Extraction of Forward-looking Financial Information for Stock Price Prediction from Annual Reports Using NLP Techniques

Alexander Glodd
HWR Berlin, Germany
alexander.glodd@gmail.com

Diana Hristova
HWR Berlin, Germany
diana.hristova@hwr-berlin.de

Abstract

Annual reports are one of the most important sources of information for financial decisions. They contain forward-looking statements (FLS), which describe future trends and expectations. Thus, several studies deal with the automated identification of FLS, where the latest ones involve a combination of a rule-based approach and machine learning classification. In this paper, we extend this research with state-of-the-art NLP methods. We use DistilBERT for FLS identification and determine their sentiment with FinBERT. The result is processed by a Random Forest model for stock price growth prediction of different periods. Our evaluation shows that DestilBERT achieves higher accuracies on FLS identification than existing methods. For short-term stock price rate prediction, the extracted FLS information together with historical stock data outperforms the sole use of historical stock data. For mid-term prediction, using FLS alone with DestilBERT shows the best result. Finally, in the long-term, FLS provide no benefit.

Keywords: forward-looking statements, annual report, 10-K, BERT, stock price prediction

1. Introduction

At the core of the finance and accounting fields is the generation and analysis of large volumes of data, ranging from company-level financial metrics to very detailed financial reports for a given product, customer or market. In recent years, also facilitated by the technological development, the data volume and its accessibility has increased even more (Fisher, Garnsey, & Hughes, 2016). Additionally, the fraction of data being in textual, unstructured form, such as annual reports, analyst reports, ad-hoc news or social media posts, rose substantially (Lewis & Young, 2019). Among those texts, annual reports are seen as one of the most important sources of information for analysts and investors (Masson & Paroubek, 2020). In particular, they contain both quantitative financial metrics on company performance and the corresponding textual analysis of those numbers from the management's perspective. These reports are mostly manually read to extract important information for decision-support (Hsieh & Hristova, 2022).

However, in recent years, both the number, length and redundancy of annual reports grew tremendously, making the task very labor-intensive (Dyer, Lang, & Stice-Lawrence, 2017). Also, according to Dyer et al. (2017) it would take "…21.21 years of formal education." (p. 11) for someone to fully understand the median 10-K annual report. This is an annual report required by the U.S. Securities and Exchange Commission (SEC) for all publicly traded companies. As a result, researchers recognized the need for approaches to automatically analyze annual reports, thus increasing efficiency and objectivity (Hsieh & Hristova, 2022; Lewis & Young, 2019).

The field of Natural Language Processing (NLP) deals with the automated analysis of textual, unstructured data such as the identification of the topics or the mood (known as sentiment) in a given annual report. Most state-of-the-art approaches in the field follow a three-step approach: 1) the text is transformed in a structured form that can be analyzed by computers, 2) important features (e.g. sentiment, cf. Krinitz and Neumann (2021)) are extracted from the resulting output and 3) those are related to company performance, such as the prediction of IPO valuation (Tao, Deokar, & Deshmukh, 2018), stock price (Hsieh & Hristova, 2022; Kraus & Feuerriegel, 2017) or period returns (Loughran & McDonald, 2011).

However, most works analyze the whole text, which implies that they consider both backward- and forward-looking statements (FLS). Here we define FLS as "…a short sentence that contains information likely to have, or reft to, a direct effect in the foreseeable future…" (Noce, Zamberletti, Gallo, Piccoli, & Rodriguez, 2014, p. 2). As opposed to FLS, backward-looking statements are not very valuable for future investment decisions, as they are usually already known upon publication and therefore reflected in indicators such as the current stock price. The SEC itself stressed the importance of more FLS in

URI: https://hdl.handle.net/10125/103313
978-0-9981331-6-4

the "Management's Discussion and Analysis of Financial Condition and Results of Operations" (MD&A) Item of 10-K reports (SEC, 2003), which then should be identified for analysis purposes.

As a result, a small number of papers focus on extracting FLS from financial texts (mostly the MD&A Item) and potentially relating those with company performance (Li, 2010; Muslu, Radhakrishnan, Subramanyam, & Lim, 2015; Noce et al., 2014; Tao et al., 2018). Initially, authors followed a so-called keyword- or rule-based approach, where the text is searched for forward-looking structures such as "will", "expect" or a number reflecting the years after the publication of the corresponding report. However, those have been criticized for their performance in terms of high number of false positives and lack of scalability (Li, 2010; Tao et al., 2018).

A solution, proposed by Tao et al. (2018) and Noce et al. (2014), is to train a machine learning (ML) classifier, capable of automatically labelling sentences as FLS or non-FLS. It essentially takes as input a sentence from the text and generates a prediction of whether it is FLS (class 1) or non-FLS (class 2). To train the model, it is necessary to obtain a labelled dataset where for each sentence the corresponding class is known. There are many possible classifiers that can be applied here, from a logistic regression to deep learning approaches (Tao et al., 2018).

