=5)$	All	26.77	226.15	1	2928	0.01	0.74	0.45	93.92
$k\text{NN}_{base}(k=7)$	All	25.49	219.64	1.04	3336	0	0.8	0.47	93.43
$k\text{NN}_{base}(k=9)$	All	21.89	176.76	1.07	2626	0.01	0.81	0.43	93.27
$k\text{NN}_{nobudget}(k=1)$	All	377.03	6608	1.3	122893	0	0.49	0.33	91.62
$k\text{NN}_{nobudget}(k=7)$	All	**98.18**	1214	1.22	21435	0	0.99	0.64	**90.79**
$k\text{NN}_{adv1}(k=1)$	All	36.8	206.38	2.02	2983	0	0.26	0.13	86.95
$k\text{NN}_{adv2}(k=1)$	All	13.85	86.94	1.39	1367	0	0.38	0.31	92.93
$k\text{NN}_{adv2}(k=3)$	All	159.82	2272.8	1.15	41136	0	0.73	0.48	92.88
$k\text{NN}_{adv2}(k=7)$	All	**80.20**	1081.3	1.13	19643	0	0.92	0.59	**92.07**
$k\text{NN}'_{adv2}(k=3)$	All	159.68	2272.7	1.15	41136	0	0.68	0.39	93.16
$k\text{NN}'_{base}(k=1)$	High	9.63	37.46	1.25	330.86	0.01	-0.43	-0.17	93.14
$k\text{NN}_{base}(k=1)$	High	14.88	181.79	0.68	2591.0	0	-0.14	-0.01	97.39
$k\text{NN}_{base}(k=3)$	High	2.69	17.63	0.65	230.62	0	-0.04	0.07	97.97
$k\text{NN}_{base}(k=5)$	High	1.15	2.57	0.55	30.53	0.01	0.04	0.11	98.85
$k\text{NN}_{base}(k=7)$	High	1.16	3.17	0.52	41.48	0	0.05	0.17	98.84
$k\text{NN}_{base}(k=9)$	High	1.17	3.74	0.48	50.5	0.01	0.05	0.14	98.83
$k\text{NN}_{nobudget}(k=1)$	High	6.59	31.04	0.68	261.86	0.01	-0.18	0.04	95.72
$k\text{NN}_{nobudget}(k=7)$	High	**1.19**	2.14	0.6	18.84	0	0.07	0.15	**98.81**
$k\text{NN}_{adv1}(k=1)$	High	11.39	47.05	1.36	556.17	0.01	-0.65	-0.37	92.2
$k\text{NN}_{adv2}(k=1)$	High	4.94	23.6	0.7	261.86	0.01	-0.2	0.04	96.2
$k\text{NN}_{adv2}(k=3)$	High	1.51	3.67	0.6	36.1	0	-0.03	0.06	98.49
$k\text{NN}_{adv2}(k=7)$	High	**1.1**	2.14	0.61	24.14	0	0.07	0.12	**98.9**
$k\text{NN}'_{adv2}(k=3)$	High	1.47	5.1	0.5	32.27	0	-0.03	0.06	98.56

Table 16: Performance comparison of k-Nearest Neighbor Models. High-grossing performance is much better than overall performance. $k\text{NN}_{adv2}(k=7)$ is the best model for movie set $Movie_{Budget+}$, while $k\text{NN}_{nobudget}(k=7)$ is the best model for movie set $Movie_{Budget-}$.

$dis(budget1, budget2) = \frac{max(budget1,budget2)-min(budget1,budget2)}{min(budget1,budget2)}$

- Opening Screen Distance d_o

  The distance of two opening screens $screen1$, $screen2$ are calculated as:
  $dis(screen1, screen2) = \frac{max(screen1,screen2)-min(screen1,screen2)}{min(screen1,screen2)}$

- Release Date Distance d_d

  If both the two movies are (or aren't) released during holiday or summer period, the distance of their release dates is 0; otherwise, their distance is 1.

- MPAA Rating Distance d_m

  We assign number 4, 3, 2, 1, and 0 for MPAA rating G, PG, PG-13, R, and NC-17 respectively. Two movies' MPAA rating distance is the absolute difference of their assigned numbers.

- Origin Country Distance d_c

  If two movies' origin countries are the same, their origin country distance is 0; if one movie is USA movies while the other is foreign movie, their distance is 1; if two movies are from two difference foreign countries, their distance is 0.5.

- Source Distance d_s

  If both the two movies are (or aren't) sequels of some previous movies, their source distance is 0; otherwise, the distance is 1.

- Genre Distance d_g

  If the primary genres of two movies are the same, their genre distance is 0; otherwise, the distance is 1.

However, we cannot add distances for all the dimensions together to make it the overall distance of two movies because difference dimensions have difference weights in terms of their impact to grosses. Therefore, the below regression equation on training set is used to get the coefficients for each dimensions.

$d_{gross} = C_0 + C_b * d_b + C_o * d_o + C_d * d_d + C_m * d_m + C_c * d_c + C_s * d_s + C_g * d_g$

Thus, the distance of two movies are calculated by:

$dis = C_b * d_b + C_o * d_o + C_d * d_d + C_m * d_m + C_c * d_c + C_s * d_s + C_g * d_g$

After this, the distances between the target movie and all movies in the training set are computed and ranked. The one with the least distance is called the nearest neighbor of the target movie. We can make the gross of the nearest neighbor or the mean gross of the k-nearest neighbors as the predicted gross of the target movie.

In the second method, we normalize the movie variables by subtracting the training set's mean and dividing by the training set's standard deviation of the variable.

For example, the budget value b can be normalized by $b_n = \frac{b-\mu}{\sigma}$. μ is the mean of the budget of the training set, while σ is the standard deviation of the training set. Therefore, the distance of two budget numbers are defined as $dis(b1, b2) = abs(b1_n - b2_n)$.

Similarly, we can normalize opening screens and all other indicators and get the distances. Because they are in the same scale after normalization, we can add them together and to make it as the distance of two movies. Euclidean distance could also be used instead of Manhattan distance, and the distance comparison result will not be affected because all the distance are positive values.

Based on the two distance methods and IMDB information availability, different K-NN models are developed. The performance data are shown in table 16.

1. K-NN Models based on Numerical Indicators Only ($k\text{NN}_{base}$)

   Numerical indicators include movie budgets and opening screens. After regressing the training data set, we get the distance formula: $dis = 0.200 * d_b + 1.693 * d_s$.

   - Comparison of the two distance methods(regression method and normalization method):
     In table 16, model $k\text{NN}_{base}$ shows the performance of regression distance method, and model $k\text{NN}'_{base}$ shows the performance of normalization distance method. Clearly both the overall performance and the high-grossing performance of regression method are much better than those of normalization method. The reason is that different movie variables have different scales in terms of their influence on movie grosses but the normalization method ignores the difference. In the following models, We all use the regression method to compute the distances of movies.
   - The impact of K:
     Table 16 shows the performance data of model $k\text{NN}_{base}$ while k is 1, 3, 5, 7, and 9 respectively. Figure 17 shows the "Actual-Predicted" plot while $k = 1$,

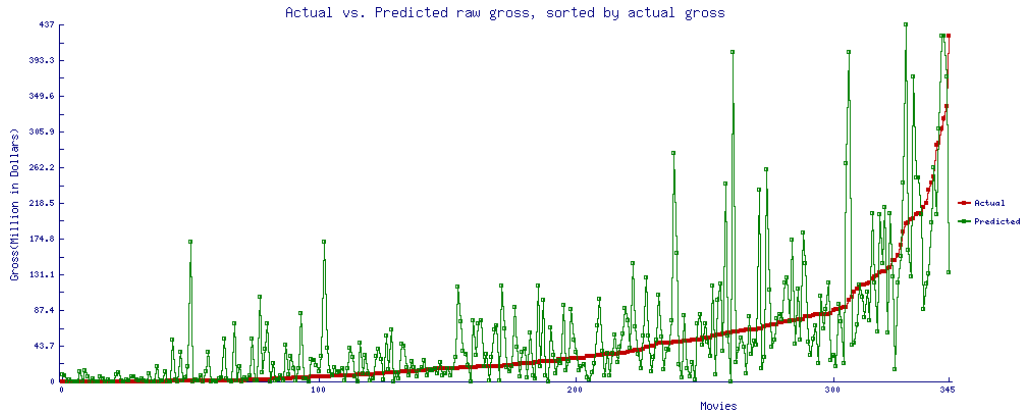

Figure 17: Predicted Grosses vs. Actual Grosses for model $k\text{NN}_{base}(k = 1)$, sorted by grosses

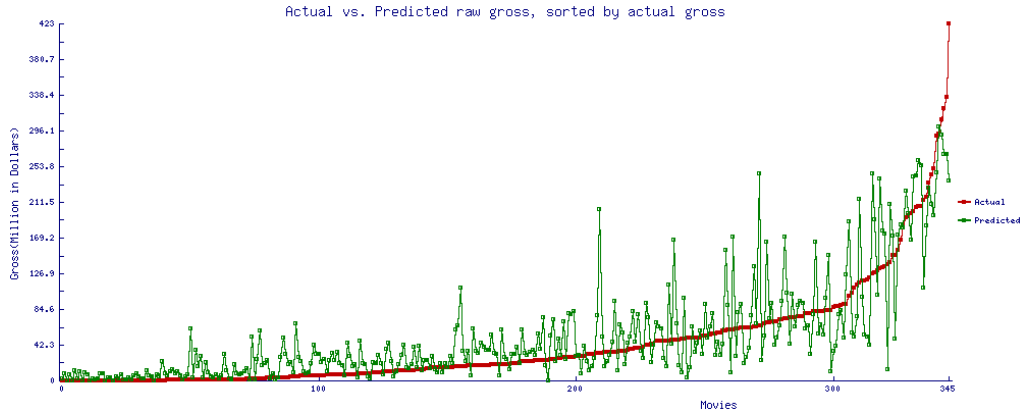

Figure 18: Predicted Grosses vs. Actual Grosses for model $k\text{NN}_{base}(k = 7)$, sorted by grosses. The result is better than that in figure 17.

and figure 18 shows the plot while $k = 7$. From the two plots, we can see some high-grossing movies are highly overestimated. However, to take the mean of k-nearest neighbors smooth those overestimated values. Figure 19 shows the percentage coverage plot for model $k\text{NN}_{base}$. As we can see, the model works worst when $K = 1$, while other models work almost equally good, but with slightly performance gain with the increasing of k. Actually, the model works good enough while $k = 7$. In the following, we will only examine k-NN models with $k = 1$ and $k = 7$.

