Towards Building Ranking Models with Annual Reports

Xin Ying Qiu  
Luter School of Business  
Christopher Newport University  
Newport News, VA 23606  
xinying.qiu@gmail.com

ABSTRACT: The textual content of company annual reports has proven to contain predictive indicators for the company future performance. This paper addresses the general research question of evaluating the effectiveness of applying machine learning and text mining techniques to building predictive models with annual reports. More specifically, we focus on these two questions: 1) the feasibility of building ranking models with annual reports to rank future firm performance and 2) the effect of integrating meta semantic features to help improve and support our prediction. We compare models built with different ranking algorithms and document models. We evaluate our models with a simulated portfolio. Our results show significantly positive average returns over 5 years with a power law trend as we increase the ranking threshold. Adding meta features to document model has shown to improve ranking performance. The SVR & Meta-augmented model outperforms the others and provides potential for explaining the textual factors behind the prediction.

Categories and Subject Descriptors  
I.2.7 [Natural Language Processing]: Text analysis; H.2.6 [Database Machines]; F.1.1 [Models of Computation]

General Terms: Annual reports, Text mining, Machine learning techniques

Keywords: Information retrieval, Documentation models, Ranking models, Annual reports

Received: 11 March 2010, Revised 19 April 2010, Accepted 28 April 2010

1. Introduction

Corporate disclosures play an important role in supporting market efficiency and integrity. There are in general three types of disclosures: mandatory (or regulated) disclosures such as annual and quarterly reports, voluntary disclosures by management such as earnings press releases, and reports from information intermediaries such as financial news stories. Annual reports, a major mandatory disclosure, are regulated by Securities and Exchange Commission (SEC) to minimize the information asymmetry between investors and management. These reports are publicly available and contain quantitative data on firms’ current financial performance, as well as qualitative analysis, narrative discussion on changes of strategies and operations, and forward-looking information. Investors and analysts depend on these financial and non-financial disclosures to assess firm values and make investment decisions such as choosing a portfolio of securities.

In accounting and finance domains, researchers have studied how the quality of mandatory disclosures is related to the forecast of company performance. For example, Barron et al.[3] studied the relationship between the SEC’s ratings of Management Discussion & Analysis section of annual reports and analysts’ earnings forecasts. They found that higher report ratings and better disclosure quality were associated with more accurate earnings forecasts. Gelb and Zarowin[12] empirically confirmed that high disclosure firms provided greater stock price informativeness to the investors. These studies verified how the disclosure level and quality affect company performance forecast and stock market efficiency. However, these studies relied on the subjective ratings from analysts and SEC which are no longer available and a smaller sample size due to the labor-intensive document analysis process.

Another focus in the study of the textual disclosure is on the narrative features and their roles in prediction. Li[20] found that the annual reports of firms with lower earnings were harder to read, while firms with more persistent positive earnings provided easier to read annual reports. Davis et al.[8] showed that the positive or negative tone in earnings press releases is associated with firm’s future performance, and captured in market returns. Most of the studies on the textual features tend to focus on a few linguistic features or writing styles that are specified ex ante such as the risk sentiment[19, 9], the tone[10], or readability[20, 30].

Core[6] suggested that the study of disclosure could substantially benefit from the techniques in natural language processing, computer science and artificial intelligence. The machine learning and text mining techniques, though widely applied in information retrieval, biomedicine, and web domains, has rarely been used in the study of textual disclosures, whether mandatory, voluntary, or through intermediary. Some exceptions include the following. Henry[15] applied the data mining algorithm (classification and regression tree) to study the market reaction to firms’ earnings announcement. She found that the verbal features and writing style of earnings press releases can improve the prediction of abnormal market returns. The predictive models, though shed light on the predictive potential of textual feature, are not implementable in practice. Magnusson et. al.[22] studied the changes in the textual content as well as the quantitative data in quarterly reports with clustering methods. They found that the changes in textual content seemed to proceed the changes in financial performance.

With the above review, we see that applying data mining and text processing technologies has advantages in processing larger sample set, learning models from data, analyzing higher-dimensional feature space rather than few pre-specified parameters. Our research goal in general is to evaluate the effectiveness of applying machine learning and text mining
In Section 4, we explore the ranking algorithm in building predictive models from annual reports. The advantage of ranking over other methods such as classification is that we could bypass the need to predefine classes or categories, which could be arbitrary even though well-founded. 2) Besides incorporating the entire vocabulary in the corpus to capture all potential predictive features, we experiment with a meta feature augmented document model to further explore semantic insights.

