Financial Sentiment Analysis:
An Investigation into Common Mistakes and Silver Bullets

Frank Z. Xing[†], **Lorenzo Malandri**[‡], **Yue Zhang**[*], **Erik Cambria**[†]
[†]Nanyang Technological University, Singapore
[‡]University of Milano-Bicocca, Italy [*]Westlake University, China
{zhutian.xing, cambria}@ntu.edu.sg
lorenzo.malandri@unimib.it, zhangyue@westlake.edu.cn

Abstract

The recent dominance of machine learning-based natural language processing methods has fostered the culture of overemphasizing model accuracies rather than studying the reasons behind their errors. Interpretability, however, is a critical requirement for many downstream AI and NLP applications, e.g., in finance, healthcare, and autonomous driving. This study, instead of proposing any "new model", investigates the error patterns of some widely acknowledged sentiment analysis methods in the finance domain. We discover that (1) those methods belonging to the same clusters are prone to similar error patterns, and (2) there are six types of linguistic features that are pervasive in the common errors. These findings provide important clues and practical considerations for improving sentiment analysis models for financial applications.

1 Introduction

Natural language processing has been widely used for financial applications in recent years. These applications include stock market / foreign exchange market prediction, volatility modeling, asset allocation, business taxonomy construction, credit scoring, initial public offering valuation (IPO), and more (Ding et al., 2015; Xing et al., 2018a; Bai et al., 2019; Xing et al., 2019; Yang et al., 2020; Xing et al., 2021). Among them, there are mainly two threads of method to process textual inputs: the first is to directly encode financial texts by neural nets and to use the representations for learning downstream tasks (Xu and Cohen, 2018); the second is to analyze financial texts with critical linguistic features such as content semantics (Keith and Stent, 2019) or investors' sentiment (Malandri et al., 2018), for the sake of interpretability.

The objective of financial sentiment analysis (FSA) is to classify a piece of financial text as expressing bullish or bearish opinions toward certain arguments. Although sentiment analysis in the general domain is extensively studied in the past decades (Cambria et al., 2013), FSA is a challenging task because of the lack of large-scale training data and the difficulty in labeling after acquiring the texts, which involves expert knowledge. As a result, the model performance for FSA are usually significantly worse than using the same sentiment analysis model for the general domain (which is referred to as a problem of domain adaptation).

In addition to the above-mentioned challenges, FSA requires more interpretability comparing to sentiment analysis in other domains (Luo et al., 2018; Sedinkina et al., 2019). Early sentiment analysis studies leveraged on textual data from movie reviews, product reviews, and social media posts. In these applications, the purpose being roughly understanding customer feedbacks, statistically aggregating straightforward opinions suffices and a single mistake does not make much difference. Whereas for financial applications, a fraction of bad sentiment analysis results may cause extreme loss, hence have to be carefully treated with exception. Therefore, the goal of FSA is more than obtaining a high accuracy number: understanding when and why the method would fail is equally important.

In this paper, we explore the FSA behavior of some of the most common sentiment analysis models and the interpretability problem by using error visualization and linguistic analysis. Unlike many of

Proceedings of the 28th International Conference on Computational Linguistics, pages 978–987
Barcelona, Spain (Online), December 8-13, 2020

the domain adaptation-based FSA effort where only distant supervision is available (Felbo et al., 2017), we resort to *self-labeling*. Noticing that some financial social platforms allow users to simultaneously label their post as bullish (positive) or bearish (negative), such platforms make it possible for us to collect financial texts with high-quality sentiment labels in a “crowd-sourced” manner. We also compare performance of the same models on both the finance domain (StockSen) and the general business review domain (Yelp).

We are specifically interested in the following research questions.

> **RQ1**: Do different sentiment analysis methods step into the same pitfalls (i.e., make the common errors) or each of them makes errors on different examples of financial texts?
>
> **RQ2**: How reliable and consistent are the usually reported sentiment analysis metrics, i.e., F-score and accuracy measure, in evaluating methods across datasets from different language domains?
>
> **RQ3**: Is the performance deterioration for financial texts due to the same reasons as those found in other language domains? If so, the problem is expected to be significantly mitigated with common domain adapted methods. Otherwise, we suspect that to understand financial sentiment, more *resources* (jargon, time, complex reference) are required than in other domains, such that domain adaptation alone cannot surmount the bottleneck.