- Low-grossing Movies vs. High-grossing Movies:
  From plot 17 and 18, we can find many movies are overestimated, especially for low-grossing movies. Based the performance data, all the LR means of the overall performance of the above five models are positive value but the LR means of high-grossing movies are negative numbers with much smaller absolute values, which indicates generally the high-grossing movies are slightly underestimated but the low-grossing movies are highly overestimated. Plot

No.	MPAA	Genre	Source	Cout.	***Scrns***	***Bgt($M)***	**Gro($M)**	Date	Name
1	R	Action	Original Screenplay	USA	2	1.000	0.000884	04/21/06	In Her Line of Fire
	R	Adventure	Original Screenplay	USA	2	1.000	9.015	12/02/05	Transamerica
2	PG-13	Fantasy	Original Screenplay	USA	3235	75.000	42.285	07/21/06	Lady in the Water
	PG-13	Comedy	Sequel	USA	3216	75.000	55.849	03/04/05	Be Cool
3	PG-13	Adventure	Based on Book or Short Story	UK	4285	150.000	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	PG-13	Action	Sequel	USA	4362	150.000	309.404	05/25/07	Pirates of the Caribbean: At World's End
4	PG-13	Comedy	Original Screenplay	USA	1500	17.000	7.314	04/21/06	American Dreamz
	PG-13	Drama	Based on Book or Short Story	USA	1510	16.000	16.358	10/11/02	White Oleander
5	PG-13	Drama	Based on Real Life Events	USA	2957	65.000	70.279	08/09/06	World Trade Center
	R	Action	Remake	USA	2979	60.000	77.907	04/23/04	Man on Fire

Table 17: Nearest Neighbor Pairs Identified with Numerical Indicators (Column names with bold and italic fonts).

19 also shows the low-grossing performance is poor, and thus the k-NN models are more suitable for high-grossing movies.

Table 17 shows some identified nearest neighbor pairs with this model. Pair 1 is a strange pair, which is a perfectly matched pair but their grosses differ a lot. We predict the gross of *In Her Line of Fire* would be $9 million but actually it is only $884. Actually the distance of the two movies in this pair is 0 because their budgets and number of opening screens are exactly the same. But generally speaking, the prediction based on this method is not bad. The other four pairs give pretty good predicted gross values.

2. K-NN Models based on Numerical and Categorical Indicators ($k\text{NN}_{adv1}$ and $k\text{NN}_{adv2}$)

   If we add the categorical indicators into previous model, the distance formula can be got using the regression method:

   $dis = 0.200 * d_b + 1.674 * d_o - 2.767 * d_d + 133.916 * d_m - 138.963 * d_c + 46.761 * d_s + 52.221 * d_g$

   However, the formula is problematic because the coefficients for dates and countries are negative values, which means they are not good predictors. After removing the two variables, the below formula is obtained:

   $dis = 0.205 * d_b + 1.666 * d_o + 129.756 * d_m + 51.275 * d_s + 53.936 * d_g$

   This formula is interesting because the coefficients of MPAA ratings, sources (sequel), and genres are big positive numbers. Therefore, the difference of those categorical data will make the distance of two movies very big, and it is not hard to conclude that most likely the nearest neighbor pairs should have the same MPAA ratings, sequel information, and primary genres. In table 16, model $k\text{NN}_{adv1}$ use the 7-dimensional distance measure and model $k\text{NN}_{adv2}$ use the 5-dimensional distance measure. The result also shows the latter model makes more sense.

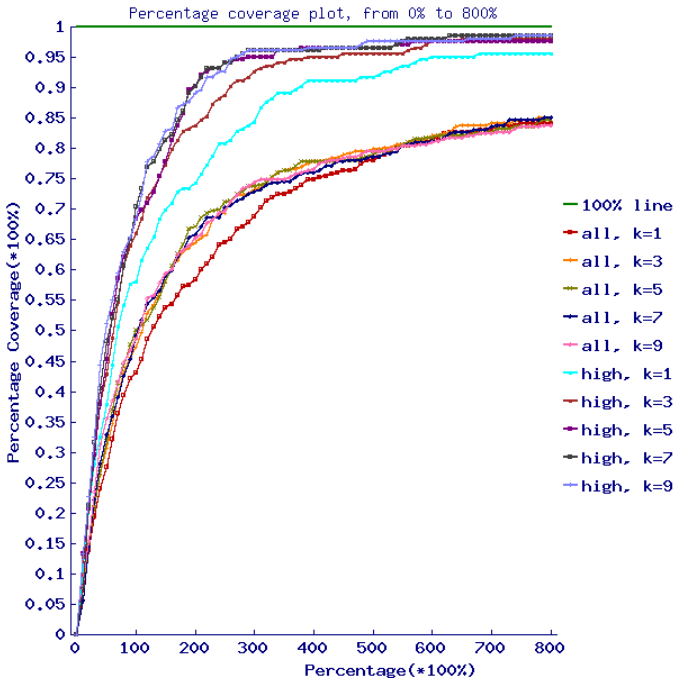

Figure 19: The Impact of k for model $k\text{NN}_{base}$. High-grossing performance is much higher than overall performance. For both overall performance and high-grossing performance, the model works worst when $k = 1$, while other k values work almost equally good, but with slightly performance gain with the increasing of k.

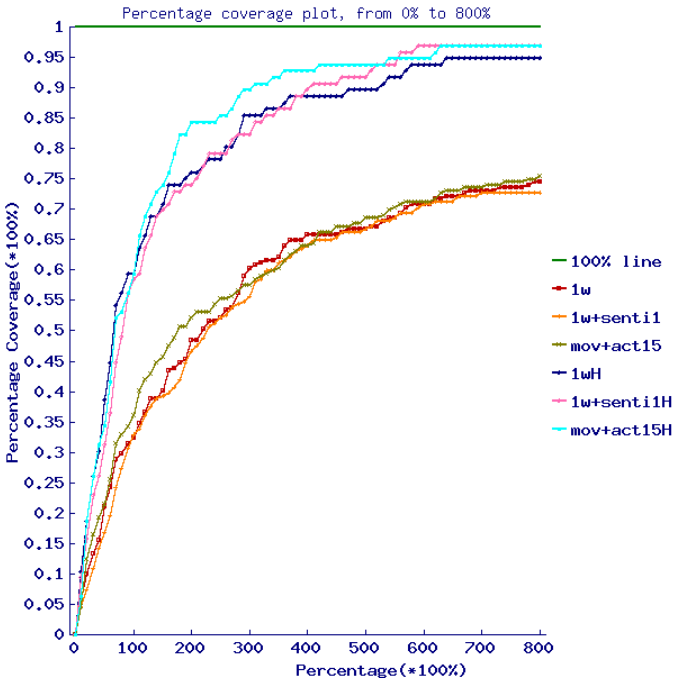

Figure 20: k-NN Models with News Data. The high-grossing performance is much better than overall performance. For high-grossing movies, model $k\text{NN}_{mov+act15}$ works better than others. Sentiment data show little but indeed some contribution to performance.

Table 18 shows some nearest neighbor pairs with models based on both numerical and categorical indicators. Compared with table 17, the target movies are the same, but the nearest neighbors are changed. We can notice that all the pairs have the same MPAA ratings, sequel information and genres. One interesting pair is pair 3' - the target movie is a "Harry Potter" movie and its nearest neighbor is another "Harry Potter" movie, which indicates a very good comparison.

- The impact of training set size:
  Undoubtedly, the bigger the training set is, the better the predicted gross is because we can find the smaller distance from the target movie. For example, if we add two more year data in the training set, a better match should be identified for the same movies. The model with a bigger training set is called $k\text{NN}'_{adv2}$. From table 16, we can see $k\text{NN}'_{adv2}(k = 3)$ works sightly better than $k\text{NN}_{adv2}(k = 3)$.

3. K-NN Models without Budget Information ($k\text{NN}_{nobudget}$)

   Sometimes the budget information is not given, so here we build another model based on $k\text{NN}_{adv2}$ model series, but without using budget indicators. Some identified nearest neighbor pairs are shown in table 19. The first pair shows the nearest neighbor pairs are different from that in table 18 because the budget information

No.	***MPAA***	***Genre***	***Source***	Cout.	***Scrns***	***Bgt($M)***	**Gro($M)**	Date	Name
1'	R	Action	Original Screenplay	USA	2	1.000	0.000884	04/21/06	In Her Line of Fire
	R	Action	Original Screenplay	USA	6	1.000	0.004	03/18/05	The Helix... Loaded
2'	PG-13	Fantasy	Original Screenplay	USA	3235	75.000	42.285	07/21/06	Lady in the Water
	PG-13	Fantasy	Original Screenplay	USA	2910	80.000	39.443	11/10/00	Little Nicky
3'	PG-13	Adventure	Based on Book or Short Story	UK	4285	150.000	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	PG-13	Adventure	Based on Book or Short Story	UK	3858	150.000	290.013	11/18/05	Harry Potter and the Goblet of Fire
4'	PG-13	Comedy	Original Screenplay	USA	1500	17.000	7.314	04/21/06	American Dreamz
	PG-13	Comedy	Original Screenplay	USA	1508	25.000	4.009	04/22/05	King's Ransom
5'	PG-13	Drama	Based on Real Life Events	USA	2957	65.000	70.279	08/09/06	World Trade Center
	PG-13	Drama	Based on Book or Short Story	USA	3018	40.000	47.958	03/12/04	Secret Window

Table 18: Nearest Neighbor Pairs Identified with Numerical and Categorical Indicators (Column names with bold and italic fonts). Please note the results are different from table 17.

No.	***MPAA***	***Genre***	***Source***	Cout.	***Scrns***	Bgt($M)	**Gro($M)**	Date	Name
5"	PG-13	Drama	Based on Real Life Events	USA	2957	65.000	70.279	08/09/06	World Trade Center
	PG-13	Drama	Based on Magazine Article	USA	3002	30.000	40.118	08/16/02	Blue Crush
6	R	Comedy	N/A	USA	7	0.015	0.195	06/02/06	The Puffy Chair
	R	Comedy	Original Screenplay	USA	7	1.000	1.745	09/29/00	The Broken Hearts Club: A Romantic Comedy

Table 19: Nearest Neighbor Pairs Identified by Models without using Budget Information.

Model	Type	Dim	Description
Reg_{base}	Regression	8	Basic Regression Model. 8 Indicators: Holiday, MPAA Rating(G, PG, PG-13, R), Foreign, Opening Screens, Sequel.
Reg_{budget}	Regression	9	Basic Budget Regression Model. 9 Indicators: Holiday, MPAA Rating(G, PG, PG-13, R), Foreign, Opening Screens, Sequel, Budget.
Reg_{genre}	Regression	14	Same with Reg_{base}, but add genre indicator(Action, Adventure, Animation, Biography, Documentary, Drama). Primary genre is used.
Reg_{adv1}	Regression	15	Advanced Regression Model (1). 15 Indicators: Holiday, MPAA Rating(G, PG, PG-13, R), Foreign, Opening Screens, Sequel, Budget, Genres(Action, Adventure, Animation, Biography, Documentary, Drama). Primary genre is used.
Reg_{adv2}	Regression	18	Advanced Regression Model (2). 18 Indicators: Holiday, MPAA Rating(G, PG, PG-13, R), Foreign, Opening Screens, Sequel, Budget, Genres(Action, Adventure, Animation, Biography, Documentary, Drama, Family, Fantasy, Sci-Fi). Primary and subsidiary genres are used.
Piece_{adv1}	Piecewise	15	Use method Reg_{adv1} to build separate models for low and high gross movies.
$k\text{NN}_{base}$	K-NN	2	K-Nearest Neighbor model using budgets and opening screens, use regression method to calculate distance.
$k\text{NN}_{base}$'	K-NN	2	K-Nearest Neighbor model using budgets and opening screens, use normalization method to calculate distance.
$k\text{NN}_{nobudget}$	K-NN	4	K-Nearest Neighbor model using opening screen, MPAA rating, source, and genre information.
$k\text{NN}_{adv1}$	K-NN	7	K-Nearest Neighbor model using budget, opening screen, release date, MPAA rating, source, origin country, and genre information.
$k\text{NN}_{adv2}$	K-NN	5	K-Nearest Neighbor model using budget, opening screen, MPAA rating, source, and genre information.

