Analysis of year-over-year changes in Risk Factors Disclosure in 10-K filings

Vipula Rawte
Computer Science Department
Rensselaer Polytechnic Institute
Troy, NY 12180
rawtev@rpi.edu

Aparna Gupta
Lally School of Management
Rensselaer Polytechnic Institute
Troy, NY 12180
guptaa@rpi.edu

Mohammed J. Zaki
Computer Science Department
Rensselaer Polytechnic Institute
Troy, NY 12180
zaki@cs.rpi.edu

ABSTRACT

Risk Factor Disclosures – Item 1A – in 10-K forms filed with SEC is one of the important sections since it contains a company's yearly risk updates, and thus helps investors decide whether to invest in a company or not. It is crucial to read this section carefully in order to make better investment choices. Given the large number of such forms filed on a yearly basis, it is very cumbersome for humans to understand and analyze them to make informed decisions. We discuss the task of bank failure classification using textual analysis on item 1A for various banks' 10-K forms, i.e., to predict whether a bank will fail or not. We also analyze other quantitative bank performance indicators like leverage and Return On Assets (ROA), and see how well text-based methods can predict those risk indicators. In particular, to create our textual corpora, we focus on the changes in the 1A sections, retaining only those sentences that have under 30% and 40% similarity over two consecutive years (for the same bank). We implement deep learning and other supervised learning techniques like Convolutional Neural Networks (CNN), Support Vector Machines (SVM) and Linear Regression. We also combine the word sentiment polarities along with their count as our weighted feature vector.

CCS Concepts

•**Information systems** → **Sentiment analysis; Near-duplicate and plagiarism detection; Clustering and classification;** *Data mining;*

Keywords

risk factors, 10-K filing, SEC, banks, change detection, CNN, SVM, regression, sentiment

DSMM '18 June 15, 2018, Houston, Texas USA

DOI: https://doi.org/10.1145/3220547.3220555

1. INTRODUCTION

The United States (US) Securities and Exchange Commission (SEC) requires an comprehensive summary of a company's financial performance known as a Form 10-K which is filed annually. The section 1A of a 10-K form is an important one because it gives the associated risk updates. The SEC started requiring risk factors disclosures in annual reports in 2005, and since then, it has been of great concern to the investors if the information in the disclosure gives any insight. Specially, companies that undergo economic changes are supposed to revise their risk disclosure section and provide the most up-to-date information that will help investors make decisions about the company. But it is seen from previous analysis that the managers do not provide accurate and latest information so as to cut on such additional costs and instead just extend the document length by providing boilerplate redundant information which hardly gives any signals about a company's status. If this information is misleading, it can lead to investment failures.

We try to classify and predict bank failures using deep neural networks on section 1A. We use CNNs (convolutional neural networks) since they performs significantly well with textual data. Further, we also implement weighted sentiment analysis on the words in the changed sentences corpus and run a SVM (support vector machines) classifier on it. Finally, we perform multiple and multivariate linear regression analysis on the performance indicators associated with banks, such as leverage and ROA. We perform all our analysis using data from 2006 to 2017 (inclusive), starting from when the SEC mandated the usage of risk factor disclosures in the 10-K form.

2. LITERATURE SURVEY

Investors are quite concerned about the informativeness of the risk factors since they can be rather generic and thus SEC emphasizes on improving the disclosures [15, 2]. For example, [1] shows that it is quite possible that the risk disclosures' information might be overlooked.

For textual analysis, the foremost step is to extract features from the text. A standard approach is to use a bag of words approach which can further be combined with the

Term Frequency - Inverse Document Frequency (TF-IDF) weighting scheme. Predicting bank failure using common approaches like neural networks, regression, SVM, and k-means clustering on numerical features like equity prices, stock prices, and returns has been considered in previous work [3, 14, 20], but the use of textual analysis is still in the early stages of research. There has been some work done in this direction as well [6, 5, 17, 12], such as, using sentiment and tone from the textual reports to predict the failure but there is still a lot of scope for improvement.