In this paper, we build upon the work of Tao et al. (2018) by applying the state-of-the-art deep learning models in the NLP field (Devlin, Chang, Lee, & Toutanova, 2018). Those are based on a so-called Transformer-architecture (Vaswani et al., 2017) and some of its variants perform better than humans on language tasks (Wang et al., 2018). We focus on 10-K reports and use sentiment of FLS among other features for stock price prediction. 10-Ks are publicly available, contain mandatory text sections, have a predefined structure and format, and are subject to auditing. Thus, they are less noisy and more reliable than social media texts. Also, as compared to news, they contain more of "what might happen in the future" than of "what has happened" information (Kearney & Liu, 2014, p. 179). The paper is structured as follows: in the next section, we provide the necessary theoretical background, discuss the relevant literature and describe our contribution. In section 3, we present our methodology, followed by its evaluation in section 4. Finally, in section 5, we derive main conclusions and paths for future research.

2. Background and related work

In this section, we first introduce 10-K reports as our unit of analysis, followed by the relevant NLP and ML concepts. After that, we discuss the related work and derive our contribution.

2.1. 10-K reports

U.S. companies are required to file the 10-K annual report with the SEC at the end of their fiscal year. It contains 15 items dealing with various business topics. This paper examines Items 1A ("Risk Factors"), 3 ("Legal Proceedings"), 7 ("MD&A") and 7A ("Quantitative and Qualitative Disclosures about Market Risk"). Item 7 has been shown to contain FLS in previous research (Muslu et al., 2015). Items 1A, 3 and 7A deal with specific risk factors the company is (or might be) facing and therefore also potentially contain important FLS. Among other things, 10-K reports have been analyzed to predict stock prices (Hsieh & Hristova, 2022) or for the report's effects on brand value (Huang, Liu, & Xie, 2020).

2.1. Relevant NLP and ML concepts

As mentioned above, the field of NLP deals with the automated analysis of textual, unstructured data, mostly following a three-step approach in finance and accounting. Since, as opposed to humans, computers require structured data (i.e. tables and numbers), Step 1 transforms the text in a structured form that adequately represents the text meaning. Initially, here a so-called bag-of-words (BOW) approach was applied, which essentially counts the occurrences of single words in a text and assumes that the more often a word appears, the more important it is. However, this approach does not consider the relationships between words and their context (e.g. "river bank" vs. "bank account"; "bad" as opposed to "good"), leading to a substantial loss of information.

To address this, embedding-based approaches were developed, where a word is represented in a continuous vector space and therefore its relationship to other words in the text can be determined. As opposed to BOW, here the structured representation of the word cannot be calculated solely from the text, but needs to be generated by a dedicated model. Most such models are based on neural networks (NN) (Mikolov, Chen, Corrado, & Dean, 2013; Peters et al., 2018) and the latest ones use a deep learning approach (Brown et al., 2020; Devlin et al., 2018). Deep NN consist of many layers of transformations. Thanks to this architecture, linguistic relationships between characters, words or sentences can be adequately modeled. One example for such a model are Long-Short-Term Memory (LSTM) NN (Peters et al., 2018). They use an extra memory cell helping remember previous information. LSTM can be bi-directional, meaning that both backward (i.e. before the word) and forward (i.e. after it) relationships are considered.

The latest development in the field is the Bidirectional Encoder Representations from Transformers (BERT) model published by Google (Devlin et al., 2018) and based on Transformers, which were designed to solve translation tasks (Vaswani et al., 2017). As opposed to LSTM, BERT uses the Transformer's attention mechanism to determine word relationships, leading to a much better performance. Additionally, the architecture allows for parallel computations, which is crucial with big amounts of data. BERT was pretrained on over 3 billion words by masking individual tokens in a sentence for the model to predict, thereby learning the dependencies *within* sentences (Masked Language Modelling). Furthermore, BERT learns relationships *between* sentences through Next Sentence Prediction, where for two sentences it has to determine if they follow each other. BERT achieved very strong performances on the GLUE benchmark (Wang et al., 2018) and has been a popular tool in NLP since its release (Rogers, Kovaleva, & Rumshisky, 2020). There are many different BERT versions that have been adapted to specific purposes. They are generally hosted at the HuggingFace Transformers library, which offers these pre-trained models for out-of-the-box usage or further fine-tuning (Wolf et al., 2019). This paper applies two of these pretrained models: DistillBERT and FinBERT. DistillBERT is a lighter version of BERT. The creators claim 97% of the original performance at 40% reduced size and 60% increased speed (Sanh, Debut, Chaumond, & Wolf, 2019). FinBERT is specifically adapted to financial texts. It was trained on 4.9 billion tokens from financial texts, including 10-K and 10-Q (i.e. published quarterly) reports. FinBERT has shown better performance than base BERT when used in a financial context (Yang et al., 2020).

Once the text is converted in a structured form, it can be processed by computers and used for decision-support. One of the most common such applications is classification, where the aim is to build a model capable of assigning a piece of text to one of a set of predefined classes (e.g. FLS vs. non-FLS; positive vs. negative sentiment). State-of-the-art models usually use the generated embeddings to predict the corresponding class. BERT for Sequence Classification, for instance, uses the BERT embeddings as an input to a classification layer, which consists of a fully connected NN and a softmax function for generating a probability distribution over the single classes (here FLS vs. non-FLS). Similarly, FinBERT embeddings are inputted to a classification layer to predict the sentiment of a sentence in financial context, possible classes being positive, negative and neutral one (Yang et al., 2020).

A classification model is a supervised model, implying that it has to be trained with labelled text data. For FinBERT, this was done using publicly available labelled datasets of financial news and analyst reports. Therefore, the model can be used out-of-the-box for our analysis. Contrary to that, we do not have labelled data for the FLS classification and thus need to generate it ourselves. This is one of the major challenges in the field, as seen in the discussion of the related work in the next section.