Table 20: Summary of Regression, Piecewise, and K-NN Models using traditional movie variables. "Dim" is the dimensions of models.

is not considered. The second pair shows the two movies have the same number of opening screens but quite different budget values, we still think they are the closest pairs. The performance data of models in this category are shown in Table 16 for $k = 1$ and $k = 7$.

4.6.4 Summary

To sum up, we can use regression models, piecewise models and K-nearest neighbor models to do movie gross prediction. For all the models, the high-grossing performance is higher than the low-gross performance. Piecewise models work better than regression models. If we compare Tables 15 and 16, we can see the overall performance of K-NN models is worse, but the high-grossing performance is better than that of regression models. If we use regression models for low-grossing movies and k-NN models for high-grossing movies, a better prediction will be expected. The performance of K-NN models strongly depended on the training set size. With more training data, and the increasing of k (but yet still a small number), the performance of K-NN models will be increasing. Table 20 shows the summary for all the models we designed so far.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
Reg_{base}	All	8.13	22.66	1.58	243.95	0	0.04	-0.01	92.69
Reg_{genre}	All	7.83	20.13	1.62	154.5	0.01	0.05	-0.02	92.8
Reg_{adv1}	All	**3.53**	6.31	1.24	38.73	0.03	0.19	0.22	**96.47**
Reg_{base}	High	9.75	31.36	0.88	243.95	0	-0.95	-0.58	92.1
Reg_{genre}	High	8.97	26.87	0.9	154.5	0.01	-0.92	-0.51	92.41
Reg_{adv1}	High	**2.03**	3.9	1.01	26.68	0.03	-0.55	-0.55	**97.97**

Table 21: Performance of Linear Regression Models for News Movie Sets.

4.7 Prediction with Traditional Movie Variables Using News Movie Set

Diverse models are built and compared in the previous section. Some high performing models are identified. In this section, we will re-run these models with news movie set, i.e., the movie sets with both IMBD database coverage and media coverage, either $\text{Movie}_{Budget-}$ or $\text{Movie}_{Budget+}$. The performance data will be compared with those of news variable based models in later sections.

4.7.1 Regression Models

Three models Reg_{base}, Reg_{genre} and Reg_{adv1} are selected to be examined: Reg_{base} is the base regression model, Reg_{genre} is the best regression model for movie set that does not contain budget information ($\text{Movie}_{Budget-}$), and Reg_{adv1} is the best regression model for movie set that contains budget information ($\text{Movie}_{Budget+}$), which is the best model among all regression models. Model Reg_{base} and Reg_{genre} are examined against movie set $\text{Movie}_{Budget-}$, while model Reg_{adv1} are examined against movie set $\text{Movie}_{Budget+}$. The result is shown at Table 21.

We can see the performance for high-grossing movies is still better than the overall performance. One reason is that the prediction for high-grossing movies is less likely to get high APD value because the actual gross is at least \$15 million. Another interesting result is that both the overall performance and high-gross performance are worse than those shown in Table 15. The main reason is that the movie set is much smaller, which weakens the genre influence for movie grosses.

4.7.2 Piecewise Linear Models

As we mentioned before, low-grossing and high-grossing movies may follow difference rules in terms of their grosses. Therefore we will be able to model them better if they are processed separately. Only high-grossing movies are consider here because people and the movie industry pay more attention to high-grossing movies.

Table 22 shows the performance of piecewise linear regression models for high-grossing movies. Compared to performance of high-grossing movies in Table 21, the improvements are significant.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
Reg_{base}	High	0.98	0.99	0.63	4.51	0.01	-0.2	-0.13	99.02
Reg_{genre}	High	0.97	1.1	0.64	5.48	0.02	-0.16	-0.11	99.03
Reg_{adv1}	High	0.97	0.9	0.65	4.2	0.01	-0.18	-0.32	99.03

Table 22: Performance of Piecewise Linear Models for News Movie Sets.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
$k\text{NN}_{nobudget}(k=1)$	All	18.18	84.91	1.33	895.39	0	0.2	0.14	92.03
$k\text{NN}_{nobudget}(k=7)$	All	18.66	83.5	1.52	875.16	0	0.87	0.5	89.9
$k\text{NN}_{base}(k=1)$	All	30.89	146.79	1.08	1016.7	0	-0.09	-0.04	92.04
$k\text{NN}_{base}(k=7)$	All	9.23	30.09	0.94	231.28	0.01	0.54	0.18	92.3
$k\text{NN}_{adv2}(k=1)$	All	18.68	99.68	0.99	873.52	0.01	0.13	0.17	92.96
$k\text{NN}_{adv2}(k=7)$	All	11.68	43.01	1.04	326.66	0.02	0.61	0.16	92.11
$k\text{NN}_{nobudget}(k=1)$	High	17.07	101.34	0.79	895.39	0	-0.55	-0.33	95.03
$k\text{NN}_{nobudget}(k=7)$	High	**2.44**	8.82	0.54	83.62	0.01	-0.41	-0.28	**97.56**
$k\text{NN}_{base}(k=1)$	High	36.03	182.08	0.64	1016.7	0.01	-0.62	-0.42	95.13
$k\text{NN}_{base}(k=7)$	High	1.94	8.85	0.44	66.58	0.01	-0.32	-0.24	98.06
$k\text{NN}_{adv2}(k=1)$	High	10.96	57.09	0.73	420.96	0.01	-0.5	-0.2	94.78
$k\text{NN}_{adv2}(k=7)$	High	**1.16**	1.78	0.58	9.55	0.02	-0.3	-0.26	**98.84**

Table 23: Performance of k-NN models for News Movie Sets. Compared to regression model shows in table 21, the high-grossing performance is much better.

4.7.3 K-Nearest Neighbor Models

$k\text{NN}_{base}$, $k\text{NN}_{nobudget}$, and $k\text{NN}_{adv2}$ are used for news movie sets because $k\text{NN}_{base}$ is the base model, $k\text{NN}_{nobudget}$ is the best k-NN model for movie set that does not contain budget information ($\text{Movie}_{Budget-}$), and $k\text{NN}_{adv2}$ is the best k-NN model for movie set that contains budget information ($\text{Movie}_{Budget+}$). In the performance table 23, only the performance data while $k=1$ and $k=7$ are listed as comparison.

We can compare the best k-NN model with the best regression model, i.e., $k\text{NN}_{nobudget}(k=7)$ vs. Reg_{genre}, and $k\text{NN}_{adv2}(k=7)$ vs. Reg_{adv1}. We find the overall performance of k-NN models is not better than that of regression models. But for the high-grossing part, k-NN models are much better. Especially, we can say k-NN models are much more accurate than linear regression models because they give more accurate predictions. For example, the APD mean of model Reg_{genre} is 8.97 for high-grossing movies, while that of model $k\text{NN}_{nobudget}(k=7)$ is only 2.44, which indicates a huge improvement.

Tables 24 shows some examples of some identified nearest neighbor pairs. The table will be used later for our comparison.

4.8 Prediction with News Variables

In this section, we are trying to predict movie grosses with merely news references, i.e., no traditional IMDB movie variables are involved. There are three kinds of news data that can be used in our prediction models:

- News reference counts: Article counts for movie entities (movie titles, directors, top 3 actors or top 15 actors) during the 1-week, 1-month, or 4-month time period before a movie's release.

No.	*MPAA*	*Genre*	*Source*	Cout.	*Scrns*	*Bgt($M)*	**Gro($M)**	Date	Name
1	PG-13	Action	Sequel	USA	3434	102.000	167.365	11/17/06	Casino Royale
	PG-13	Action	Original Screenplay	USA	3495	138.000	32.117	07/29/05	Stealth
2	PG	Action	Based on Book or Short Story	USA	3685	110.000	250.863	12/22/06	Night at the Museum
	PG	Action	Based on Book or Short Story	USA	3020	100.000	75.030	12/15/06	Eragon
3	PG-13	Adventure	Based on Book or Short Story	UK	4285	150.000	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	PG-13	Adventure	Based on Book or Short Story	UK	3858	150.000	290.013	11/18/05/	Harry Potter and the Goblet of Fire
4	R	Comedy	Original Screenplay	USA	7	3.500	0.221	09/14/07	Ira and Abby
	R	Comedy	Original Screenplay	USA	7	3.500	0.107	02/17/06	Winter Passing
5	R	Crime	Sequel	USA	3183	10.000	63.300	10/26/07	Saw IV
	R	Crime	Sequel	USA	3167	10.000	80.239	10/27/06	Saw III

Table 24: Nearest Neighbor Pairs Identified with Traditional Movie Variables.

- News sentiment counts: News positive or negative sentiment counts for movie entities (movie titles, directors, top 3 actors or top 15 actors) during 1-week, 1-month, or 4-month time period before a movie's release.
- Other news sentiment statistic data, including polarity, subjectivity, positive references per reference, negative references per reference, and positive-negative differences per reference.

Now, we use these news indicators to set up regression models, piecewise linear models, and k-NN models in the following subsections.

4.8.1 Regression Models

The selection for news indicators is based on our previous correlation analysis. We build three kinds of regression models.

1. Use news reference counts only (Model nReg_{1w}, nReg_{mov}, $\text{nReg}_{mov+act15}$): Model nReg_{1w} takes three indicators, the pre-release 1-week news article counts in terms of movie titles, top 3 actors, and top 15 actors. Model nReg_{mov} takes the pre-release 1-week, 1-month, and 4-month news article counts in terms of movie titles as indicators. By contrast, $\text{nReg}_{mov+act15}$ takes 6 indicators, the pre-release 1-week, 1-month and 4-month news article counts in terms of movie titles and top 15 actors. From the performance table 25, we can see models nReg_{1w} and $\text{nReg}_{mov+act15}$ perform better than nReg_{mov}. The reason is that the 1-month and 4-month data have less correlation with grosses than 1-week data.
2. Use news reference counts plus sentiment counts (Model $\text{nReg}_{mov+senti1}$ and $\text{nReg}_{1w+senti1}$): $\text{nReg}_{mov+senti1}$ is based on model nReg_{mov}, but adding 2 sentiment indicators, the pre-release 1-week positive and negative news references in terms of movie titles. While $\text{nReg}_{1w+senti1}$ add 6 sentiment indicators, the pre-release 1-week positive and negative news references in terms of movie titles, top

3 actors, and top 15 actors. However, both the overall performance and the high-grossing movie performance are quite similar with the models without adding these sentiment indicators. The result is understandable because the sentiment counts are highly correlated with the news article counts and we can find they are less relevant to movie grosses than news article counts if we compare table table 9 to 7, which makes the sentiment counts carry little extra valuable information.