The major contributions of this paper are as follows: (1) we show that the same sentiment analysis methods usually perform worse in the finance domain; (2) by evaluating three lexicon-based models and five machine learning-based models on the sentiment classification task on two datasets, we show that the machine learning-based models, e.g., SVM and BERT, usually make more false positive errors than false negatives; (3) we describe six error types which contribute to the understanding of linguistic features of tweets from the finance domain; (4) we introduce a new corpus (StockSen) for the FSA task.

2 Compared Models for the FSA Task

Sentiment analysis models may be trained from label supervisions, leverage external knowledge about word polarities, or been a hybrid of those techniques (Ye et al., 2018). To have a good coverage of different types, we choose to investigate eight representative models from three clusters, i.e., lexicon-based (OpinionLex, SenticNet, and L&M), machine learning-based (SVM and fastText), and deep learning NLP models (bi-LSTM, S-LSTM, and BERT) as elaborated below.

Formally, each financial text T consists of k sentences $S_1, S_2, ..., S_k$, where S_i is a word sequence $w_{i1}, w_{i2}, ..., w_{it}$. The FSA model outputs an average of binary sentence polarities: $p(T) = \texttt{avg}(p(S_i))$.

1. **OpinionLex**: a list of around 6,800 sentiment-carrying words collected by Hu and Liu (2004). This model simply classifies a sentence as positive if positive words occur more times than negative words and vice versa.

2. **SenticNet**: a general-purpose sentiment knowledge base (Cambria et al., 2020) that contains not only word entries, but also phrases and multi-word concepts. We use it in couple with a group of dependency grammar patterns to determine the polarity of each sentence.

3. **L&M dictionary**: a lexicon that is manually cast for analyzing financial texts by Loughran and McDonald (2011). The L&M dictionary is used the same way as per OpinionLex.

4. **SVM**: a robust and classical model for binary text classification. We only use term frequency-inverse document frequency (TF-IDF) features, i.e., the input is a TF-IDF vector of the token-size length. $p(S_i) = \texttt{SVM}(\text{TF-IDF}^{||w||})$.

5. **fastText**: a CBOW-like model that also considers sub-word information (n-gram features), hence the token number increases to predict a categorical output. The model is claimed to be on par with deep learning models for text classification (Joulin et al., 2017).

6. **bi-LSTM**: a model for representation learning that concatenates LSTM hidden states from both directions of a sentence to mitigate the problem of memory decay (Schuster and Paliwal, 1997). bi-LSTM has a forward and a backward component. For the forward component,

$$\begin{aligned} i_t &= \sigma(W_i \cdot [\, \overrightarrow{h}_{t-1}, x_t \,] + b_i) \\ f_t &= \sigma(W_f \cdot [\, \overrightarrow{h}_{t-1}, x_t \,] + b_f) \\ o_t &= \sigma(W_o \cdot [\, \overrightarrow{h}_{t-1}, x_t \,] + b_o) \\ c_t &= f_t \odot c_{t-1} + i_t \odot (W_c \cdot [\, \overrightarrow{h}_{t-1}, x_t \,] + b_c) \\ \overrightarrow{h}_{t-1} &= o_t \odot \tanh(c_{t-1}), \end{aligned} \tag{1}$$

where i_t, o_t, f_t, c_t are values of the input gate, output gate, forget gate, and cell state respectively; σ denotes the sigmoid function; x_t is the word embedding for word w_{it}; $\overrightarrow{h}_{t-1}$ is the hidden state of the previous time step $t-1$; $W_{()}$ denotes the state transfer matrices and $b_{()}$ is the bias. A different set of parameters are used for the backward component, which reads the sentence from w_{it} to w_{i0}. Finally, we calculate $p(S_i) = \texttt{softmax}(W \cdot [\, \overrightarrow{h}_{t+1}; \overleftarrow{h}_0 \,] + b)$.

7. **S-LSTM**: a recent LSTM variant for encoding text with a parallel "sentence-level sub state g_t" for each time step (Zhang et al., 2018). Formally, for the S-LSTM model, $[\, \overrightarrow{h}_{t-1}, x_t \,]$ in Equation (1) is replaced by $[\, \xi_t, x_t, g_{t-1} \,]$ and c_t takes information flow from the left context cell c^l_{t-1}, the right context cell c^r_{t-1}, and the sentence context cell c^g_{t-1}, i.e.,

$$c_t = f_t \odot c_{t-1} + l_t \odot c^l_{t-1} + r_t \odot c^r_{t-1} + s_t \odot c^g_{t-1} + i_t \odot (W_c \cdot [\, \xi_t, x_t, g_{t-1} \,] + b_c), \tag{2}$$