2.3. Related work

As mentioned above, initially researchers used keyword/rule-based approach to extract FLS from financial texts and later combined that with ML classification. Both Muslu et al. (2015) and Li (2010) analyze the MD&A Item of 10-K reports using a keyword/rule-based approach for their identification. Li (2010) applies a list of 15 keywords such as "will", "expect", "anticipate" and "forecast". Additionally, the results are cleaned for non-FLS, using phrases such as "was expected". Muslu et al. (2015) use three rules for the identification of FLS: either a sentence 1) contains certain forward-looking keywords, similar to Li (2010), or 2) a "conjunction of verbs that imply future (such as "the company plans to…")" (p. 11), or 3) a number after the filling year (e.g. 2023). In this manner, Muslu et al. (2015) label on average 12.5% of the statements in an MD&A Item as FLS. To validate their approach, Muslu et al. (2015) ask students to manually label the sentences of 50 MD&A items and compare the result with the automated output. They find that out of the 25.5 sentences on average labelled as FLS, the algorithm captured 18.1 (about 82%). This is not a bad performance, but could possibly be improved by text structuring and the application of ML classification. In addition, the results are expected to suffer from high level of false positives (Tao et al., 2018). Also, the approach could be resource-intensive for longer texts and datasets with many reports.

The above points are addressed by Noce et al. (2014) who apply a BOW approach for structuring earning call transcripts and use the result as input to a Support Vector Machine (SVM) model to classify sentences as FLS or non-FLS. The SVM is trained on 2092 manually labelled sentences and achieve an accuracy of 87.57% on a test set of 1046 sentences.

Tao et al. (2018) achieve even higher accuracies by employing more advanced ML models, including an LSTM model, when extracting FLS from IPO prospectuses. They use a training set of over 40,000 labeled FLS and non-FLS and achieve the highest accuracy of 96.74% with the LSTM model, accrediting its consideration of word ordering and contextual information for this success. To decrease the labelling effort, Tao et al. (2018) first apply a rule-based approach similar to Muslu et al. (2015) to extract a set of FLS statements. Those are then manually reviewed and balanced in terms of FLS and non-FLS. The resulting set is used for training. For text structuring with embeddings, Tao et al. (2018) apply the word2vec Skip-gram model (Mikolov et al., 2013).

After the identification of the FLS statements in financial texts, existing works apply Steps 1 to 3 to those statements. Li (2010) uses a BOW approach (Step 1) to build a classifier for predicting the sentiment and topic of FLS statements (Step 2). He regresses the result on important financial indicators, such as current earnings, stock returns, size, accruals, return/earning volatility (Step 3) and finds a positive association between the average sentiment of FLS and future earnings and liquidity. Muslu et al. (2015) focus on the number instead of the sentiment (Step 2) and conduct a similar regression analysis as Li (2010) (Step 3). They show that generally, many FLS lead to a higher discrepancy between the stock returns and future earnings, but FLS could be beneficial for companies with little information disclosure (so-called dark firms). Tao et al. (2018) derive a set of features such as top five topics, average sentiment, semantic similarity and readability (Step 2) and input those to multiple classification models for IPO price prediction (Step 3). They find that FLS improve prediction for pre-IPO price revisions when used in combination with traditional IPO characteristics, but generate no benefit when used to predict post-IPO first day pricing.

2.4. Research contribution

Existing works already achieved impressive results. However, the state-of-the-art NLP developments substantially improved the performance for similar automated text analysis tasks. In this paper, we contribute to the literature both by applying those NLP models and using different data as follows:

1. *Data:* We focus on 10-K reports, similar to Muslu et al. (2015) and Li (2010), but also extend the analysis, by considering Items 1A, 3, and 7A in addition to Item 7.

2. *Model:* We extend the work by Tao et al. (2018) by applying DistilBERT (instead of word2vec) for text structuring and BERT for Sequence Classification for subsequent identification of FLS (instead of standard classification models).

3. *Model:* The identified FLS are processed for sentiment analysis in Step 1 and Step 2 with FinBERT as opposed to SentiWordNet in Tao et al. (2018).

4. *Model+ Data:* The sentiment is used as an input to a Random Forest (RF) model for stock price growth prediction as opposed to regression models in Muslu et al. (2015) and Li (2010). Tao et al. (2018) applies a RF model, but predicts the IPO prospectuses and IPO pricing. In the next section, we present our approach.

3. Methodology

As mentioned above, our methodology follows the approach by Tao et al. (2018) by first identifying FLS in the text (we call this Step 0 to avoid confusion) and then applying Steps 1 to 3 to the FLS parts (see Figure 1). The text here consist of Items 1A, 3, 7 and 7A of 10-K reports. In Step 0, we combine the rule-based approach by Muslu et al. (2015) with manual labelling and ML-classification, resulting in three classification models for FLS identification. The models are then applied to all potential FLS from the rules, generating the final FLS. Those are the inputs to Steps 1 and 2, where we determine the FLS sentiment in each report using FinBERT. Finally, in Step 3, this sentiment together with the number of FLS per item is used to train a RF model to predict stock price growth for periods of one day, one week, one month, six months, and one year after publication.