3. Use news reference counts plus other sentiment statistics data (Model $\text{nReg}_{mov+senti2}$ and $\text{nReg}_{1w+senti2}$): As we know, the raw sentiment counts is not much valuable as extra indicators. However, we can get some valuable information with using the sentiment statistic data computed from raw sentiment counts, like polarity, subjectivity, and per-reference sentiment data. According to table 12 and 13, for $\text{Movie}_{Budget-}$, we will add the the pre-release 1-week polarity, subjectivity, and negative references per reference in terms of movie titles. While for $\text{Movie}_{Budget+}$, except above 3 sentiment statistics, we will also add the pre-release 1-month polarity, subjectivity, and negative references per reference in terms of top 15 actors. From the performance table, there are still no significant performance gains with adding these sentiment statistics. Actually we can see from table 25, the sentiment data improves the performance a little bit in some cases, while they make the performance worse in other cases. There are two possible reasons, one is that these sentiment statistics are not strongly correlated with grosses, another reason is that the sentiment data output from *Lydia* need to be improved.

4.8.2 Piecewise Linear Models

Again, we only consider the high-grossing movies for piecewise linear models. Because the sentiment data do not show promising results for forecasting, here only the piecewise models for nReg_{1w}, nReg_{mov}, and $\text{nReg}_{mov+act15}$ are considered. Table 26 shows they have better performance than classic linear regression models. Moreover, $\text{nReg}_{mov+act15}$ works the best among the three piecewise models.

4.8.3 K-Nearest Neighbor Models

k-NN models use the same indicators as regression models. Models using merely news references and news references plus sentiment data are examined respectively. Specifically, the performance data of models $k\text{NN}_{1w}$, $k\text{NN}_{1w+senti1}$, and $k\text{NN}_{mov+act15}$ are listed in table 27.

If we compare table 27 and table 25, the k-NN models have worse overall performance but better high-grossing performance. One surprising result is that the sentiment data in the k-NN models shows some predictive power. Table 27 shows model $k\text{NN}_{1w+senti1}$ has much better performance than model $k\text{NN}_{1w}$. In fact, table 27 shows the sentiment data makes slight but yet quite stable performance improvements. The basic reason is that the sentiment data will be helpful in identifying more similar movies. For example, two candidate movies may have the same amount of news exposure with the target movies, but the candidate with closer sentiment counts win. Figure 20 shows

Model	Set	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
nReg_{1w}	$\text{Movie}_{Budget-}$	All	8.72	22.55	2.04	184.74	0	0.31	0	92.1
nReg_{mov}	$\text{Movie}_{Budget-}$	All	10.24	34.91	2.18	341.46	0.01	0.24	0.08	92.07
$\text{nReg}_{mov+senti1}$	$\text{Movie}_{Budget-}$	All	12	61.98	2.01	835.62	0.01	0.21	0.14	92.64
$\text{nReg}_{1w+senti1}$	$\text{Movie}_{Budget-}$	All	8.1	20.58	1.7	158.79	0.01	0.32	0.07	92.63
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget-}$	All	10.46	29.81	1.83	253.71	0.01	0.55	0.33	91.5
$\text{nReg}_{mov+senti2}$	$\text{Movie}_{Budget-}$	All	10.32	35.38	2.01	362.84	0.01	0.24	0.18	92.02
$\text{nReg}_{1w+senti2}$	$\text{Movie}_{Budget-}$	All	8.72	22.15	2.05	168.05	0	0.31	0.05	92.07
nReg_{1w}	$\text{Movie}_{Budget+}$	All	6.29	15.58	1.59	134.02	0.01	0.24	0.07	94.07
nReg_{mov}	$\text{Movie}_{Budget+}$	All	**16.68**	109.29	1.72	1054.45	0.04	0.02	-0.01	**93.72**
$\text{nReg}_{mov+senti1}$	$\text{Movie}_{Budget+}$	All	13.93	81.18	1.59	774.77	0.04	0.01	0.01	93.68
$\text{nReg}_{1w+senti1}$	$\text{Movie}_{Budget+}$	All	6.69	16.18	1.66	136.64	0	0.23	0.07	93.7
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget+}$	All	6.49	14.5	1.72	109.56	0.01	0.46	0.25	93.61
$\text{nReg}_{mov+senti2}$	$\text{Movie}_{Budget+}$	All	**169.77**	1479.6	1.61	14309	0.02	-0.07	-0.08	**93.54**
$\text{nReg}_{1w+senti2}$	$\text{Movie}_{Budget+}$	All	7.85	25.78	1.44	239.6	0.01	0.12	-0.05	93.63
nReg_{1w}	$\text{Movie}_{Budget-}$	High	**4.02**	12.86	1.11	121.42	0	-0.67	-0.58	**96.2**
nReg_{mov}	$\text{Movie}_{Budget-}$	High	6.94	35.28	1.5	341.46	0.01	-0.78	-0.77	95.57
$\text{nReg}_{mov+senti1}$	$\text{Movie}_{Budget-}$	High	12.21	85.24	1.36	835.62	0.03	-0.8	-0.76	95.45
$\text{nReg}_{1w+senti1}$	$\text{Movie}_{Budget-}$	High	4.45	16.52	1.08	158.79	0.01	-0.67	-0.54	96.16
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget-}$	High	2.87	5.4	0.95	31.39	0.01	-0.36	-0.23	97.13
$\text{nReg}_{mov+senti2}$	$\text{Movie}_{Budget-}$	High	7.32	37.63	1.4	362.84	0.01	-0.78	-0.76	95.41
$\text{nReg}_{1w+senti2}$	$\text{Movie}_{Budget-}$	High	**3.89**	11.39	1.25	105.72	0	-0.66	-0.54	**96.17**
nReg_{1w}	$\text{Movie}_{Budget+}$	High	2.81	5.8	0.86	39.38	0.03	-0.57	-0.48	97.19
nReg_{mov}	$\text{Movie}_{Budget+}$	High	21.22	140.64	1.3	1054.45	0.04	-0.74	-0.77	95.82
$\text{nReg}_{mov+senti1}$	$\text{Movie}_{Budget+}$	High	16.21	103.29	1.23	774.77	0.04	-0.73	-0.67	95.84
$\text{nReg}_{1w+senti1}$	$\text{Movie}_{Budget+}$	High	2.87	5.18	1.16	31.96	0	-0.6	-0.56	97.13
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget+}$	High	2.68	6.71	1.03	47.56	0.01	-0.34	-0.34	97.32
$\text{nReg}_{mov+senti2}$	$\text{Movie}_{Budget+}$	High	258.04	1911.9	1.41	14309	0.02	-0.82	-0.75	95.7
$\text{nReg}_{1w+senti2}$	$\text{Movie}_{Budget+}$	High	3.23	4.99	1.21	23.48	0.01	-0.64	-0.63	96.77

Table 25: Performance of Linear Regression Models Using News Variables. The performance of nReg_{1w} and $\text{nReg}_{mov+act15}$ is higher than that of nReg_{mov}. To add sentiment data does not make significant improvement in terms of performance.

Model	Set	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
nReg_{1w}	$\text{Movie}_{Budget-}$	High	0.99	0.93	0.73	4.68	0.03	-0.06	-0.09	99.01
nReg_{mov}	$\text{Movie}_{Budget-}$	High	1.09	1.32	0.7	8.51	0	-0.16	-0.03	98.91
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget-}$	High	0.99	0.9	0.77	4.02	0.01	-0.05	-0.07	99.01
nReg_{1w}	$\text{Movie}_{Budget+}$	High	1.34	1.04	0.95	4.5	0	-0.22	-0.32	98.87
nReg_{mov}	$\text{Movie}_{Budget+}$	High	1.45	1.75	0.96	10.45	0.02	-0.32	-0.34	98.55
$\text{nReg}_{mov+act15}$	$\text{Movie}_{Budget+}$	High	1.12	1.03	0.76	4.23	0	-0.19	-0.25	98.88

Table 26: Performance of Piecewise Linear Models Using News References. The performance of nReg_{1w} and $\text{nReg}_{mov+act15}$ is higher than that of nReg_{mov}.

Model	Set	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
$k\text{NN}_{1w}(k=1)$	$\text{Movie}_{Budget-}$	All	**40.88**	129.26	2.36	965.76	0.01	0.1	0.14	**82.91**
$k\text{NN}_{1w}(k=7)$	$\text{Movie}_{Budget-}$	All	24.22	86.93	2.18	916.06	0	1.06	0.8	87.25
$k\text{NN}_{1w+senti1}(k=1)$	$\text{Movie}_{Budget-}$	All	**34.4**	149.91	2.4	1569.22	0.03	0.1	0.07	**86.27**
$k\text{NN}_{1w+senti1}(k=7)$	$\text{Movie}_{Budget-}$	All	25	91.87	2.29	965.07	0.02	1.02	0.72	87.51
$k\text{NN}_{mov+act15}(k=1)$	$\text{Movie}_{Budget-}$	All	31.25	105.5	2.13	837.63	0	0.22	0.25	85.76
$k\text{NN}_{mov+act15}(k=7)$	$\text{Movie}_{Budget-}$	All	21.6	74	1.78	727.71	0.01	0.99	0.53	87.57
$k\text{NN}_{1w}(k=1)$	$\text{Movie}_{Budget+}$	All	**34.97**	103.46	1.82	640.67	0.05	-0.49	-0.24	**84.77**
$k\text{NN}_{1w}(k=7)$	$\text{Movie}_{Budget+}$	All	8.38	21.24	1.51	145.04	0	0.51	0.31	92.29
$k\text{NN}_{1w+senti1}(k=1)$	$\text{Movie}_{Budget+}$	All	**23.94**	94.28	1.78	645.56	0.02	-0.01	-0.12	**89.27**
$k\text{NN}_{1w+senti1}(k=7)$	$\text{Movie}_{Budget+}$	All	15.87	52.41	1.99	405.78	0.02	0.79	0.42	89.53
$k\text{NN}_{mov+act15}(k=1)$	$\text{Movie}_{Budget+}$	All	22.42	90.35	2.02	626.35	0	0.27	0.24	88.77
$k\text{NN}_{mov+act15}(k=7)$	$\text{Movie}_{Budget+}$	All	17.95	58.6	1.66	448.73	0.01	0.93	0.46	88.89
$k\text{NN}_{1w}(k=1)$	$\text{Movie}_{Budget-}$	High	**22.34**	70.6	1.41	459.59	0.01	-0.85	-0.4	**87.91**
$k\text{NN}_{1w}(k=7)$	$\text{Movie}_{Budget-}$	High	1.79	2.99	0.67	17	0	-0.25	-0.15	98.21
$k\text{NN}_{1w+senti1}(k=1)$	$\text{Movie}_{Budget-}$	High	**30.32**	168.05	1.3	1569.22	0.03	-0.78	-0.37	**90.78**
$k\text{NN}_{1w+senti1}(k=7)$	$\text{Movie}_{Budget-}$	High	1.63	2.17	0.82	12.87	0.03	-0.32	-0.26	98.37
$k\text{NN}_{mov+act15}(k=1)$	$\text{Movie}_{Budget-}$	High	28.17	121.08	0.89	837.63	0	-0.76	-0.23	90.42
$k\text{NN}_{mov+act15}(k=7)$	$\text{Movie}_{Budget-}$	High	2.2	9.26	0.67	90.47	0.01	-0.23	-0.19	97.8
$k\text{NN}_{1w}(k=1)$	$\text{Movie}_{Budget+}$	High	**31.41**	87.64	1.41	459.59	0.11	-1.21	-0.54	**84.37**
$k\text{NN}_{1w}(k=7)$	$\text{Movie}_{Budget+}$	High	1.98	3.09	0.87	17	0	-0.41	-0.33	98.02
$k\text{NN}_{1w+senti1}(k=1)$	$\text{Movie}_{Budget+}$	High	**17.63**	83.17	0.95	599.27	0.02	-0.7	-0.45	**92.67**
$k\text{NN}_{1w+senti1}(k=7)$	$\text{Movie}_{Budget+}$	High	1.32	1.45	0.93	6.71	0.02	-0.3	-0.36	98.68
$k\text{NN}_{mov+act15}(k=1)$	$\text{Movie}_{Budget+}$	High	16.47	81.54	0.97	599.27	0.01	-0.61	-0.21	92.49
$k\text{NN}_{mov+act15}(k=7)$	$\text{Movie}_{Budget+}$	High	1.14	1.17	0.78	5.94	0.01	-0.24	-0.36	98.86