The rest of the paper is organized as follows. In Section 2, we review two important components of our methodology: ranking algorithm and document representation. We describe our data and experiment design in Section 3. We present our results and analysis in Section 4. In Section 5, we conclude with summary and thoughts on future work.

2. Methodology

In this paper, we focus on the following two research questions: 1) the feasibility of building ranking models with annual reports to rank future firm performance, and 2) the effect of integrating meta semantic features to help improve and support our prediction. Our methodologies are designed around these two questions. In this section, we briefly describe the two popular ranking techniques that we experiment with, and two different methods to model documents.

2.1. Ranking Algorithms

Learning to rank is a popular research topic in machine learning, and has been applied to many domains such as document retrieval from the web[4], collaborative filtering for movie rating[7], and web content object ordering[11, 28]. Many different algorithms have been proposed to improve ranking effectiveness, including SVM[5], neural networks[4], boosting[11], and regression-based models[13]. The more commonly used methods to evaluate different ranking algorithms include precision, recall, and F-measures. Originating from information retrieval domain, these ranking evaluation measures focus on the “positive” and the “relevant” data points instead of the general ranking order of the list. In our particular question of predicting firm performance, we are interested in both the top performing and the under-performing firms, as these assessment will influence more our investment decisions. Therefore, defining only “relevant” or “positive” firms (with documents as surrogate) to formulate standard F measures will be irrelevant here. We introduce in Section3 our methods to evaluate different models. We pick SVMs (support vector machines) as our basis algorithms based on 1) their proven efficiency in handling feature space of larger dimensions[16], 2) their ranking performance in information retrieval compared with other algorithms[23], and 3) their comparability with our previous work using SVM classifiers[2].

1. Ranking SVM

Ranking SVMs [17] take a training set of pairwise partially ordered data points and learn a ranking function. This ranking function, when applied to an unseen data set, calculates a score for each new data point with which we could generate a global ordering. Such a ranking function can generalize well beyond the partial ordering of the training set by choosing a weight factor that projects the largest distance of the two closest data points. The optimization formulation is as follows:

\[
\min \frac{1}{2}\|w\|^2 + C\sum_{i,j}\xi_{ij} \\
\text{s.t.} \quad \forall (x_i, x_j) \in R^d : (w \cdot x_i - w \cdot x_j) \geq 1 - \xi_{ij} \\
\forall (i, j) : \xi_{ij} \geq 0
\]

where \(w\) is the weight vector, \(x\) is the data vector, \(R\) is the partially-ordered training set, \(C\) is a tuning parameter that controls the training error. Ranking SVMs have been proven effective in various application such as document retrieval[5], summarization[23, 26], and web search[18].

2. SVM regression (SVR)

SVR regression[25] method takes a training data set of vectors and their associated target values, and tries to fit a function that approximates the true relation between the vectors and the targets. This learned function is designed to tolerate at most deviation from the true function. Slack variables are introduced for tolerance of larger than deviation and for feasible dual problem formulation. At the same time, the learned function needs to be as flat as possible (e.g. by minimizing the norm of the weight vector). SVR can be formulated as follows:

\[
\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\
\text{s.t.} \quad y_i - (w \cdot x_i + b) \leq \epsilon + \xi_i \\
(w \cdot x_i + b) - y_i \leq \epsilon + \xi_i^* \\
\xi_i, \xi_i^* \geq 0
\]

where \(w\) is the weight vector, \(x\) is the data vector, \(y\) is the target, \(b, R, C\) is the cost parameter, \(\epsilon\) is the error, and \(\xi\) and \(\xi^*\) are the slack variables. SVM regression has been applied to various domains such as face detection[21], and travel-time prediction[29].

2.2. Document Models

In information retrieval research, documents are typically represented as vectors of weighted terms. This is generally referred to as the vector representation model. There are three aspects to consider when building term based vector representation model: 1) how to define a term; 2) whether to use the full set of terms or a selected subset and if the latter, how to select a subset; 3) how to weight the terms in the vector model.