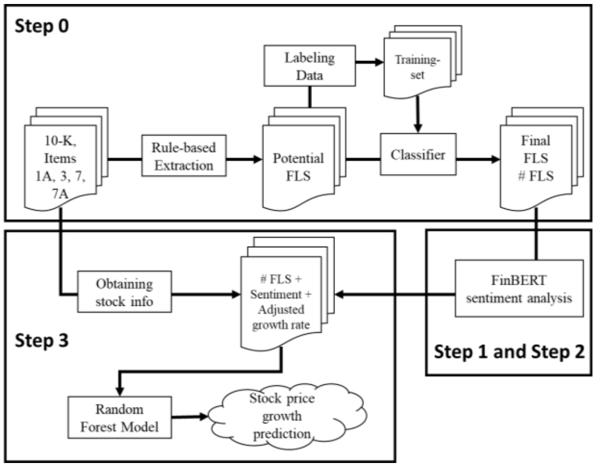

Figure 1: Methodology.

3.1 Step 0: FLS identification

In this step, we filter the FLS from the text. To achieve this, we first use a rule-based approach to generate the potential FLS. The aim here is to reduce the enormous effort required for manually labelling every single sentence in the text. We search for the following three patterns:

1. Combinations of the words "next", "subsequent", "following", "upcoming", "incoming",

"coming", "succeeding", "carryforward" with time-period indicators "month", "quarter", "year", "fiscal", "taxable", "period".

2. The words "aim", "anticipate", "assume", "commit", "estimate", "expect", "forecast", "foresee", "hope", "intend", "plan", "predict", "project", "seek" and "target" used as a verb (to differentiate from their nouns that are not necessarily used with temporal meaning, e.g. “a project”).

3. Years (defined as 4-digit numbers starting with a “1” or “2”) greater than the year of publication.

A sentence is a potential FLS, if it contains any of the above patterns. A random sample of the potential FLS is then manually reviewed and sentences labeled as FLS or non-FLS. As FLS, we consider only sentences that contain precise information (e.g. “we expect this trend to continue”), while uncertain sentences without indication of likelihood are non-FLS (e.g. “sales may decrease if we fail to deliver on our promises”). We use this labeled dataset to train three ML classifiers: a SVM model (as in Noce et al. (2014)), a LSTM model (as in Tao et al. (2018)) and a BERT for Sequence Classification model based on DistilBERT, reflecting state-of-the-art developments. The trained models are then applied to all potential FLS in the report and therefore the set of final FLS is generated. We also use only the potential FLS for model training. In section 4.2, we additionally analyze this point. Since the three classification models do not generate exactly the same prediction, this results in three FLS datasets, one for each classification model. Based on Muslu et al. (2015), we additionally calculate the number of FLS per item and dataset and consider them in Step 3.

3.2 Steps 1 and 2: Sentiment analysis

The sentiment of the FLS from Step 0 is then calculated using FinBERT for sentiment analysis. For each sentence in a report, the model calculates logit scores, later transformed with a softmax function to a probability distribution over the classes of positive, negative and neutral sentiment. A sentence is assigned to the class with the highest score corresponding to the highest probability. In our approach, we use the logit score instead of the final class. The reason is that, for instance, a distribution of 34% positive, 33% neutral and 33% negative would result in a positive class, even though the distance to the other two is very small. Thus, for each sentence, each sentiment class, each report and each FLS dataset, we obtain the corresponding score from the model. To aggregate that at a report level, we calculate the score mean for each sentiment class over all FLS in the report (Hsieh & Hristova, 2022) resulting in *mean_positive*, *mean_neutral* and *mean_negative*. In order to be able to represent the report with a single sentiment value, we define the variable *sentiment_score* as *mean_positive-mean_negative*. There are several reasons for this approach. First, the literature has shown that the reports contain a high number of neutral statements (Hsieh & Hristova, 2022). Thus, including the neutral scores in the final variable would result in mostly neutral *sentiment_score* not providing much information. Second, the idea to substract the score for the negative values from the positive ones has also been presented by the authors of FinBERT (Prosus AI, 2022). It can be interpreted as follows: with a maximum positive score (i.e. probability of 1 for positive), *sentiment_score* takes the maximum positive value*;* with a maximum negative score it takes the minimum negative value, and with positive and negative having the same probability (e.g. 0.5), it takes a value of zero. Thus, it can be easily interpreted. The result of this step is the *sentiment_score* for each report and each of the three FLS datasets.

3.3 Step 3: Stock price growth prediction

In order to examine the contribution of FLS to decision making, we build a prediction model that uses the results from the previous steps to determine the adjusted growth rate of the company. This is defined as the relative stock growth rate for the given period minus the relative growth rate of the S&P 500 Index for the same period. The S&P Index tracks the performance of the 500 largest publicly traded U.S. companies, which are also obliged to publish 10-K reports. We consider the relative stock growth rate for better comparability between companies and subtract the S&P 500 Index to incorporate general economic effects. Since it is not clear for which time span FLS can affect the adjusted growth rate, we calculate this for periods of one day, one week, one month, six months and one year after publication. These growth rates serve as the target variable for multiple RF models (one for each period and FLS dataset).