Table 27: Performance of k-Nearest Neighbor Models Using News Variables. Model $k\text{NN}_{mov+act15}(k=7)$ performs better than $k\text{NN}_{1w}(k=7)$. To add sentiment data into model $k\text{NN}_{1w}(k=7)$ makes slight but yet quite stable performance gain.

the comparison of different k-NN models with news data, in which it indicates model $k\text{NN}_{mov+act15}$ works better then the others, especially for high-grossing movies.

Table 28 lists some nearest neighbor pairs. Compared to table 24, most of them are different, but pair 5' is the same with that in table 28.

4.8.4 Summary

In this section, we prove that movie grosses are predictable with merely news data, and we build different models to do the prediction. Some models using news data achieve similar performance to those using traditional IMDB data, especially for high gross movies because of their high media exposure. Compared to regression models, the k-NN models have worse overall performance but better performance for high-grossing movies. The sentiment data do not show much predictive power with regression models, but it is capable to improve the performance with k-NN models.

Table 29 shows all the models built with news data.

Figure 21 shows the performance comparison of models using traditional movie data and models using news data.

No.	*Movie Title Refs*			*Top 15 Actors Refs*			**Gross($M)**	Date	Name
	1 week	*1 month*	*4 month*	*1 week*	*1 month*	*4 month*			
1'	588	320	980	2257	2373	5846	167.365	11/17/06	Casino Royale
	607	369	945	2631	1925	7126	244.083	06/09/06	Cars
2'	116	75	71	1102	2076	10325	250.863	12/22/06	Night at the Museum
	97	66	26	1157	1911	6439	8.536	09/09/05	An Unfinished Life
3'	1440	1236	1203	2855	2558	3420	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	1078	952	1198	4065	3113	9332	200.120	06/28/06	Superman Returns
4'	6	1	0	151	981	1116	0.221	09/14/07	Ira and Abby
	9	16	11	204	854	1660	0.049	08/05/05	The Chumscrubber
5'	106	17	244	303	248	911	63.300	10/26/07	Saw IV
	62	39	164	336	73	231	80.239	10/27/06	Saw III

Table 28: Nearest Neighbor Pairs Identified with News Data. Compared to table 24, most of the pairs are changed, but pair 5' is identical.

Model	Type	Dim	Description
nReg_{1w}	Regression	3	1 Week Regression Model. 3 Indicators: the pre-release 1-week news article counts in terms of movie titles, top 3 actors, and top 15 actors.
nReg_{mov}	Regression	3	Movie References Regression Model. 3 Indicators: the pre-release 1-week, 1-month, and 4-month news article counts in terms of movie titles.
$\text{nReg}_{mov+senti1}$	Regression	5	Movie Reference Regression Model, with Sentiment Data (1). 5 Indicators: the pre-release 1-week, 1-month, and 4-month news article counts in terms of movie titles; the pre-release 1-week positive and negative news references in terms of movie titles.
$\text{nReg}_{1w+senti1}$	Regression	9	1 Week Regression Model, with Sentiment Data (1). 9 Indicators: the pre-release 1-week news article counts in terms of movie titles, top 3 actors, and top 15 actors; the pre-release 1-week positive and negative news references in terms of movie titles, top 3 actors, and top 15 actors respectively.
$\text{nReg}_{mov+act15}$	Regression	6	Movie and Top 15 Actors Regression Model. 6 Indicators: the pre-release 1-week, 1-month and 4-month news article counts in terms of movie titles and top 15 actors.
$\text{nReg}_{mov+senti2}$	Regression	6/9	Movie Reference Regression Model, with Sentiment Data (2). 6 Indicators in terms of $\text{Movie}_{Budget-}$: the pre-release 1-week, 1-month, and 4-month news article counts for movie titles; the pre-release 1-week polarity, subjectivity, and negative references per reference in terms of movie titles. 3 More Indicators for $\text{Movie}_{Budget+}$: the pre-release 1-month polarity, subjectivity, and negative references per reference in terms of top 15 actors.
$\text{nReg}_{1w+senti2}$	Regression	6/9	1 Week Regression Model, with Sentiment Data (2). 6 Indicators for $\text{Movie}_{Budget-}$: the pre-release 1-week news article counts for movie titles, top 3 actors, and top 15 actors; the pre-release 1-week polarity, subjectivity, and negative references per reference in terms of movie titles. 3 More Indicators for $\text{Movie}_{Budget+}$: the pre-release 1-month polarity, subjectivity, and negative references per reference in terms of top 15 actors.
$k\text{NN}_{1w}$	K-NN	3	Use the same indicators with nReg_{1w}.
$k\text{NN}_{1w+senti1}$	K-NN	9	Use the same indicators with $\text{nReg}_{1w+senti1}$.
$k\text{NN}_{mov+act15}$	K-NN	6	Use the same indicators with $\text{nReg}_{mov+act15}$.

Table 29: Summary of Regression, Piecewise, and K-NN Models Using News Variables.

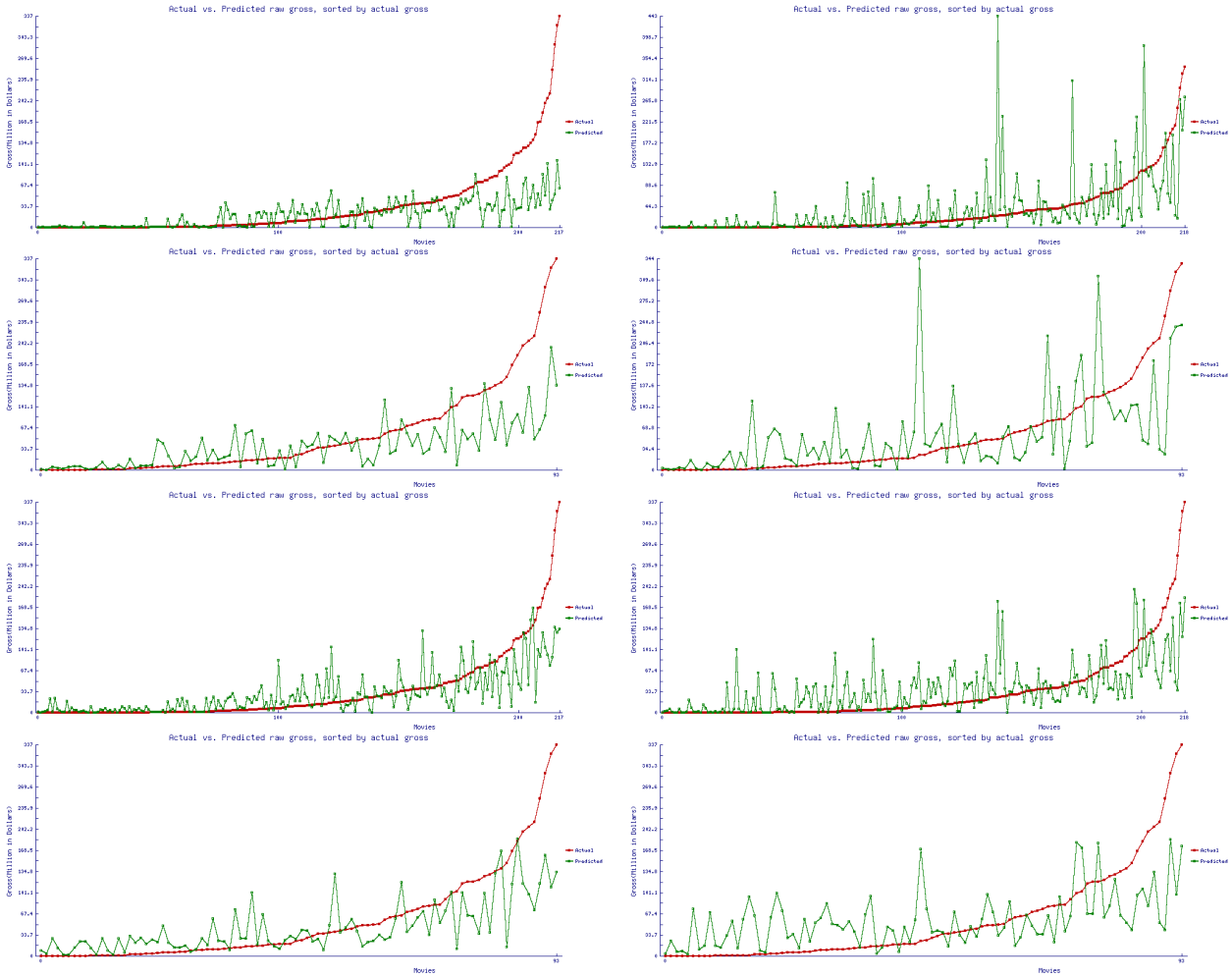

Figure 21: Comparison of models using traditional movie data (left hand side) and models using news data (right hand side). From the top to the bottom, they are plots of regression model for Movie$_{Budget-}$, regression model for Movie$_{Budget+}$, k-NN model for Movie$_{Budget-}$, and k-NN model for Movie$_{Budget+}$ respectively. We can see the regression model work better for low-grossing movies, but k-NN models work better for high-gross movies.