where $\xi_t = [h^{i-k}_{t-1}, ..., h^{i-1}_{t-1}, h^i_{t-1}, h^{i+1}_{t-1}, ..., h^{i+k}_{t-1}]$, k is the window size for controlling information exchange between neighboring words. The value of g_t is computed from h^i_{t-1} for all word index i, i.e., g_t is the hidden state for an LSTM cell, where $\bar{h} = \texttt{avg}(h^i_{t-1})$ is the input. Finally, the classification layer takes global g as the sentence representation, $p(S_i) = \texttt{softmax}(W \cdot g + b)$.

8. **BERT**: a powerful representation learning model that uses a transformer network to pre-train a language model with a masked tokens prediction task and a next sentence prediction task (Devlin et al., 2019). We fine-tune the 340M parameters model released by Google with the vocabularies of the training sets. Next, we use the cross entropy loss to train a binary classifier that takes the BERT context embedding for sentence S_i.

3 Experiments

In this section, we provide further details on the datasets, experimental settings that produce the classification results, and evaluation metrics.

3.1 Datasets

We conduct sentiment analysis on two datasets to enable a comparison: (1) the Yelp dataset by Zhang et al. (2015) for the business review domain and, (2) the StockTwits Sentiment (StockSen) dataset[1] for the finance domain. Table 1 summarizes the statistics of the datasets.

Dataset	training pos.	training neg.	test pos.	test neg.	token size (vocab.)
Yelp	280,000	280,000	19,000	19,000	810,382
StockSen	10,558	3,899	4,542	1,676	40,069

Table 1: Statistical information for the two datasets.

[1] The StockSen dataset is available from the authors upon reasonable request and for research purpose.

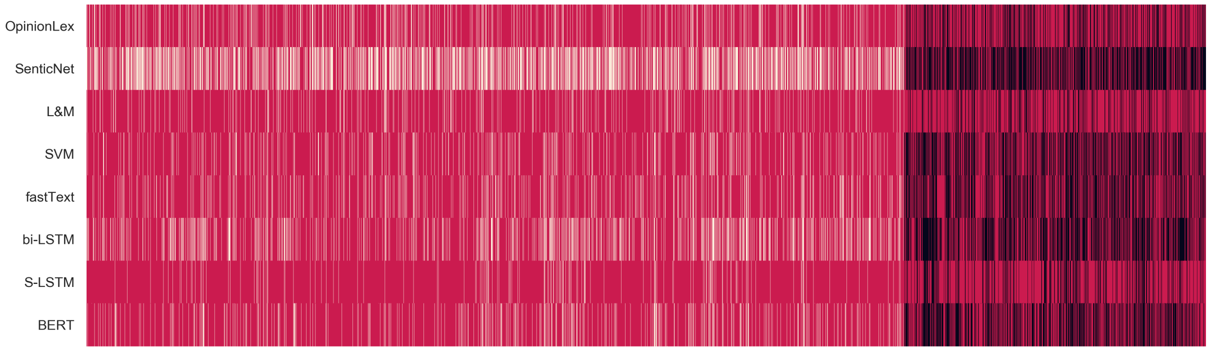

Figure 1: A visualization of the financial sentiment classification error distributions by different methods. The x-axis aligns texts in chronological order. White stripes denote false negative (type II), and black stripes denote false positive (type I) errors. We can observe that machine learning-based and deep models make denser false positive errors than false negatives, except S-LSTM.

The StockSen dataset comprises 55,171 texts accessed from the StockTwits platform dated between 2019-06-06 to 2019-08-26. After filtering out the entries without self-labeled sentiment, we obtained 20,675 financial tweets (labeled either positive or negative), where 6,218 tweets in total (4,542 positive and 1,676 negative tweets) are randomly selected and used for testing. We manually checked the self-labels to confirm that they are of high-quality. In fact, understanding the sentiment of some examples without the help of self-labels is barely possible. The quality ensures the high confidence level of metrics even though the size of the StockSen dataset is much smaller than the Yelp dataset. The StockSen dataset is imbalanced by nature: positive texts posted triples the number of negative ones. In contrast, the Yelp dataset is balanced. We keep these prior distributions to see whether it affects the evaluation.