1. Baseline Model

The most widely-used bag of words approach is to use all the terms in the training corpus regardless of the order of the terms, to represent document vectors. This appears to be the default standard in text classification[24]. Functional or connective words are considered as stop words and are generally removed since they are assumed to have no information content. Stemming is sometimes performed to remove the suffixes and to map words to their morphological forms. This is the model we employed in our previous work[2], which we use as a default benchmark in this paper. We filtered our term space with document frequency threshold for feature selection. We weight the terms with the atn construction of TF-IDF weighting scheme. Formally:

\[
\text{Doc} = (w_1 \cdot x_1, w_2 \cdot x_2, \ldots, w_m \cdot x_m) \\
w_i = (0.5 + 0.5 \times \frac{tf}{max tf}) \times \ln \left(\frac{N}{n}\right)
\]
where $w_i$ is the *an* construction of TF×IDF term weighting scheme, $x_i$ is stemmed term, $f$ is the raw term frequency of $x_i$ in a document, $maxf$ is the highest term frequency in the document, $N$ is the total number of documents in the collection; and $n$ is the number of documents containing term $x_i$. Our method has proven effective in capturing the reports’ predictive power[2]. The advantage of this model is that it incorporates all the possible semantic details in the document at the most granulated level. On the other hand, we could easily extend this model by adding other features of interest.

2. Meta-augmented

Researchers have explored methods to improve the “bag of word” representation by integrating additional knowledge source such as Wikipedia into the model[27]. Our second model is built on top of the baseline model by adding 36 newly defined term features that capture the semantic meaning and writing style that may help capture the general features of the document and provide explanation potential. The weighting of these meta features is designed to have value range similar to the *an* version of the TF×IDF scheme. In particular, we choose from *Dictionary*, a research software that examines a text for its verbal tones[14], 31 dictionaries that capture the general semantics of a document, such as praise, inspiration, and past concern. Additionally, inspired by other researchers’ work, we identify semantic features about optimism and pessimism[8], tone[15], risk sentiment[19], and document’s readability[20].

We formulate a total of 36 meta semantic features to augment the baseline vector model as defined follows. The document frequency threshold for feature selection also apply to filtering meta features for a relatively more concise and informative feature space.

$$Doc = \{w_1 \cdot x_1, w_2 \cdot x_2, \ldots, w_j \cdot x_j, \ldots, w_m \cdot x_m, w_{D_1} \cdot x_{D_1}, w_{risk} \cdot x_{risk}, w_{tone} \cdot x_{tone}, w_{readability} \cdot x_{readability}\}$$

subject to:

$$w_i = (0.5 + 0.5 \times \frac{tf}{maxf}) \times \ln\left(\frac{N}{n}\right)$$

$$w_{D_j} = \ln(1 + TF_{D_j}) \quad \forall D_j \in \text{Dictionary}$$

$$w_{risk} = \ln(1 + TF_{risk})$$

$$w_{tone} = \ln\left(\frac{2 + TF_{optimism} - TF_{pessimism}}{TF_{optimism} + TF_{pessimism}}\right)$$

$$w_{readability} = \ln\left(\frac{\text{words}}{\text{sentences}}\right) + 100\left(\frac{\text{words width >= 2 syllables}}{\text{words}}\right)$$

(4)

where $TF$ is the term frequency of all the terms in a particular dictionary or word list, such as one of the 31 dictionaries in *Dictionary*, or the list of terms representing optimism.

3. Experiment Design

We select SAR (size-adjusted buy-and-hold return for a year) as the measure for company financial performance. Our data set covers US firms in the manufacturing industry (SIC codes from 2000 to 3999), with the fiscal year ending in December, from 1997 to 2003. We calculate SAR as the size-adjusted buy-and-hold return cumulated from 12 months from April 1 of the fiscal year to the next April. Our final experiment data set with matching financial measures contains 4280 documents from 1236 firms. This relatively restricted sample set helps us to keep track of our experiments and to ensure some degree of sample homogeneity. Figure 1 presents an overview of our experiment design, where $SAR_t$ is the size adjusted return cumulated from April 1 of year $t$ to March 31 of year $t + 1$; $Doct^t$ – is annual report for year $t$, usually available in March of year $t$.

**Notation:**

A: For firms in year $t$ – 1, build predictive model of year $t$ using firms’ SAR in year $t$ (i.e. size-adjusted return cumulated from April of year $t$ to March of year $t + 1$) and annual reports for year $t$ – 1 which are usually published in March of year $t$.

B: For firms in year $t$, apply the predictive model built in step A) to the annual reports for year $t$ which are published in March year $t + 1$, and predict the ranked list of these firms in year $t + 1$ based on SAR performance.

C: On March 31st, year $t + 1$, given a predicted ranked list of firms from step B), we sell the stocks of (for example) bottom 10% of firms from the the ranked list at a total value of (for example) 10 million dollars and buy the stocks of top 10% of the firms from the ranked list with a total value of 10 million dollars. In both the buying and selling transactions, we will allocate equal values of stocks among the firms. On March 31st, year $t + 2$, we will sell the stocks of the top firms and buy the stocks of the bottom firms. If our prediction was correct, this transaction should generate non-negative profit.