Brown & Tucker discuss the informativeness of the Management Discussion and Analysis (MD&A) section – item 7 of the 10-K form – where they develop a modification score based on the the length and similarity of the documents over two consecutive years. Their analysis is done on the entire section 7 and thus does not investigate the changed subsections, which is important since there is a lot of redundant information for that section for a given bank over two consecutive years. Their study shows that the modification score decreases as the length of the document increases over years, thus signaling the use of boilerplate information which simply adds to the length of the document without giving any significant information [4].

Cohen and Lou [8] study the similarities in the MD&A section using the similarity measures such as cosine similarity, Jaccard similarity, minimum edit distance and simple similarity. They focus more on how the changes would impact the future stock returns and the future litigation events.

Hanley [13] uses Latent Dirichlet Allocation and Semantic vector analysis to extract risk themes from the financial firm 10-K files.

Recently, CNNs have proved popular in text analytics, especially since they work faster and better for textual analysis than Recurrent Neural Networks (RNNs). Also, CNNs maintain the semantics of every word since the entire sentence is encoded as a vector and is fed as an input to the network [16, 22]. For example, [9] shows that building features from the context of the corpus is more efficient than manually crafted features. Carefully choosing and tuning the hyperparameters has also shown to give significantly better results [22].

Furthermore, word embedding approaches have proven to be very successful in capturing the latent semantics and underlying relationships between co-occurring terms in a context. In our work we use GloVe [19], since it gives better accuracy as compared to Word2Vec [18]. While Word2Vec predicts the context of given word, GloVe learns from a co-occurrence matrix based on the frequency with which a word appears in a context. It is useful to use word embeddings since they are geometrical encodings of words determined by their frequency in text corpus which also captures the semantics.

3. DATASET

We extracted the 10-K filings in html format from SEC's Electronic Data Gathering, Analysis, and Retrieval (EDGAR) website and parsed them into individual sections, and stored each section as an individual text document. We selected a total of 883 Bank Holding Companies (BHC) out of which 826 and 57 are identified as non-failed (NF) and failed (F) banks respectively (many of these failed during the 2008 financial crisis). Our section extraction script successfully extracted 93.55% of the 1A sections given the complexity of extracting text from the html 10-K Forms.

We considered only those 1A sections which are more than 150 bytes in size, ignoring the ones which have unimportant information like "Not applicable" or "Not required for smaller reporting companies", and so on.

Table 1: Dataset

	Total Banks	F	NF
All Banks	883	57	826
Filtered Banks	730	55	675

Additional filtering involves only keeping banks, identified by their Central Index Key (CIK), if section 1A appears in two consecutive years and ignore the others. Thus, we finally get a total number of 730 CIKs with 675 non-failed and 55 failed banks, respectively. Federal Deposit Insurance Corporation (FDIC; www.fdic.gov/bank/individual/failed/banklist.html) lists the failed and non-failed banks, tat we use to make the determination on bank failure. This is shown in Table 1.

Text preprocessing involves several steps. Each textual document is split into sentences. Further, each sentence is tokenized into words. We used Natural Language Toolkit (NLTK) to remove stopwords and performed lemmatization to get rid of redundant words.

We create three corpora of 1A sections: (1) using all the sentences in section 1A, and (2) using only the changed sentences via (a) 30% similarity and (b) 40% similarity threshold. For the latter, we discard overly similar sentences from one year to the next, since our aim is to see what information has changed. We used a python library called difflib (pymotw.com/2/difflib) to create our corpus of changed sentences by comparing each sentence with every other sentence over two consecutive years for a given bank. Since unlike the much smaller 8-K forms, 10-K forms have around a median of 93% similarity in 1A section for NF to F and 96.5% for NF to NF between two consecutive bank-years [7], we compute the similarity of two sentences and extract 30% and 40% similar sentences as part of our changed corpus.

4. METHODS

We study and implement supervised learning techniques. We focus mainly on Convolutional Neural Network using GloVe word embeddings. Others include SVM and Random Forest Classifier based classification using weighted sentiment word polarities along with linear and nonlinear regression using weighted word sentiment polarities as features with (i) leverage, (ii) ROA and (iii) leverage and ROA as dependent variables. We apply all the above mentioned techniques on a corpus of all versus changed data.