3.2 Experimental Setup

The lexicon-based models (OpinionLex, SenticNet, and L&M), as this name manifests, do not make use of the sentiment labels, hence they are deterministic and training-free. For them, we follow (Taboada et al., 2011) for handling (double) negations. The SVM implementation uses the default regularization and kernel of Scikit-learn (Pedregosa et al., 2011). Our implementation of both bi-LSTM and S-LSTM use the 6B-tokens-uncased-300d GloVe embeddings (Pennington et al., 2014). All the three deep models are trained with an Adam optimizer (Kingma and Ba, 2015) with the initial learning rate equal to $2e^{-5}$. For BERT, the training and testing batch sizes are set to 24 and 8 for efficiency. Among the eight models, OpinionLex, L&M, SVM, and fastText are bag-of-words type models while the rest consider syntactic or sequential features of a text.

To facilitate fair comparisons between different methods in a real environment, we apply minimum pre-processing for the tweets. The texts are uncased and URLs are represented as single tokens. Stock tickers, emojis, microtext, and mis-spellings remain unchanged.

3.3 Result Evaluation

We are less interested in the performance metrics alone but also the error patterns. Therefore, instead of simply counting the type I error and type II error, we visualize the sample-wise sentiment classification errors in Figure 1. Furthermore, we illustrate and analyze the correlations between predictions of the eight experimented methods in Figure 2. That is, if two methods give opposite predictions (one positive one negative) on every sample, their correlation will be -1; if the predictions are completely the same, their correlation will be 1.

Considering the fact that the StockSen dataset is unbalanced by nature, i.e., positive examples triple the number of negative ones, we follow the idea of (Xing et al., 2018b) and include Matthews Correlation Coefficient (MCC), which is specifically designed for imbalanced data along with the more common F-score and accuracy measure to comprehensively evaluate the classification results. The number of true positive, false positive, true negative, and false negative samples are denoted by tp, fp, tn, fn,

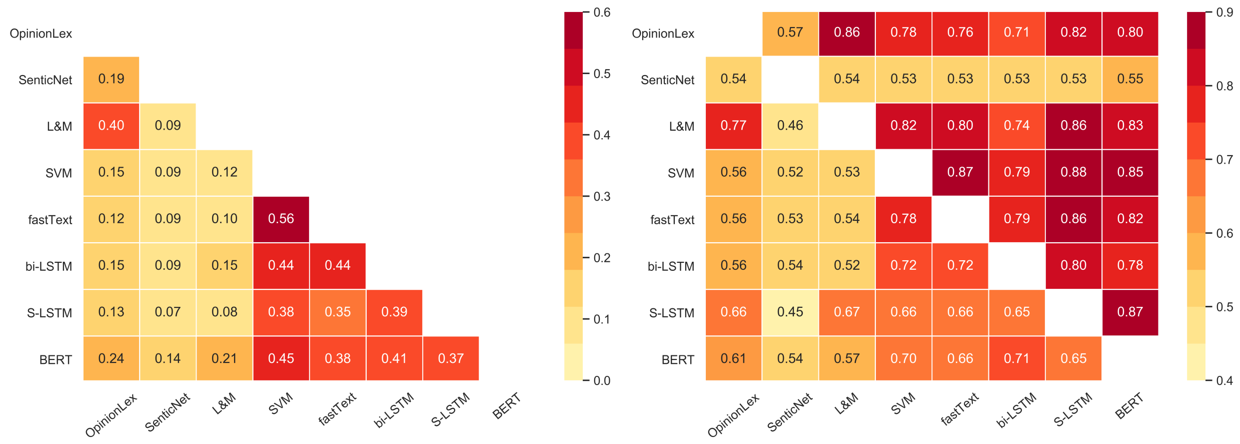

Figure 2: Pairwise correlation matrices of eight model predictions on the StockSen test data. The left matrix is symmetric, showing high correlations across learning-based models. The right matrix shows correlations for the bullish / positive (upper triangular) and bearish / negative samples (lower triangular).

respectively. Then we calculate the metrics as follows:

$$\text{MCC} = \frac{tp \times tn - fp \times fn}{\sqrt{(tp+fp)(tp+fn)(tn+fp)(tn+fn)}}, \tag{3}$$

$$\text{F-score} = \frac{2 \times tp}{2 \times tp + fn + fp}, \tag{4}$$

$$\text{Accuracy} = \frac{tp + tn}{tp + fp + tn + fn}. \tag{5}$$

The experimental results on both datasets are reported in Table 2.