A RF model consists, as the name says, of a collection of decision trees (DT). A DT contains nodes and branches, where the nodes represent independent variables (here *sentiment_score* and number of FLS) and the branches stand for decision boundaries on their values. In this way, based on the values of the independent variables, the value of the target variable (here adjusted growth rate) can be determined. A RF model has been successfully applied in the past in similar contexts (Hsieh & Hristova, 2022; Huang et al., 2020) and is known for its feature interpretability, helping better understand the reasons behind the predictions. This completes the description of our methodology. In the next section, we evaluate it on a 10-K dataset based on S&P 500 companies.

4. Evaluation

4.1 Dataset

In order to apply our approach, we need to combine 10-K reports for the FLS identification in Step 0 with their corresponding stock prices for the stock price growth prediction in Step 3. 10-K reports were obtained through the SEC EDGAR database. Overall, 10-Ks from 80 randomly chosen S&P 500 companies between the years 2006-2020 are included, resulting in 1030 reports. The timespan starts with 2006 as previous years contain differences in the structure. 2020 is chosen as the cutoff point in order not to include the economic effects of the COVID-19 pandemic. To restrict the reports to Items 1A, 3, 7 and 7A, their titles are matched using regular expressions.

Additionally, for Step 3 we need the adjusted growth rate for the companies corresponding to the extracted 10-K reports. We downloaded stock data from Yahoo! Finance using the opening price at the day of publication as the baseline for the relative and consequently adjusted growth rate. Based on this, we calculated the adjusted growth rate for one day, one week, one month, six months and one year after publication, which is the target variable in Step 3. Additionally, we determined the adjusted growth rate for the same periods *before* publication (variables *p_adj_grwth_xxx* with *xxx* being *d, w, m, 6m* and *y* correspondingly*)* as well as the average daily trading volume over the past year *before* publication (variable *volume*). The reason for calculating those is to later compare the prediction model based on FLS variables with the one based on historical stock data, representing more traditional approaches.

4.2 Step 0: FLS identification

The aim of Step 0 is to apply the rule-based approach to the 10-K reports, followed by manual labelling and training of the three classifiers. For this, we search for the patterns in 3.1 using the spaCy package in Python. Some of the reports returned no matches for the rule-based extraction and were therefore dropped. Additionally, the first results after the manual labelling generated an unbalanced dataset with 76% non-FLS and 24% FLS. To avoid non-FLS bias in the classifiers, this dataset was extended to 856 labeled sentences with a 50%/50% split between FLS and non-FLS (cf. Tao et al. (2018)). These were then used to train the three classifiers, where for the SVM we use a BOW structuring approach and for LSTM and BERT the corresponding embeddings. The performance is measured using another human-labeled validation set of 220 previously unseen sentences. 76 of the 220 sentences are FLS, the remaining 144 are non-FLS. The dataset is purposefully left unbalanced to better represent the output of the rule-based approach. On this validation set, the SVM model achieved an overall accuracy of 0.80, the LSTM model of 0.75 and the BERT model of 0.91. Tables 1, 2 and 3 show additional performance metrics.

Table 1. Classification results SVM.

SVM	**Precision**	**Recall**	**F1-Score**
False	0.90	0.78	0.84
True	0.67	0.84	0.74

Table 2. Classification results LSTM.

LSTM	**Precision**	**Recall**	**F1-Score**
False	0.85	0.75	0.80
True	0.61	0.75	0.67

Table 3. Classification results BERT.

BERT	**Precision**	**Recall**	**F1-Score**
False	0.98	0.88	0.93
True	0.81	0.96	0.88

We can see that the BERT model outperforms the other models in every regard, making it the best tool for FLS-classification. Furthermore, as DistilBERT trades performance in favor of faster speed and smaller size, even better results could possibly be achieved with other variants such as BERT Large (Devlin et al., 2018). However, it should be noted that the BERT model was also the slowest of the three, requiring roughly twice the processing time of the other models to go through all reports.

After applying the trained models to the initial dataset of 1030 reports, the SVM model identified 705 reports with FLS, the LSTM model 719 and the DistilBERT model 709. The difference in numbers between the three datasets is due to the models performing differently as the tables above show. To ensure comparability, only reports are kept that are contained in all three datasets, resulting in 700 entries each, from 72 different companies. The mean number of FLS per report extracted by each classifier are: 36.81 (LSTM), 34.49 (SVM) and 29.62 (BERT). The numbers demonstrate that on average, LSTM classifies more statements as FLS than the other models, with BERT being the most conservative with the FLS class.

Figure 2 shows the distribution of FLS between the 10-K items. Item 7 contains the most FLS, which is in-line with previous research. However, when combined, Items 1A, 3, and 7A provide more than 50% additional FLSs. Therefore, these items should not be neglected in future research into this topic.

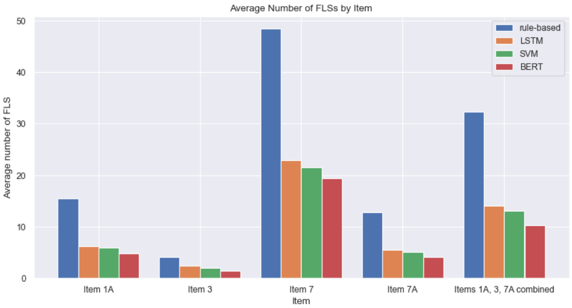

Figure 2. Distribution of FLSs between Items.