Our design is similar to [2], but has three fundamental changes in these areas:

1. Instead of classifying firms into three categories: outperforming, average-performing and under-performing in terms of SAR, we apply ranking algorithms as described in Section 2.1 without predefining the number of classes and their distributions.

2. We employ a thresholding method to construct investment portfolio and evaluate the ranking performance of our models. Typical evaluation measures of ranking algorithm include precision, F-measure, recall, and ROC area[1]. These measures are suitable for learning tasks such as information retrieval and document query, but are not applicable to ranking firms. When we evaluate a predicted ranked list of firms, it will be hard to define a “relevant” or “positive” firm as required for example
To construct a portfolio using our ranking model's prediction, we can we integrate meta-semantic features to help support our prediction. We observe the portfolio return given by different models built with 2 ranking algorithms (Ranking SVM and SVR) and 2 document representation methods (Baseline Document Vector and Meta-Augmented Document Vector). Table 1 shows the 5-year average portfolio return. The percentage threshold of 1%, for example, means that we select the top 1% and bottom 1% of firms from the predicted rank list to construct the portfolio.

We observe from Table 1 that the average returns achieved with all 4 models are all significantly positive, implying the success in correctly predicting the outperforming and under-performing firms. We can also tell that as we increase the threshold, the percentage of firms that we select from the top and bottom of the predicted rank list, the portfolio return decreases for all models, showing generally a power law trend.

We can infer that the predicted ranking of firms is able to reflect the true order of firms in terms of future SAR return.

We use the SVMlight\(^1\) implementation of Ranking SVMs and SVR. Due to the time lag in our implementable design as shown in Figure 1, we could only use data available in 1997 to build model and predict 1998. Predicting 2003 involves data of cumulative return from 2003 to 2004 which we do not have. Therefore our 6 years of data generate experiment results for a 5-year period from 1998 to 2002. In this paper, we address these two questions: 1) can the advantages of the ranking algorithm help achieve better predictive performance? and 2) can we integrate meta-semantic features to help support our prediction? The following results will provide insights to these questions.

To construct a portfolio using our ranking model’s prediction, we vary the percentage of firms that we select from the top and bottom of the predicted rank list. We observe the portfolio return given by different models built with 2 ranking algorithms (Ranking SVM and SVR) and 2 document representation methods (Baseline Document Vector and Meta-Augmented Document Vector). Table 1 shows the 5-year average portfolio return. The percentage threshold of 1%, for example, means that we select the top 1% and bottom 1% of firms from the predicted rank list to construct the portfolio.

Table 1. Portfolio Return by Changing % Threshold for Top and Bottom Firms, with Different Ranking and Document Models

<table>
<thead>
<tr>
<th>% Threshold of Top &amp; Bottom Firms</th>
<th>Ranking SVM &amp; Baseline Doc. Model</th>
<th>Ranking SVM &amp; Meta-Augmented</th>
<th>SVR &amp; Baseline Doc. Model</th>
<th>SVR &amp; Meta-Augmented</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.50%</td>
<td>47.81%</td>
<td>40.40%</td>
<td>30.63%</td>
<td>25.53%</td>
</tr>
<tr>
<td>1%</td>
<td>34.58%</td>
<td>23.40%</td>
<td>38.16%</td>
<td>40.38%</td>
</tr>
<tr>
<td>2%</td>
<td>20.51%</td>
<td>15.51%</td>
<td>21.24%</td>
<td>31.23%</td>
</tr>
<tr>
<td>3%</td>
<td>14.25%</td>
<td>7.48%</td>
<td>19.22%</td>
<td>33.72%</td>
</tr>
<tr>
<td>5%</td>
<td>10.70%</td>
<td>2.12%</td>
<td>17.43%</td>
<td>24.77%</td>
</tr>
<tr>
<td>10%</td>
<td>8.95%</td>
<td>1.95%</td>
<td>17.22%</td>
<td>20.53%</td>
</tr>
<tr>
<td>15%</td>
<td>12.07%</td>
<td>5.16%</td>
<td>19.09%</td>
<td>17.32%</td>
</tr>
<tr>
<td>20%</td>
<td>12.40%</td>
<td>8.03%</td>
<td>16.31%</td>
<td>18.99%</td>
</tr>
<tr>
<td>25%</td>
<td>10.21%</td>
<td>4.77%</td>
<td>16.01%</td>
<td>17.71%</td>
</tr>
</tbody>
</table>

Next, we compare our best model performance, SVR & Meta-Augmented Model, with our previous work\(^2\). Previously we apply a three-class classification approach and document vector model to predict company future return with annual reports. We formulate a three-class classification problem with 25%-50%-25% true class distribution for out-performing, average-performing and under-performing firms. Table 3 compares the performance of classification.