4 Discoveries and the Answer to Research Questions

We observe from Table 2 that the sentiment analysis performance is significantly less accurate in the finance domain given the same models. On average, MCC drops from 74.4 to 42.6; F-score drops from 86.0 to 80.6; accuracy drops from 84.4 to 71.6. The visualized error patterns (Figure 1) show that lexicon-based models make uniform errors, e.g., SenticNet makes more errors for both positive and negative samples. Four out of five learning-based models (SVM, fastText, bi-LSTM, and BERT), in contrast, all make more false positive errors than false negatives. This observation may indicate that they fail to tackle the imbalanced data problem, however, S-LSTM somehow learned much-balanced errors. BERT has the highest scores on all metrics for the Yelp dataset thanks to its strong expressive power. However, on the StockSen dataset, Table 2 shows that a simple SVM classifier could also achieve the state-of-the-art performance of BERT.

Figure 2 shows the subtle phenomenon that the pairwise correlations are higher inside "model clusters". We observe the highest correlation from the left matrix between SVM and fastText, hence forms the machine learning-based cluster. Similarly, the deep learning model cluster (bi-LSTM, S-LSTM, BERT) has a darker color. OpinionLex and L&M ($corr = 0.40$) are also highly correlated, though SenticNet makes the lexicon-based cluster loose. This may attribute to the fact that SenticNet leverages syntactic rules for sentiment classification.

Considering that all the models are trained on the same StockSen training data, the highest correlation (between SVM and fastText, $corr = 0.56$) is surprisingly low. Figure 1 also confirms no clear pattern. For **RQ1**, it seems that the errors are model-specific and even less consensus are on the bearish samples (the upper part is darker in the right matrix of Figure 2). For **RQ2**, though each metric roughly tell the model performance, inside clusters they may not rank models in the same way. For example, MCC and accuracy rank SenticNet as the best lexicon-based model on the Yelp dataset, while F-score ranks

Method	MCC(Yelp)	MCC(StockSen)	F-1(Yelp)	F-1(StockSen)	Acc.(Yelp)	Acc.(StockSen)
OpinionLex	47.6	39.3	74.6	80.1	67.1	69.3
SenticNet	60.6	44.8	73.8	62.4	73.0	54.5
L&M	51.3	34.5	74.3	81.7	68.1	70.7
SVM	88.3	**52.0**	93.8	**84.7**	93.8	**76.9**
fastText	87.2	38.4	93.6	83.7	93.6	76.0
bi-LSTM	85.3	42.1	91.6	83.0	92.0	73.7
S-LSTM	84.2	37.8	90.8	**84.7**	91.5	75.1
BERT	**90.8**	51.9	**95.6**	84.5	**95.7**	**76.9**
Average	74.4	42.6	86.0	80.6	84.4	71.6

Table 2: Sentiment classification results on the StockSen dataset (6,218 test samples in sum) and on the Yelp-2015 Dataset (38,000 test samples in sum), measured by three metrics. The best performance for each column are in bold. Numbers are in percentage (%).

OpinionLex the best; MCC and accuracy rank BERT as the best deep model on the StockSen dataset, while F-score, again, chooses S-LSTM. Therefore, it is dangerous to compare models with a single metric. Although L&M is specifically cast for the finance domain, the three lexicon-based models all perform better on the Yelp dataset than on the (very noisy) StockSen dataset. In terms of MCC, L&M is even the worst among lexicon-based methods on the StockSen dataset (see Table 2). As an answer to **RQ3**, simply using a domain adapted lexicon does not necessarily solve the FSA problem.

5 Reasons behind Classification Errors

Since this paper studies the reasons for classification errors, especially for FSA, we take a deeper look into 384 "overlapping errors" in sum, whereby 237 positive examples are correctly predicted by less than four models, and 147 negative examples are wrongly predicted by all eight models experimented. This error set contains some extremely difficult cases, such as "*what goes up goes down and vice versa*", which flips its polarity for many times; and "*the downtrend is not because [...] is a bad investment*", which express a positive sentiment with all negative words. Although not all the errors from the StockSen dataset are explainable, the majority are explainable and we found the six interesting error types as follows.

Unlike the error analysis by Abbasi et al. (2014), which generally focused on machine learning features and is specific to Twitter data, our analysis emphasizes more on linguistic phenomena. One may ask whether these error types really afflict the sentiment analysis performance and are specific only to the finance domain. Therefore, we also look into 237 positive and 147 negative examples randomly sampled from the "overlapping errors", and 384 random texts from the test population on the Yelp dataset. Indeed, we find more significant concentrations of these linguistic features in the errors from financial domain. Table 3 shows the estimated concentrations of each type in the datasets.