Model	Data Set	IMDB	News	IMDB + News
Regression	$\text{Movie}_{Budget-}$	Reg_{genre}	nReg_{1w}	Reg_{genre}+nReg_{1w}
	$\text{Movie}_{Budget+}$	Reg_{adv1}	$\text{nReg}_{mov+act15}$	Reg_{adv1}+$\text{nReg}_{mov+act15}$
k-NN	$\text{Movie}_{Budget-}$	$k\text{NN}_{nobudget}$	$k\text{NN}_{mov+act15}(k=7)$	$k\text{NN}_{nobudget}$+$k\text{NN}_{mov+act15}(k=7)$
	$\text{Movie}_{Budget+}$	$k\text{NN}_{adv2}$	$k\text{NN}_{mov+act15}(k=7)$	$k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k=7)$

Table 30: This table shows the best regression models and k-NN models we built using IMDB data, news data, or IMDB plus news data, for sets $\text{Movie}_{Budget-}$ and $\text{Movie}_{Budget+}$ repectively.

4.9 Prediction with Traditional Movie Variables plus News Variables

From the previous sections, we can see that decent models can be built with either traditional IMDB data or news data. If we go further, can we build even more successful models with the combination of IMDB data and news data? In this section, we will combine the best IMDB models and the best news models and try to generate even better prediction models.

Table 30 shows the best regression and k-NN models we get so far, for movie set $\text{Movie}_{Budget+}$ and $\text{Movie}_{Budget-}$ respectively. Intuitively, we use the combination of the best IMDB model and the best news model as the best combined model.

4.9.1 Regression Models

So far, Reg_{adv1} and Reg_{genre} are the best IMDB models for movie sets that have or have no available budget information respectively. nReg_{1w} and $\text{nReg}_{mov+act15}$ are the best regression models using news data. Table 31 shows the performance data for four combined models.

Compared to table 21, which shows the performance of IMDB regression models, the improvement is significant. For example, the overall and high-grossing APD mean are 3.79 and 2.4 respectively for Reg_{genre}+$\text{nReg}_{mov+act15}$, while they are as big as 7.83 and 8.97 for Reg_{genre}. Likewise, the combined models also work much better than pure news-data-based models (shown in table 25).

4.9.2 Piecewise Linear Models

Table 32 shows the high-grossing movie performance for piecewise models. There is also some performance gain from the previous IMDB or news models. However, the performance gain is relatively small because the piecewise models for high-gross movies were good enough before the news data are added.

4.9.3 K-Nearest Neighbor Models

We design the combined model according to table 30, the performance of which is shown in table 33. If we compare it with table 23 and 27, we can find both the overall performance and high-grossing performance are improved greatly. The combined models have much smaller APD value, which means the predicted gross is much

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
Reg_{genre}+nReg_{1w}	All	3.82	6.51	1.23	37.88	0.01	0.23	0.07	96.18
Reg_{genre}+$\text{nReg}_{mov+act15}$	All	3.79	6.3	1.26	39.79	0	0.25	0.11	96.21
Reg_{adv1}+nReg_{1w}	All	2.76	4.34	1.23	26.73	0	0.22	0.11	97.24
Reg_{adv1}+$\text{nReg}_{mov+act15}$	All	**2.63**	3.95	1.16	21.94	0.03	0.21	0.07	**97.37**
Reg_{genre}+nReg_{1w}	High	2.48	4.95	0.98	37.21	0.01	-0.55	-0.36	97.52
Reg_{genre}+$\text{nReg}_{mov+act15}$	High	2.4	4.49	1.04	31.63	0.01	-0.53	-0.43	97.6
Reg_{adv1}+nReg_{1w}	High	1.57	2.04	0.97	12.24	0	-0.4	-0.3	98.43
Reg_{adv1}+$\text{nReg}_{mov+act15}$	High	**1.54**	2.31	0.9	14.95	0.03	-0.41	-0.35	**98.46**

Table 31: Performance of Linear Regression Models Using Traditional Movie Variables plus News Variables.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
Reg_{genre}+nReg_{1w}	High	0.84	0.86	0.59	5.46	0.01	-0.08	-0.07	99.16
Reg_{genre}+$\text{nReg}_{mov+act15}$	High	0.87	0.9	0.6	5.73	0.06	-0.09	-0.12	99.13
Reg_{adv1}+nReg_{1w}	High	0.96	0.83	0.71	3.75	0.05	-0.17	-0.31	99.04
Reg_{adv1}+$\text{nReg}_{mov+act15}$	High	0.99	0.87	0.73	4.07	0.02	-0.19	-0.36	99.01

Table 32: Performance of Piecewise Linear Models Using Traditional Movie Variables plus News Variables.

closer to the actual gross. For example, the *APD* value for all movies of model $k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k = 7)$ is only 5.82. By contrast, they are 11.68 and 17.95 for model $k\text{NN}_{adv2}$ and $k\text{NN}_{mov+act15}(k = 7)$ respectively. Clearly, there is a big enhancement.

Table 34 shows some nearest neighbor pairs identified with IMDB and news data. With adding news controlling variables, some pairs are identical with those in IMDB models, some pairs are identical with those in news reference models, while others are different from those in previous models.

Model	Perf	APD Mean	APD SD	APD Med	APD Max	APD Min	LR Mean	LR Med	Score
$k\text{NN}_{base}$+$k\text{NN}_{1w}(k = 1)$	All	5.24	13.84	1.23	94.88	0.01	0.02	0.02	94.76
$k\text{NN}_{base}$+$k\text{NN}_{1w}(k = 7)$	All	3.28	8.27	1.07	72.82	0.02	0.4	0.19	96.72
$k\text{NN}_{nobudget}$+$k\text{NN}_{mov+act15}(k = 1)$	All	6.98	21.84	1.18	219.81	0.01	0.1	0.13	93.7
$k\text{NN}_{nobudget}$+$k\text{NN}_{mov+act15}(k = 7)$	All	**10.89**	36.11	1.75	261.41	0.01	0.83	0.56	**92.17**
$k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k = 1)$	All	4.4	10.44	1.09	78.59	0.01	0.31	0.19	95.6
$k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k = 7)$	All	**5.82**	20.44	1.16	188.75	0	0.6	1.35	**95.13**
$k\text{NN}_{base}$+$k\text{NN}_{1w}(k = 1)$	High	4.85	14.57	1.04	94.88	0.01	-0.57	-0.47	95.15
$k\text{NN}_{base}$+$k\text{NN}_{1w}(k = 7)$	High	1.22	2.23	0.68	14.33	0.02	-0.26	-0.25	98.78
$k\text{NN}_{nobudget}$+$k\text{NN}_{mov+act15}(k = 1)$	High	4	12.76	0.87	92.23	0.02	-0.45	1.18	96.0
$k\text{NN}_{nobudget}$+$k\text{NN}_{mov+act15}(k = 7)$	High	**1.16**	1.81	0.56	13.8	0.01	-0.24	-0.17	**98.84**
$k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k = 1)$	High	2.91	10.51	0.78	78.59	0.01	-0.3	-0.27	97.09
$k\text{NN}_{adv2}$+$k\text{NN}_{mov+act15}(k = 7)$	High	**1.01**	1.12	0.58	5.57	0.01	-0.21	-0.2	**98.99**

Table 33: Performance of k-Nearest Neighbor Models Using Traditional Movie Variables plus News Variables.

No.	*MPAA*	*Genre*	*Source*	Cout.	*Scrns*	*Bgt($M)*	**Gro($M)**	Date	Name
1”	PG-13	Action	Sequel	USA	3434	102.000	167.365	11/17/06	Casino Royale
	PG-13	Action	Based on Comic or Grph. Novel	USA	3858	150.000	250.344	06/15/05	Batman Begins
2”	PG	Action	Based on Book or Short Story	USA	3685	110.000	250.863	12/22/06	Night at the Museum
	PG	Action	Based on Book or Short Story	USA	3020	100.000	75.030	12/15/06	Eragon
3”	PG-13	Adventure	Based on Book or Short Story	UK	4285	150.000	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	PG-13	Adventure	Remake	New Zea.	3568	207.000	218.080	12/14/05	King Kong
4”	R	Comedy	Original Screenplay	USA	7	3.500	0.221	09/14/07	Ira and Abby
	R	Comedy	Original Screenplay	USA	28	6.800	0.049	08/05/05	The Chumscrubber
5”	R	Crime	Sequel	USA	3183	10.000	63.300	10/26/07	Saw IV
	R	Crime	Sequel	USA	3167	10.000	80.239	10/27/06	Saw III

No.	*Movie Title Refs*			*Top 15 Actors Refs*			**Gross($M)**	Date	Name
	1 week	*1 month*	*4 month*	*1 week*	*1 month*	*4 month*			
1”	588	320	980	2257	2373	5846	167.365	11/17/06	Casino Royale
	361	371	484	1434	1769	8480	250.344	06/15/05	Batman Begins
2”	116	75	71	1102	2076	10325	250.863	12/22/06	Night at the Museum
	258	176	233	1500	1929	4125	75.030	12/15/06	Eragon
3”	1440	1236	1203	2855	2558	3420	292.005	07/11/07	Harry Potter and the Order of the Phoenix
	3006	2502	2364	2926	2810	4074	218.080	12/14/05	King Kong
4”	6	1	0	151	981	1116	0.221	09/14/07	Ira and Abby
	9	16	11	204	854	1660	0.049	08/05/05	The Chumscrubber
5”	106	17	244	303	248	911	63.300	10/26/07	Saw IV
	62	39	164	336	73	231	80.239	10/27/06	Saw III

Table 34: Nearest Neighbor Pairs Identified with Traditional Movie Variables plus News Variables. If we compare this table to table 24 and 28, we can see pair 2” is the same with that in table 24, pair 4” is the same with that in table 28, pair 5” is the same with that in both table 24 and 28, but pairs 1 and 3 are different from all of them.

4.9.4 Summary

From the above experiments, we proved that we can set up pretty decent models with merely news data. However, we will be able to set up better models when we combine IMDB movie data and *Lydia* news data together. The performance of the models using the combined data is better than those use either IMDB data or news data alone, and they will give smaller APD value, higher scores, and better percentage coverage. In terms of news data, the most useful data are news article counts. Sentiment data output from *Lydia* show little predictive power in regression models, but show great predictive power in k-nearest neighbor models.

Figure 22 shows the comparison of models using only IMDB data and models using combined IMDB data and news data. All the plots intuitively show the combined models are better than IMDB models.