4.1 Convolutional Neural Network with GloVe word embeddings

We use GloVe word embeddings for representing words and a CNN for learning how to classify the text documents. Neural networks generally perform better than traditional linear classifiers, specially when combined with word embeddings [11]. CNN shows superior results at document classification because it can pick out features such as tokens or sequences of tokens irrespective of the position in the sentence [21, 10].

Table 2: Train Test Dataset

	Total Banks	F	NF	number of words
Training Set	511	38	473	289021
Testing Set	219	16	203	112780

We use Keras deep learning framework (keras.io) to train our model. We use 100-dimensional GloVe word embedding vectors trained on our corpora instead of the already pre-trained embeddings. We maintain a maximum sequence length of 1000 words – truncate if more than that, and pad with zeros if less than that. We observe that the sentences in 1A section are usually not longer than 1000 words and thus we do not lose any textual information. For training, we use a 1D convnet with three layers. We set our network parameters as follows: filter=128, max_pooling=5, prediction_activation_function=softmax. Further, we use RMSprop optimizer with parameters tuned to lr=0.001, rho=0.9, epsilon=1e-08, loss=categorical_crossentropy and finally use callbacks including earlystop and reducer. Table 2 shows the split of our corpus into training and testing sets.

4.2 Weighted Sentiment Analysis

We find out word sentiment polarity using a python library called TextBlob (http://textblob.readthedocs.io/en/dev). These word features are then multiplied with the count features of the word to get a weighted sentiment polarity feature. This involves removing neutral words via two approaches: (1) pre - before taking the polarity and count product and (2) post - after multiplying both the feature vectors. We set a range of -0.5 to +0.5 and -2.0 to +2.0 to get rid of the neutral words for the pre- and post-filtering, respectively. We then run Support Vector Machine and Random Forest Classifier (max_depth=2) on these feature vectors to again predict bank failures.

4.3 Regression

We extend our analysis further to other quantitative bank performance indicators such as leverage and ROA (Return on Assets). We perform both linear and non-linear (Support Vector Regression (SVR)) regression (C=1e3, gamma=0.1, degree=2). We use scikit-learn (http://scikit-learn.org) for these tasks. We take the feature vector obtained from the previous weighted sentiment polarities and perform linear and nonlinear regression. We carried out 10 runs with shuffled train and test datasets with a 30% split.

5. RESULTS

We now present the results of the methods implemented in this paper. We start with the CNN + GloVe method. Table 3 shows the comparison for all versus changed corpora on failed versus non-failed bank prediction. We conclude that CNN + GloVe performs better in classifying the banks as failed or non-failed based on the textual information and the pre-defined labels.

Performing linear and nonlinear regression on weighted sentiment features gives r2 score and mean squared error. The values show that nonlinear model can fit the data relatively well. The statistics are given in Table 4. This also shows that a relation between the information present in the risk factors disclosures can be mapped to bank failures as well as bank performance indicators, and thus supports our analysis of section 1A.

6. CONCLUSION AND FUTURE WORK

We analyzed the 1A section of 10-K forms to see if it can be related to classifying banks as failed or non-failed. We find that CNN does well on this task, with an accuracy of 96.8%. We also tried to relate the sentiment with the word counts, and used SVM and Random Forest classifier. Finally, the regression on bank performance indicators gives quite convincing results. The results can be further improved by more fine tuning and optimization and also implementing dimensionality reduction techniques. Overall, our work gives a good sense of how textual information can be related to bank failures via text mining and learning.

As part of our future work, we will use weighted sentiment as extra features in the CNN. Another aspect would be to carry out the analysis over 3-4 years' segments instead of over entire period. We also intend to carry out similar analysis on section 7 of 10-K filings using additional supervised and unsupervised techniques including the ones implemented in this paper. Another interesting direction is dynamic topic modeling to see how the performance of the banks changes over years in terms of latent topics.

7. ACKNOWLEDGMENTS

This work was supported in part by NSF Award III-1738895.

8. REFERENCES

[1] Disclosure overload and complexity: hidden in plain sight, 2011.

[2] *The Corporate Risk Factor Disclosure Landscape.* 2016.

[3] J. E. Boritz and D. B. Kennedy. Effectiveness of neural network types for prediction of business failure. *Expert Systems with Applications*, 9(4):503 – 512, 1995. Expert systems in accounting, auditing, and finance.