5.1 Irrealis Moods

We found a rich spectrum of irrealis, and usually counterfactual moods (McShane et al., 2004; Negi and Buitelaar, 2015) in the finance domain errors (22 out of 384 test cases, 5.73% and 3.3× more in concentration). In the business review domain, the concentration of irrealis moods is even higher (6.7×).

Conditional mood: "*if there was any better opportunity to exit long term holdings [...] It would be this month*". In this example, the speaker is trying to say that the best exit opportunity is this month, hence being pessimistic for the future. If the model fails to detect the conditional mood, spotting phrase "better opportunity" would lead to a wrong polarity as positive. Counterfactual conditional sentences, or second and third conditionals, usually have a connective "*if*" and a past/pluperfect tense in the hypothetical clause, i.e., "if+ VBD/VBN, would/should/might+VB/VBN" (Narayanan et al., 2009).

Dataset	Irrealis	Rhetoric	Depend. Opinion	Unspec. Aspects	Unrecog. Words	External Ref.
Yelp test	1.74	1.50	0.52	0.50	1.95	0.50
Yelp error	**11.68**	1.82	0.52	0.78	2.60	0.52
StockSen test	1.75	1.95	2.00	2.08	9.50	2.01
StockSen error	**5.73**	**6.51**	**10.16**	**14.58**	10.94	5.21

Table 3: Estimated concentrations of the linguistic features of errors in both datasets, with concentrations at least three times higher (3×) in error sets in bold font. Numbers are in percentage (%).

Subjunctive mood: "*Would be shocked if this closes under 1900 tmrw*"; "*would be amazing if it touch 210 tomorrow*". The two sentences are more than conditional because the first person actions imply judgment for a future time expression "*tomorrow*". The subjunctive mood is important because the action words usually carry sentiment, and the sentiment is later reversed. In the first sentence, the speaker actually means that he is confident that the close price tomorrow will be higher than 1900 and in the second, he expresses that he would not believe it can touch 210. Failure to detect such moods only passes "amazing" to the bag-of-words like models and produces a false positive sentiment.

Imperative mood: "*2 negative articles paid for by short sellers. Dont believe them!*". This example has the typical imperative syntax with the bare infinitive form of the verb, an omitted second person pronoun subject, and an exclamation at the end. The speaker requests not to believe the negative articles written by short sellers. So, he is positive toward it.

5.2 Rhetoric

Rhetoric, in contrast to irrealis moods, is more difficult to detect because it is based not only on syntactic features but also on semantic meanings. It includes but is not limited to the sub-types below.

Negative assertion: "*In the market for a iwatch and airpods too. What recession?!*"; "*Who buys at 205? Not me.*". These two are typical rhetoric questions that are intended to propose a challenge, not to receive an answer. By asking "*what recession*?!", the speaker denies there is any economic recession and by asking "*who buys*", he means no one would buy. A negative assertion is signaled when the question is followed by an answer or conveys opposite sentiment to its indicative context.

Personification: "*$TSLA fighting for its life here*". This example refers to the price movement of a company ($TSLA) as a human "fighting for his life" to describe its toughness. Personification can be detected with part-of-speech (POS) and named entity recognition (NER) tags.

Sarcasm: "*$AMZN the Amazon board is hilarious*". In this example, "*hilarious*" is a positive word. However, when "*beautiful*" or "*hilarious*" are used to describe a board, these uncommon collocations express a sarcastic meaning, which resonates with the theoretical study that "inappropriateness" is a key feature for sarcasm (Attardo, 2000). In StockSen and Yelp datasets, sarcasm is almost always found in negative test cases. Sarcasm is closely connected to affective information and joint modeling can improve detection rates of both linguistic devices (Hernańdez Farías et al., 2016).

5.3 Dependent Opinion

The canonical form of sentiment expression addresses the object itself. However, we found that addressing the third person is pervasive in financial tweets, though they express their sentiment toward the stocks. Therefore, their opinion depends on "others' opinion", for example, in "*[...] I think many underestimate it. spring how*" and in "*bulls need 236.. not far away..*".

In the first example, "*underestimate*" is the action of the third person. The speaker holds a different opinion. In the second example, "*bulls need 236*" is just a statement-of-fact. The positivity comes from the latter part "*not far away*": an agreement to the bulls. In another example "*any bear who says the company will fail is ignorant, [...]*", though it is filled with negative phrases, such as "*company fail*" and "*ignorant bear*", he actually disagrees with the bear-opinionated people and advocates for the company.