Differences can also be found between companies. Some companies' reports contain significantly more FLS than others. For example, the maximum number of FLS are found in the 2018 report of DuPont de Nemours. DuPont also has the highest average number of FLSs with 169.5 (2 reports of DuPont are contained in the dataset), followed by The Coca-Cola Company with 112.09 (11 reports) and Tesla with 99.33 (6 reports). On the lower end of the spectrum are Archer Daniels Midland (2 reports) and Clorox (7 reports), both with an average of 1 FLS, followed by Pepsi with 2.625 (8 reports) (averages taken from the LSTM dataset).

As mentioned in 2.3, using only the rule-based approach without the classifier could result in a large number of false positives. This is already visible in the distribution of the manually labelled classes. To examine the issue further, we analyze a randomly chosen 10-K reduced to the relevant items, as discussed above. We label each sentence manually for FLS vs. non-FLS, resulting in 49 FLS out of 627.

Table 4. Performance of rule-based extraction.

Rule-based	**Precision**	**Recall**	**F1-Score**
False	0.97	0.93	0.95
True	0.47	0.71	0.56
Accuracy			**0.91**

The rule-based approach classified 75 sentences as FLSs, 40 of which were false positives with the remaining 35 being true positives. Based on these numbers, we derive the performance metrics in Table 4. It shows that while the precision for the FLS class is low (0.47), the precision and recall of the false label is comparatively high (0.97 and 0.93), mainly due to the high number of non-FLS contained in the 10-K. This confirms the results in the literature and speaks for the use of classifiers in addition to a rule-based approached.

Additionally, as mentioned above, we apply the trained classifiers only to the potential FLS and not to the whole text. To examine the consequences of this decision, we applied the three models to all sentences of the above report. The SVM classified 175 of the sentences as FLS, the LSTM 268 and the BERT model 229. This is in all cases much higher than the 49 representing the true number. Therefore, the rule-based approach serves as a useful first step by reducing noise for the classification models.

The result of Step 0 are three datasets containing the FLS statements for all reports and each classifier. Additionally, the number of FLS per report, item and classifier is calculated and represented by the variables *# FLS Item xxx*, with *xxx* being *1A*, *3*, *7* and *7A.*

4.3 Steps 1 and 2: Sentiment analysis

In Steps 1 and 2, we apply FinBERT to each of the three datasets. We also remove outliers, based on *sentiment_score*. The result is a *sentiment_score* for each of the reports and each of the three datasets.

4.4 Step 3: Stock price growth prediction

In this step, we train a RF model to predict the adjusted growth rate for different periods after publication based on the FLS variables above (i.e. *sentiment_score* and *# FLS Item xxx*). In a data-cleaning step, we remove the adjusted growth rate outliers based on their 5% and 95% quantiles. Thus, 490 reports remain in each of the three datasets. In order to determine the contribution of FLS information to the adjusted growth rate, we follow a similar approach as in Tao et al. (2018), where different configurations of input variables are compared to each other. We consider the following configurations:

a. FLS: input variables are *sentiment_score* and *# FLS Item xxx*, where *xxx* stands for *1A, 3, 7* and *7A*

b. Stock data: input variables are *p_adj_grwth_xxx* (*xxx* = *d*, *w*, *m*, *6m* and *y)* and *volume*

c. FLS+ Stock data

Here, b. determines adjusted growth rate solely based on historical stock data, whereas a. considers only the extracted FLS variables. c. combines both of them. Our aim is to determine the contribution of our approach against more traditional methods based on historical stock data.

For each dataset, each period as a target variable and each configuration, we train a RF model using grid search cross validation. This results in 5 (for the 5 future periods as a target variable) times 7 (3 datasets each for a. and c.; 1 dataset for b.) models. The models are trained with a 90/10 train/test split. We use the following parameters for the grid search:

- *max_features*: [2, 4, 5] for configurations a. and b., [2, 4, 6, 8, 10] for configuration c.
- *n_estimators*: [50, 100, 200, 400]
- *min_samples_leaf*: [1, 3, 5]
- *max_depth*: [None, 2, 3, 4]

Any parameter that is not mentioned is set to its standard value from the scikit-learn implementation. Also note that a. has only five features. Performance is measured with the Relative squared error (RSE). RSE measures the squared error of a prediction compared to the squared error of a "lazy predictor", which always predicts the average of the target variable. A RSE above one means that the prediction is worse than the lazy predictor, below one means it is better. The RSE scores for each model can be found in Table 5, with the best RSE for each period marked in bold and the overall best one marked in red.

Table 5. RSE of RF Predictions (1=SVM, 2=LSTM, 3=BERT FLS Dataset).

Configuration	FLS Dataset	One day	One week	One month	Six month	One year
a.	1.	1.020	1.068	1.011	1.103	1.009
	2.	1.000	1.008	1.132	1.062	1.050
	3.	0.979	0.986	**1.004**	1.096	1.034
b.		0.999	1.064	1.021	**1.024**	**0.992**
c.	1.	1.010	1.064	1.005	1.077	1.026
	2.	**0.944**	**0.974**	1.113	1.041	1.090
	3.	1.008	1.020	1.006	1.074	1.036

The best overall RSE is achieved by the model trained on the LSTM Dataset (2.) using both FLS and stock variables (configuration c.) when predicting the adjusted growth rate one day after publication. The same combination shows the best results for the prediction of one-week growth rates. In both cases, it is narrowly followed by the BERT Dataset (3.) and FLS only (a.). This confirms the approach by Tao et al. (2018) and implies that for short-term performance a combination of historical stock data and FLS provides the best prediction basis. Since the RSE is lower than for b., it demonstrates the relevance of extracted FLS. For mid-term prediction (i.e. one month), the best performance is delivered by the BERT Dataset (3.) and using only FLS variables (a.). This, together with the fact that the BERT Dataset shows the best results in configuration a. in 5 out of 6 cases, demonstrates the potential of state-of-the-art NLP approaches in the area. Finally, for long-term prediction only historical stock data play a role (b.). This could be because in the long term, FLS are not forward-looking anymore, but could be verified based on the current company performance. Note that some of the models have an RSE above one, meaning that they were outperformed by the lazy predictor. This could be due to the size of the dataset, the chosen RF model or the sentiment variables as discussed in the next section.