Finally, figures 23, 24, 25, and 26 show the percentage coverage comparison of IMDB models, news models, and the combined models. These plots show both overall performance and high-gross performance of combined models are higher than that of IMDB models and news models. One interesting fact is, the combined k-NN models are worse than the corresponding IMDB models in terms of 300% percentage coverage

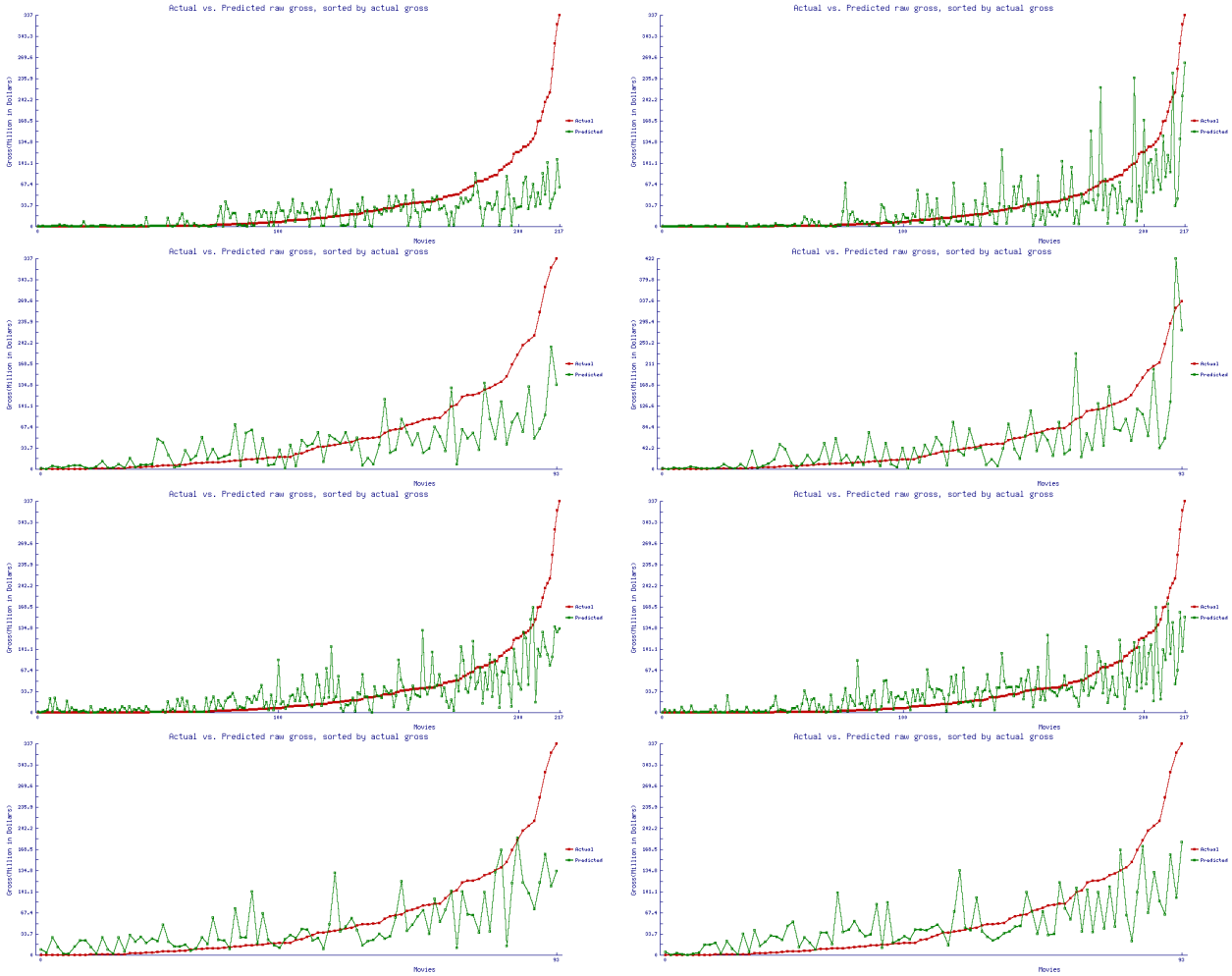

Figure 22: Comparison of models using IMDB data (left hand side) and models using both IMDB data and news data (right hand side). From the top to the bottom, they are plots of regression model for Movie$_{Budget-}$, regression model for Movie$_{Budget+}$, k-NN model for Movie$_{Budget-}$, and k-NN model for Movie$_{Budget+}$ respectively. All the plots use the best models, as we shown in table 30. Each pair uses the same prediction variables. We can see the regression model work better for low-grossing movies, but k-NN models work better for high-gross movies. Moreover, the combined models are superior to IMDB models in all the four cases. If we compare to figure 21, the combined models are better than news models as well.

or below, but they perform better after that. The reason is that the dimension of the combined models doubles, but the training set is not increased accordingly. The combined k-NN model will perform much better with more training data.

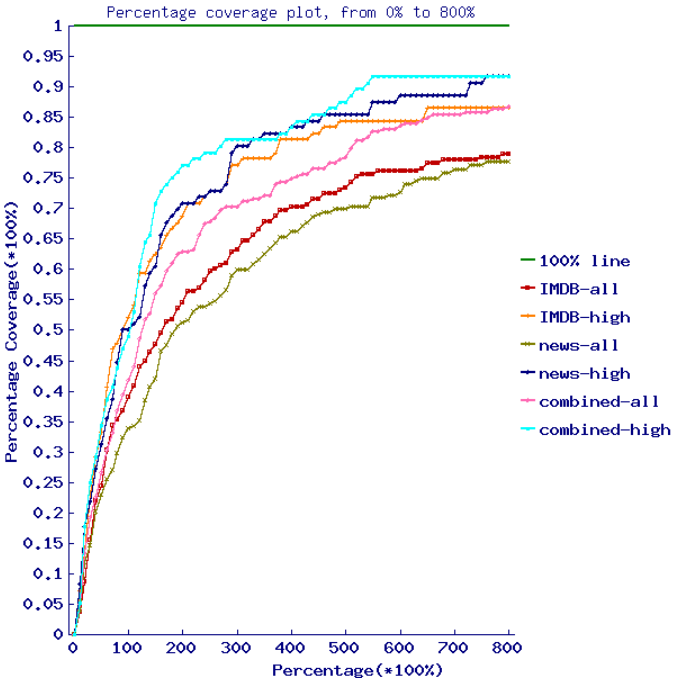

Figure 23: Comparison of Regression Models for movie set Movie$_{Budget-}$. These models use IMDB data, news data and their combination respectively. The combined model works best among all three models, both overall performance and high-grossing performance.

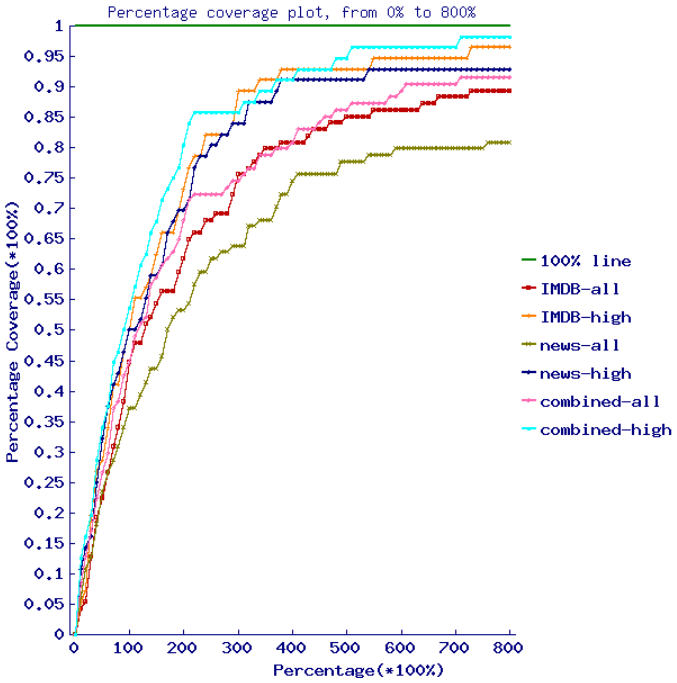

Figure 24: Comparison of Regression Models for movie set Movie$_{Budget+}$. These models use IMDB data, news data and their combination respectively. The combined model works best among all three models, both overall performance and high-grossing performance.

4.10 Conclusion and Future Work

We have discussed the correlation of movie grosses with both traditional IMDB data and movie news data, and built the models with IMDB data, news data, and their combination respectively. We conclude that:

1. Movie news data are highly correlated with movie grosses: Movie grosses are highly correlated with the news references of movie entities, e.g., movie title, directors, and actors. The news references around movies' release time have higher correlation than the references far away from movies' release time. The post-release news data are higher correlated with gross than pre-release data. The positive reference counts are higher correlated with gross than negative reference counts.
2. The movie gross prediction can be done by either IMDB data, news data, or their combination. Prediction models using merely news data can achieve similar performance with models using IMDB data, especially for high-grossing movies.

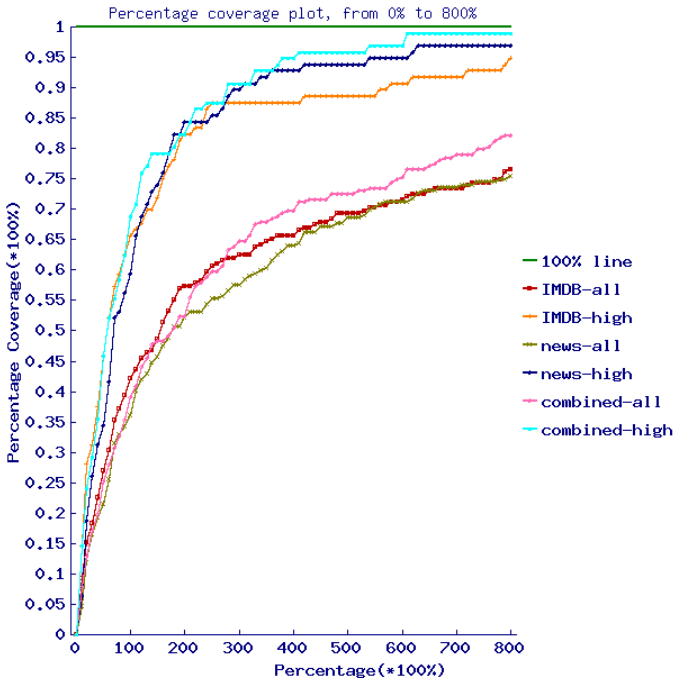

Figure 25: Comparison of k-NN Models for movie set $\text{Movie}_{Budget-}$. These models use IMDB data, news data and their combination respectively. The combined model works best among all three models, both overall performance and high-grossing performance. One interesting fact is that the combined model is worse than IMDB model in terms of 250% percentage coverage or below, but it performs better after that.

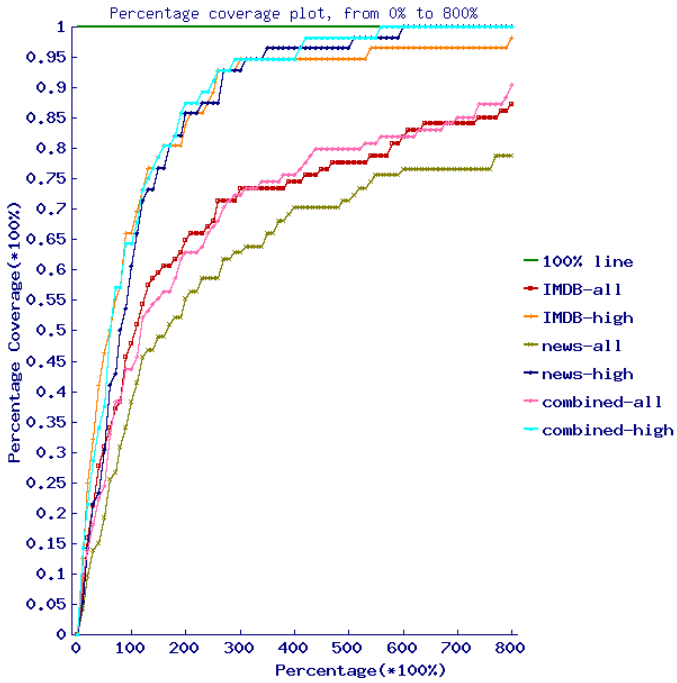

Figure 26: Comparison of k-NN Models for movie set $\text{Movie}_{Budget+}$. These models use IMDB data, news data and their combination respectively. The combined model works best among all three models, both overall performance and high-grossing performance. One interesting fact is that the combined model is worse than IMDB model in terms of 300% percentage coverage or below, but it performs better after that.