5.4 Unspecified Aspects

In addition to the complexity of dependent opinion, there can be more than one possible sentiment aspect in a sentence. For example, in "*$SOLY Allergan should be concerned that cool sculpting will be rendered obsolete*", the speaker thinks Allergan (a pharmaceutical company)'s technique is obsolete. However, he is commenting on its competitor Soliton ($SOLY). Therefore, the sentence is labeled as positive by the speaker. Another example is "*$AAPL most recession fears are media infused. I hate to say it but fake news is becoming a daily reality*". In this example, though "*fake news*" and hence "*recession fears*" are negative, he is blaming this negative sentiment to "media". The commented aspect "$AAPL", as a real company, is regarded as positive. It is relatively easy to identify the target aspect as it often appears as a "cashtag". While sentiment analysis models have to decide whether the sentiment is toward other aspects.

5.5 Unrecognized Words

Because the financial tweets are from a professional microblogging platform, an informal interactive cyberspace that has character limit, investors tend to use words that are difficult to link to their semantic meanings. These words at least include unrecognized entities, microtext, and jargons.

Entity: "*$AAPL time to upgrade my 6s. [...] Otherwise flawless.*". Here "*6s*" refers to the "*iPhone 6s*" model, though in another context it may be resolved to "*6 seconds*". Detection and completion of such entities sometimes require financial facts and commonsense knowledge.

Microtext: These are non-standard spellings such as phonetic substitutions and acronyms. Like in example, "*it will break 1800 EOW.*", "*EOW*" often appears for "*end of week*". Similarly, we found *EOD* (end of day), *tmrw* (tomorrow), *mkt* (market), *n* (and), *maga* (make america great again), *tezzzzla* (Tesla), *imo* (in my opinion), etc.

Jargons: Unlike microtext, jargons are not out-of-vocabulary (OOV) words. However, they bear domain-specific meanings that may convey different sentiment. In "*the stock formed a head and shoulders on a 5 day*", "*head*" and "*shoulder*" are terms from technical analysis chart, which indicate a recent downward trend. Therefore, instead of neutral, the sentence is negative. In "*$AAPL Head and shoulders bottom is forming on AAPL*", "*head and shoulders bottom*" (the inverse shape of a head and shoulders) signals a positive sentiment.

5.6 External Reference

Reference to external facts and knowledge that are absent from the text is common. Usually, the facts are considered accessible to the whole community. Different from entity recognition, reference requires logical reasoning and detection of temporal orientation (Hasanuzzaman et al., 2015) to understand the sentiment. For instance, "*$TSLA 200 tomorrow XD*" refers to the current price (today), and "*Lets get down to price levels beginning of the year!!!*" refers to a historical price (beginning of the year). For the first case, we have to access the timestamp (2019-06-20) and the current price (219.62) to know that "200 tomorrow" is a downside, negative prediction (200<219.62).

6 Conclusion

In this paper, we evaluated the performance of some of the widely acknowledged lexicon-based, machine learning-based, and deep models for financial sentiment analysis. We went beyond simply comparing model metrics by visualizing the error patterns and conducting linguistic analysis. Results confirm that models within the same cluster are prone to similar error patterns. We also conclude six reasons that cause financial sentiment analysis to fail, i.e., *irrealis mood*, *rhetoric*, *dependent opinion*, *unspecified aspects*, *unrecognized words*, and *external reference*. Irrealis mood, in particular, also accumulates in the failed cases of sentiment analysis in the business review domain. The error types, of course not exclusive for financial sentiment analysis and may appear elsewhere, are possible avenues for improving model performance. These findings also contribute to the understanding of semantics and pragmatics of financial tweets.

It is frustrating to see a number of errors, which are not difficult for human, whereas all the experimented methods failed on them. Therefore, we argue that having 'silver bullets' for financial sentiment analysis at the current stage is an illusion. More broadly speaking, sentiment analysis is a suitcase problem, which cannot be perfectly solved by an end-to-end model without handling the many NLP sub-problems. A divide-and-conquer approach is particularly needed from financial sentiment analysis, which has limited training data and unique language styles. In future work, we plan to extend our experiments to more datasets, and to incorporate separate NLP modules that deal with the discovered error types to improve existing sentiment analysis models.

Acknowledgments

This work was partially supported by the PPF grant: innovative approaches to portfolio optimization. The authors would like to thank anonymous reviewers for their suggestions and comments.