As mentioned above, we chose RF for Step 3 also because it allows for an easy derivation of the most important features for the prediction. To demonstrate that, Table 6 shows the top 10 most important features for the best model overall (LSTM Dataset with configuration c. on one day) and the best model for one month (i.e. BERT Dataset with configuration a.). The results show that for short-term prediction (i.e. one day) the historical stock data is more important than the FLS variables. In addition, the sentiment of the FLS is more important than their number, where the order changes for mid-term predictions. Moreover, the number of FLS in Item 7 (MD&A) has in both cases lower importance than the number of FLS in other items. This confirms our approach, which as opposed to existing works, also uses other items for FLS identification.

Table 6. Feature importance for LSTM + c. and one day and BERT + b. and one month.

Rank	LSTM+ c.	BERT + a.
1	*p_adj_grwth_d*	*# FLS Item 3*
2	*p_adj_grwth_w*	*# FLS Item 1A*
3	*p_adj_grwth_6m*	*# FLS Item 7A*
4	*p_adj_grwth_y*	*sentiment_score*
5	*volume*	*# FLS Item 7*
6	*p_adj_grwth_m*	
7	*sentiment_score*	
8	*# FLS Item 1A*	
9	*# FLS Item 7*	
10	*# FLS Item 7A*	

To sum up, in this section, we applied our methodology to a dataset of 10-K statements to extract FLS and used those to predict adjusted growth rates for different periods. We additionally compared the results with a model based solely on historical stock data and models considering both FLS and historical stock data. Our results show that by using state-of-the-art NLP approaches for FLS identification (here DistillBERT), we could improve the performance as compared to models from previous studies (i.e. SVM and LTSM). In addition, when predicting adjusted growth rates, the best performance for short-term periods was achieved by the LSTM Dataset using both stock and FLS variables, followed by the BERT Dataset with solely FLS variables. In the mid-term, the latter outperformed the rest and in the long-term only historical stock data played a role. Finally, Item 7, which is solely used in existing research, has one of the lowest importance for prediction, supporting the use of additional items as we do here. In the next section, we draw some main conclusions and provide paths for future research.

5. Conclusion

In this paper, we develop a methodology for the extraction of FLS from 10-K reports and use those for stock price growth prediction. Similar to existing works, we use Item 7 of the 10-K report and extend the analysis by additionally considering Items 1A, 3, and 7A. Our methodology represents a four-step approach based on the literature. In Step 0, we identify the FLS with a combination of a rule-based approach, manual labelling and ML classification models. In particular, we extend the existing works by applying the state-of-the-art NLP model DistillBERT and comparing it with models from the literature (SVM and LSTM). This generates three datasets of FLS for each report, depending on the used model. In Steps 1 and 2, we calculate the sentiment of the identified FLS in a given report using FinBERT. This is also a state-of-the-art NLP model trained on financial texts that was not used in this setting in the literature before. The sentiment, together with the number of FLS per item are used in Step 3 in a RF model to predict stock price growth rate for the period of one day, one week, one month, six months and one year after report publication.

We evaluate our approach on a set of 10-Ks representing randomly chosen S&P 500 companies in the period 2006-2020. In order to measure its contribution, we define three configurations for the final RF model: a. FLS variables, b. historical stock data, and c. both FLS variables and historical stock data. b. represents more traditional approaches for stock price growth prediction, whereas a. stands for our approach. The results generate the following practical implications: 1) DistillBERT outperforms SVM and LSTM and thus should be used for FLS identification. The extracted statements can then later be additionally analyzed to support investment decisions for instance by manual analysis or using topic modelling. 2) The analysis should not be limited to Item 7, as the other items considered here provide more than 50% of the FLS and have higher importance for stock price growth rate prediction. 3) In the short-run, we could not derive a clear recommendation. The best performance is with historical stock data and FLS extracted with the LSTM model. However, it is narrowly followed by the BERT model and the use of FLS only, in line with 1). Thus, future research should examine this point. 4) For mid-term predictions, solely FLS should be used and extracted with DistillBERT. Generally, for a., the BERT model shows the best results in 5 out of 6 cases. 5) In the long term, FLS information does not seem to provide added-value and should thus not be extracted.

However, we note that some of the RSE values are above one, which makes the models worse than a lazy classifier. This could be due to a number of reasons, which can be the focus of future research. First, due to resource restrictions, we trained the classification models with a comparatively small number of labelled statements, which could influence performance. Thus, in the future, more data could be manually labeled and more reports extracted. Second, we used DestilBERT as a lighter version of BERT for FLS identification. Here, other state-of-the-art models such as BERT Large or T5 could be applied instead. Third, we focused on one sentiment variable representing the FLS in a given report. Using other sentiment variables instead could potentially improve results. Fourth, we chose a RF model for prediction due to its interpretability. Alternatives, based on NN could improve performance, but are also black-box models. Last, in addition to 10-K reports, other sources, such as news, can be considered to improve the short-term prediction performance.