Threshold | (No. of Top Firms, No. of Bottom Firms) |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.50%</td>
<td>(4, 5)</td>
</tr>
<tr>
<td>1%</td>
<td>(7, 8)</td>
</tr>
<tr>
<td>2%</td>
<td>(15, 16)</td>
</tr>
<tr>
<td>3%</td>
<td>(22, 23)</td>
</tr>
<tr>
<td>5%</td>
<td>(37, 28)</td>
</tr>
<tr>
<td>10%</td>
<td>(73, 24)</td>
</tr>
<tr>
<td>15%</td>
<td>(110, 111)</td>
</tr>
</tbody>
</table>

\(^1\)http://svmlight.joachims.org/

---

4. Results and Analysis

We can further compare the models using Figure 2. We see that Ranking SVM is able to achieve higher return than SVR (SVM regression) at smaller threshold of 0.5% (47.81% v.s. 39.63%, and 40.40% v.s. 25.53%), with only 7 or 9 firms included in the portfolio as presented in Table 2. This implies that Ranking SVM are better able at identifying the very few top and bottom firms. However, as the threshold increases, the portfolio returns of Ranking SVM drop much more sharply and stay lower than those of SVR. We can infer that the predicted order from SVR is more accurate than Ranking SVM over the entire list. To compare the effect of document models, we observe that using Meta-augmented model helps improve SVR ranking algorithms by achieving higher returns than the Baseline Document Vector model. However, when applied to Ranking SVM, Baseline document model is better than Meta-augmented model in supporting better ranking results. Considering both the overall portfolio return and the descending trend as the threshold increases, we find that SVR & Meta-Augmented Document Model is the over-all best as it positions above all the other three models for most of the threshold points.

Next, we compare our best model performance, SVR & Meta-Augmented Model, with our previous work\(^2\). Previously we apply a three-class classification approach and document vector model to predict company future return with annual reports. We formulate a three-class classification problem with 25%-50%-25% true class distribution for out-performing, average-performing and under-performing firms. Table 3 compares the performance of classification.
with the our ranking model at 25% threshold. We see that SVR & Meta Model has higher average return (17.71% vs. 12.16%), follows similar temporal trend over years, but has greater fluctuation at times.

Table 2. Number of Top and Bottom Firms by Threshold

<table>
<thead>
<tr>
<th>Year</th>
<th>25-50-25% Class Definition</th>
<th>With 25% Threshold Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>1998</td>
<td>-2.46%</td>
<td>-4.4%</td>
</tr>
<tr>
<td>1999</td>
<td>63.68%</td>
<td>100.52%</td>
</tr>
<tr>
<td>2000</td>
<td>-36.48%</td>
<td>-52.15%</td>
</tr>
<tr>
<td>2001</td>
<td>19.6%</td>
<td>23.46%</td>
</tr>
<tr>
<td>2002</td>
<td>16.82%</td>
<td>21.12%</td>
</tr>
<tr>
<td>Average</td>
<td>12.16%</td>
<td>17.71%</td>
</tr>
</tbody>
</table>

Table 3. Comparing 25-50-25% Classification Model with 25% Threshold Ranking with SVR & Meta Model

Lastly, since we observe that meta-augmented document model improves the performance of SVR & Meta-augmented model combined with default document vector, it would be interesting to see how the semantic meta features contribute to the prediction. Our best model of SVR & Meta-augmented fits a regression model on the data set. Therefore, the sign of the meta feature weights from the model is a starting point to look into the semantic aspect of the predictive model. We build one SVR & Meta-augmented model with the entire 6-year of data. Table 4 shows the signs of the 36 meta features from the model. We find that features of inspiration, praise, and hardship correlate with the semantic as aspect of the predictive model. We build one SVR & Meta-augmented model with just the documents vocabulary. Meta features also provide the potential of explaining the semantic aspects of the prediction. We find that we could further extend our research in many directions such as refining the semantic features to better reference the business and accounting language, gaining insights on the semantic differences between firms and across different years, and distinguishing language expressing opinions, projections, and facts.

6. Acknowledgment

The author acknowledges, with gratitude, the constructive comments and helpful insights from Ramji Balakrishnan, Padmini Srinivasan, and Nick Street.

References