Better prediction models can be built with the combination of IMDB data and news data.

3. Three different modeling methodologies can be used for movie gross prediction: regression modeling, piecewise linear modeling, and k-nearest neighbor classification. With the same indicators, regression models have better performance than k-NN models for low-grossing movies, but k-NN models are more "accurate" and perform much better than regression models for high-grossing movies. Piecewise linear regression models are perform better than general linear regression models if the movies are grouped properly.

4. The article counts for movie entities are good predictors for movie grosses. The news sentiment data are good predictors for k-NN models, but not good predictors for regression models.

From this paper, we can see the *Lydia* news data are very helpful to forecast movie grosses. Obviously, more accurate news reference data generate better prediction results. In the future, we can do further work as the following:

1. In terms of the news data generation, we may improve our text processing ap-

proach or sentiment analysis mechanism of *Lydia* to output more accurate news reference counts or sentiment counts.

2. In terms of news data source, we may feed *Lydia* some movie specific articles or reviews, which is expected to generate more informative results than just using general daily public news.
3. In terms of movie indicators, we can try to add some other predictors, for example, the movie's director, distributor, or the fame of the actors. Actually the fame of the actors can be computed from the historical news data based on the corresponding references and sentiment references.
4. In terms of statistical learning methods, we can use Bayes or neural net classifiers.
5. In terms of the target of forecasting, we can try to predict other movie related variables as well, e.g., IMDB movie rating, etc.

5 Summary and Future Work

To summarize, financial analysis using text data is an interesting research topic for finance people, computer science people as well as the finance industry. This paper investigates the previous related works, which proves that linguistic information has some predictive power for finance variables like daily prices, currency exchange rates, or earnings, and thus it can be used in the broad area of financial analysis. Moreover, the paper also examines the predictive power of linguistic information with using data output from *Lydia*, a large-scale news analysis system, by verifying the effectiveness of prediction models based on *Lydia* data. More specifically, we find the news data have significant correlation with some quantitative information in the society, such as movie grosses, and therefore can be used for forecasting. In addition, this paper set up models and evaluates their performance for movie gross prediction. The result shows models using only *Lydia* data have comparable forecasting accuracy with models using IMDB data, and models using the combination of *Lydia* data and IMDB data achieve even better performance than models using IMDB data only. These result encourage us to explore more research topics in financial analysis area with *Lydia* data.

Belows are some identified future works:

1. Volatility: We have news from the Wall Street Journal, Dow Jones News Services, and news from other marketing research firms. After these text sources are processed by *Lydia*, we can expect to predict the volatilities of single stocks or options of the following day. Intuitively, we think highly controversial opinions, e.g., high subjectivities, would increase risk or volatilities, but we can simulate and verify that.
2. Trading volume: Similarly, high subjectivities may also cause more trades. We can prove this by *Lydia* data and daily trading figures.
3. Earnings and other less time sensitive finance variables: Currently, *Lydia* is only capable of doing daily processing, which means we cannot do intraday or finer time scale forecast for prices. However, we can predict other finance variables that do not require fine-granularity data points, like earnings.

4. *Lydia* improvement: We can try to improve the accuracy of the *Lydia* text analysis mechanism, or improve time granularity, then more interesting finance variables can be forecasted with the help of *Lydia*.
5. A general forecasting environment: We can try to build a uniform "analyzing-modeling-predicting-evaluating" environment, which will be capable of analyzing and forecasting any financial variables we are interested in with input text content.

References

[AF04] Werner Antweiler and Murray Z. Frank. Is all that talk just noise? the information content of internet stock message boards. *Journal of Finance*, 3:1259–1294, June 2004.

[AF06] Werner Antweiler and Murray Z. Frank. Do u.s. stock markets typically overreact to corporate news stories? *Working Paper, University of British Columbia, Vancouver*, 2006.

[BSV98] Nicholas Barberis, Andrei Shleifer, and Robert Vishny. A model of investor sentiment. *Journal of Financial Economics*, 49:307–43, 1998.

[BW07] Malcolm Baker and Jeffrey Wurgler. Investor sentiment in the stock market. *Journal of Economic Perspectives*, 21, no.2:129–151, Spring 2007.

[Cha03] Wesley S. Chan. Stock price reaction to news and no-news: Drift and reversal after headlines. *Journal of Financial Economics*, 70:223–260, 2003.

[Che02] Andrew Chen. Forecasting gross revenues at the movie box office. *Working paper, University of Washington, Seattle, WA*, June 2002.

[Cho99] Wing-Sing Vincent Cho. Knowledge discovery from distributed and textual data. *Dissertation of Hong Kong University of Science and Technology*, 1999.

[CPS89] David M. Cutler, James M. Poterba, and Lawrence H. Summers. What moves stock prices? *Journal of Portfolio Management*, 15:4–12, 1989.

[CWZ99] V. Cho, B. Wuthrich, and J. Zhang. Text processing for classification. *Journal of Computational Intelligence in Finance*, 7:6–22, 1999.

[Dow] DowJonesNewsAnalytics. http://www.djnewsanalytics.com.

[FYL02] G.P.C. Fung, J.X. Yu, and W. Lam. News sensitive stock trend prediction. In *Proceedings of 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 481–493, 2002.

[FYL03] G.P.C. Fung, J.X. Yu, and W. Lam. Stock prediction: Integrating text mining approach using real-time news. In *Proceedings of IEEE Int. Conference on Computational Intelligence for Financial Engineering*, pages 395–402, 2003.

[GE03] G. Gidófalvi and C. Elkan. Using news articles to predict stock price movements. *Technical Report, Department of Computer Science and Engineering. University of California, San Diego*, 2003.

[Gid01] G. Gidófalvi. Using news articles to predict stock price movements. *Project Report, Department of Computer Science and Engineering, University of California, San Diego*, 2001.

[GSS07] N. Godbole, M. Srinivasaiah, and S. Skiena. Large-Scale Sentiment Analysis for News and Blogs. In *Proc. of First Int. Conf. on Weblogs and Social Media*, pages 219–222, March 2007.

[HLS00] Harrison Hong, Terence Lim, and Jeremy Stein. Bad news travels slowly. *Journal of Finance*, 55:265–95, 2000.

[Hol] HollywoodSockExchange. http://www.hsx.com.

[Ivo] Ivontu. http://www.ivontu.com.

[LKS05] L. Lloyd, D. Kechagias, and S. Skiena. Lydia: A system for large-scale news analysis. In *Proc. 12th String Processing and Information Retrieval (SPIRE 2005)*, volume LNCS 3772, pages 161–166, Buenos Aires, Argentina, 2005.

[LKS06] L. Lloyd, P. Kaulgud, and S. Skiena. Newspapers vs. blogs: Who gets the scoop? In *Proceedings of Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006)*, Stanford University, 2006.

[LSL+00a] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan. Language models for financial news recommendation. In *Proceedings of 9th Int. Conference on Information and Knowledge Management*, pages 389–396, 2000.

[LSL+00b] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan. Mining of concurrent text and time series. In *Proceedings of 6th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, pages 37–44, 2000.

[MBL+06] Andrew Mehler, Yunfan Bao, Xin Li, Yue Wang, and Steven Skiena. Spatial Analysis of News Sources. *IEEE Trans. Vis. Comput. Graph.*, 12:765–772, 2006.

[Mit04] M.-A. Mittermary. Forecasting intraday stock price trends with text mining techniques. In *Proceedings of 37th Annual Hawaii International Conference on System Sciences (HICSS)*, pages 64–73, 2004.

[MK06a] M.-A. Mittermayer and G.F. Knolmayer. Newscats: A news categorization and trading system. In *Proceedings of the International Conference in Data Mining (ICDM06)*, 2006.

[MK06b] Marc-Andr Mittermayer and Gerhand F. Knolmayer. Text mining system for market response to news: A survey. *Working Paper No 184*, August 2006.

[Pre] PredictWallStreet. http://www.predictwallstreet.com/.

[PW02] D. Peramunetilleke and R.K. Wong. Currency exchange rate forecasting from news. In *Proceedings of 13th Australasian Database Conference*, 2002.

[Rol88] Richard W. Roll. *R*-squared. *Journal of Finance*, pages 541–566, 1988.

[SD06] R. Sharda and D. Delen. Forecasting box-office success of motion pictures with neural networks. *Expert Systems with Applications*, 30:243–254, 2006.

[SE96] Mohanbir S. Sawhney and Jehoshua Eliashberg. A parsimonious model for forecasting gross box-office revenues of motion pictures. *Marketing Science*, Vol. 15, No. 2:113–131, 1996.

[SGS04] Y. Seo, J.A. Giampapa, and K. Sycara. Financial news analysis for intelligent portfolio management. *Technical Report CMU-RI-TR-04-04, Robotics Institute, Carnegie Mellon University, Pittsburgh*, January 2004.

[SGS05] Y. Seo, J.A. Giampapa, and K. Sycara. Text classification for intelligent portfolio management. *Technical Report CMU-RI-TR-02-14, Robotics Institute, Carnegie Mellon University, Pittsburgh*, May 2005.

[Shi81] Robert J. Shiller. Do stock prices move too much to be justified by subsequent changes in dividends? *American Economic Review*, 71:421–436, 1981.

[SM00] R. Sharda and E. Meany. Forecasting gate receipts using neural network and rough sets. In *Proceedings of the International DSI Conference*, pages 1–5, 2000.

[SS00] Jeffrey S. Simonoff and Ilanna R. Sparrow. Predicting movie grosses: Winners and losers, blockbusters and sleepers. *Chance*, 13(3):15–24, 2000.

[Tho] ThomsonOne. http://thomsononeim.com.

[Tho03] J.D. Thomas. News and trading rules. *Dissertation of Carnegie Mellon University, Pittsburgh*, 2003.

[TSTM07] Paul C. Tetlock, Maytal Saar-Tsechansky, and Sofus Macskassy. More than words: Quantifying language to measure firms' fundamentals. In *Proceedings of 9th Annual Texas Finance Festival*, May 2007.

[WCe98] B. Wuthrich, V. Cho, and etc. Daily prediction of major stock indices from textual www data. In *Proceedings of 4th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, pages 364–368, 1998.

[YJDS06] Ting Yu, Tony Jan, John Debenham, and Simeon Simoff. Classify unexpected news impacts to stock price by incorporating time series analysis into support vector machine. pages 2993–2998, July 2006.