6. References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 1877–1901). Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018, October 11). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Retrieved from http://arxiv.org/pdf/1810.04805v2

Dyer, T., Lang, M., & Stice-Lawrence, L. (2017). The evolution of 10-K textual disclosure: Evidence from Latent Dirichlet Allocation. *Journal of Accounting and Economics*, *64*(2), 221–245. https://doi.org/10.1016/j.jacceco.2017.07.002

Fisher, I. E., Garnsey, M. R., & Hughes, M. E. (2016). Natural Language Processing in Accounting, Auditing and Finance: A Synthesis of the Literature with a Roadmap for Future Research. *Intelligent Systems in Accounting, Finance and Management*, *23*(3), 157–214. https://doi.org/10.1002/isaf.1386

Hsieh, H.-T., & Hristova, D. (2022). Transformer-based Summarization and Sentiment Analysis of SEC 10-K Annual Reports for Company Performance Prediction. In *Proceedings of the 55th Hawaii International Conference on System Sciences (HICSS)*, Hawaii.

Huang, C.-Y., Liu, P.-Y., & Xie, S.-M. (2020). Predicting brand equity by text-analyzing annual reports. *International Journal of Market Research*, *62*(3), 300–313. https://doi.org/10.1177/1470785319883201

Kearney, C., & Liu, S. (2014). Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, *33*, 171–185. https://doi.org/10.1016/j.irfa.2014.02.006

Kraus, M., & Feuerriegel, S. (2017). Decision support from financial disclosures with deep neural networks and transfer learning. *Decision Support Systems*, *104*, 38–48. https://doi.org/10.1016/j.dss.2017.10.001

Krinitz, J., & Neumann, D. (2021). Decision Analytics for Initial Public Offerings: How Filing Sentiment Influences Stock Market Returns. In H. Gimpel, J. Krämer, D. Neumann, J. Pfeiffer, S. Seifert, T. Teubner, . . . A. Weidlich (Eds.), *Market Engineering* (pp. 45–67). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-66661-3_3

Lewis, C., & Young, S. (2019). Fad or future? Automated analysis of financial text and its implications for corporate reporting. *Accounting and Business Research*, *49*(5), 587–615. https://doi.org/10.1080/00014788.2019.1611730

Li, F. (2010). The Information Content of Forward-Looking Statements in Corporate Filings-A Naïve Bayesian Machine Learning Approach. *Journal of Accounting Research*, *48*(5), 1049–1102. https://doi.org/10.1111/j.1475-679X.2010.00382.x

Loughran, T., & McDonald, B. (2011). When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x

Masson, C., & Paroubek, P. (2020). Nlp Analytics in Finance with DoRe: A French 250M Tokens Corpus of Corporate Annual Reports. In *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 2261–2267). Marseille, France: European Language Resources Association. Retrieved from https://aclanthology.org/2020.lrec-1.275

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *ArXiv Preprint ArXiv:1301.3781*.

Muslu, V., Radhakrishnan, S., Subramanyam, K. R., & Lim, D. (2015). Forward-Looking MD&A Disclosures and the Information Environment. *Management Science*, *61*(5), 931–948. https://doi.org/10.1287/mnsc.2014.1921

Noce, L., Zamberletti, A., Gallo, I., Piccoli, G., & Rodriguez, J. A. (2014). Automatic Prediction of Future Business Conditions. In A. Przepiórkowski & M. Ogrodniczuk (Eds.), *Lecture Notes in Computer Science. Advances in Natural Language Processing* (Vol. 8686, pp. 371–383). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-10888-9_37

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* (pp. 2227–2237). New Orleans, Louisiana: Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-1202

Prosus AI (2022). FinBERT: Financial Sentiment Analysis with BERT [Jupyter Notebook]. Retrieved from https://github.com/ProsusAI/finBERT

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, *8*, 842–866. https://doi.org/10.1162/tacl_a_00349

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019, October 2). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. Retrieved from http://arxiv.org/pdf/1910.01108v4

SEC (2003). *Financial Reporting Release No. 72. Commission Guidance on Management's Discussion and Analysis of Financial Condition and Results of Operations*. Securities and Exchange Commission (SEC): Securities and Exchange Commission (SEC).

Tao, J., Deokar, A. V., & Deshmukh, A. (2018). Analysing forward-looking statements in initial public offering prospectuses: a text analytics approach. *Journal of Business Analytics*, *1*(1), 54–70. https://doi.org/10.1080/2573234X.2018.1507604

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. (2018). GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In T. Linzen, G. Chrupała, & A. Alishahi (Eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP* (pp. 353–355). Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-5446

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., . . . Rush, A. M. (2019, October 9). *HuggingFace's Transformers: State-of-the-art Natural Language Processing*. Retrieved from http://arxiv.org/pdf/1910.03771v5

Yang, Y., UY, M. C. S., & Huang, A. (2020, June 15). *FinBERT: A Pretrained Language Model for Financial Communications*. Retrieved from http://arxiv.org/pdf/2006.08097v2