[4] S. Brown and J. Tucker. Large-sample evidence on firms year-over-year MD&A modifications. *Journal of Accounting Research*, 49(2):309–346, 5 2011.

[5] P. C. Tetlock. Giving content to investor sentiment: The role of media in the stock market. 62:1139–1168,

Table 3: Document Classification Results: Acc denotes Accuracy, Prec denotes Precision

Method	All Data				40% Similar Data				30% Similar Data			
	Acc	Prec	Recall	F1 score	Acc	Prec	Recall	F1 score	Acc	Prec	Recall	F1 score
CNN + GloVe	0.932	0.932	0.932	0.932	0.941	0.941	0.941	0.941	**0.968**	0.968	0.968	0.968
LSTM + GloVe	0.895	0.895	0.895	0.895	0.945	0.945	0.945	0.945	0.926	0.926	0.926	0.926
Pre-Weighted Sentiment Analysis + SVM	0.932	0.94	0.93	0.90	0.932	0.87	0.93	0.90	0.913	0.83	0.91	0.87
Post-Weighted Sentiment Analysis + SVM	0.941	0.88	0.94	0.91	0.922	0.85	0.92	0.89	0.913	0.83	0.91	0.87
Pre-Weighted Sentiment Analysis + Random Forest	0.922	0.85	0.92	0.89	0.936	0.88	0.94	0.91	0.936	0.88	0.94	0.91
Post-Weighted Sentiment Analysis + Random Forest	0.913	0.83	0.91	0.87	0.922	0.85	0.92	0.89	0.932	0.87	0.93	0.90

Table 4: Regression Analysis Results

	Leverage				ROA				Leverage + ROA
	Linear Regression	SVR RBF	SVR Linear	SVR Polynomial	Linear Regression	SVR RBF	SVR Linear	SVR Polynomial	Multivariate Linear Regression
r2 score: Train	0.89	0.24	0.04	-0.15	0.9	-0.84	-0.84	-0.84	0.88
rmse score: Train	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00
rmse score: Test	77.6	0.01	0.06	1.42	0.01	0.00	0.00	0.00	1.30

02 2007.

[6] M. Cecchini, H. Aytug, G. Koehler, and P. Pathak. Making words work: Using financial text as a predictor of financial events. 50:164–175, 12 2010.

[7] Y. Chen, R. M. Rabbani, A. Gupta, and M. J. Zaki. Comparative text analytics via topic modeling in banking. In *IEEE Symposium on Computational Intelligence for Financial Engineering and Economics*, 2017.

[8] L. Cohen and D. Lou. Lazy prices. *SSRN Electronic Journal*, 2010.

[9] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. P. Kuksa. Natural language processing (almost) from scratch. *CoRR*, abs/1103.0398, 2011.

[10] C. N. Dos Santos and B. Zadrozny. Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1818–II–1826. JMLR.org, 2014.

[11] Y. Goldberg. A primer on neural network models for natural language processing. *CoRR*, abs/1510.00726, 2015.

[12] A. Gupta, M. Simaan, and M. J. Zaki. When positive sentiment is not so positive: Textual analytics and bank failures. *Available at SSRN: Social Science Research Network, 2773939*, 2016.

[13] K. W. Hanley. Dynamic interpretation of emerging systemic risks. *SSRN Electronic Journal*, 2016.

[14] T. J. Curry, P. J. Elmer, and G. Fissel. Can the equity markets help predict bank failures? 07 2004.

[15] S. Johnson. Sec pushes companies for more risk information -, Aug 2010.

[16] Y. Kim. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882, 2014.

[17] F. Li. Do stock market investors understand the risk sentiment of corporate annual reports? 04 2006.

[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[19] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *In EMNLP*, 2014.

[20] K. Y. Tam and M. Y. Kiang. Managerial applications of neural networks: The case of bank failure predictions. *Management Science*, 38(7):926–947, 1992.

[21] X. Zhang, J. J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. *CoRR*, abs/1509.01626, 2015.

[22] Y. Zhang and B. C. Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *CoRR*, abs/1510.03820, 2015.