References

Ahmed Abbasi, Ammar Hassan, and Milan Dhar. 2014. Benchmarking Twitter sentiment analysis tools. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 823–829.

Salvatore Attardo. 2000. Irony as relevant inappropriateness. *Journal of Pragmatics*, 32(6):793–826.

Haodong Bai, Frank Z. Xing, Erik Cambria, and Win-Bin Huang. 2019. Business taxonomy construction using concept-level hierarchical clustering. In *Proceedings of the First Workshop on Financial Technology and Natural Language Processing (FinNLP)*, pages 1–7.

Erik Cambria, Dheeraj Rajagopal, Daniel Olsher, and Dipankar Das. 2013. Big social data analysis. In R Akerkar, editor, *Big Data Computing*, chapter 13, pages 401–414. Chapman and Hall/CRC.

Erik Cambria, Yang Li, Frank Z. Xing, Soujanya Poria, and Kenneth Kwok. 2020. SenticNet 6: Ensemble application of symbolic and subsymbolic AI for sentiment analysis. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, pages 105–114.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2015. Deep learning for event-driven stock prediction. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2327–2333.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1615–1625.

Mohammed Hasanuzzaman, Sriparna Saha, Gaël Dias, and Stephane Ferrari. 2015. Understanding temporal query intent. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 823–826.

Delia Irazú Hernańdez Farías, Viviana Patti, and Paolo Rosso. 2016. Irony detection in twitter: The role of affective content. *ACM Transactions on Internet Technology*, 16(3):19:1–24.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth international conference on Knowledge discovery and data mining (KDD)*, pages 168–177.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of EACL*, pages 427–431.

Katherine A. Keith and Amanda Stent. 2019. Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 493–503.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, pages arxiv[v8]:1–15.

Tim Loughran and Bill McDonald. 2011. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *Journal of Finance*, 66(1):35–65.

Ling Luo, Xiang Ao, Feiyang Pan, Jin Wang, Tong Zhao, Ningzi Yu, and Qing He. 2018. Beyond polarity: Interpretable financial sentiment analysis with hierarchical query-driven attention. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4244–4250.

Lorenzo Malandri, Frank Z. Xing, Carlotta Orsenigo, Carlo Vercellis, and Erik Cambria. 2018. Public mood-driven asset allocation: the importance of financial sentiment in portfolio management. *Cognitive Computation*, 10(6):1167–1176.

Marjorie McShane, Sergei Nirenburg, and Ron Zacharski. 2004. Mood and modality: out of theory and into the fray. *Natural Language Engineering*, 10(1):57–89.

Ramanathan Narayanan, Bing Liu, and Alok Choudhary. 2009. Sentiment analysis of conditional sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 180–189.

Sapna Negi and Paul Buitelaar. 2015. Curse or boon? presence of subjunctive mood in opinionated text. In *Proceedings of the 11th International Conference on Computational Semantics (IWCS)*, pages 101–106.

Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Marina Sedinkina, Nikolas Breitkopf, and Hinrich Schutze. 2019. Automatic domain adaptation outperforms manual domain adaptation for predicting financial outcomes. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 346–359.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly D. Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

Frank Z. Xing, Erik Cambria, Lorenzo Malandri, and Carlo Vercellis. 2018a. Discovering bayesian market views for intelligent asset allocation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, pages 120–135.

Frank Z. Xing, Erik Cambria, and Roy E. Welsch. 2018b. Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1):49–73.

Frank Z. Xing, Erik Cambria, and Yue Zhang. 2019. Sentiment-aware volatility forecasting. *Knowledge-Based Systems*, 176:68 – 76.

Frank Z. Xing, Duc-Hong Hoang, and Dinh-Vinh Vo. 2021. High-frequency news sentiment and its application to forex market prediction. In *Proceedings of the Hawaii International Conference on System Sciences (HICSS)*.

Yumo Xu and Shay B. Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1970–1979.

Linyi Yang, Tin Lok James Ng, Barry Smyth, and Ruihai Dong. 2020. HTML: Hierarchical transformer-based multi-task learning for volatility prediction. In *Proceedings of The Web Conference*, pages 441–451.

Zhe Ye, Fang Li, and Timothy Baldwin. 2018. Encoding sentiment information into word vectors for sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 997–1007.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*, pages 649–657.

Yue Zhang, Qi Liu, and Linfeng Song. 2018. Sentence-state lstm for text representation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 317–327.