ESG UQÀM

Centre d'intelligence en surveillance des marchés financiers

Les cahiers de recherche du CISMF
CISMF Research Paper Series

Targeted Financial-Oriented Social Media Sentiment Measurement: Natural Language Processing Approach

Gilles Caporossi [a], Pan Liu[a], Feng Zhan[b] & Xiaozhou Zhou[c]

[a] Department of Decision Sciences, HEC Montréal, Canada
[b] DAN Management and Organizational Studies, University of Western Ontario, Canada
[c] School of Management, University of Quebec at Montreal, Canada

Février / February 2024

Targeted Financial-Oriented Social Media Sentiment Measurement: Natural Language Processing Approach*

Gilles Caporossi[1], Pan Liu[1], Feng Zhan[2], and Xiaozhou Zhou[3]

[1] HEC Montréal
[2] DAN Management and Organizational Studies, University of Western Ontario
[3] School of Management, University of Quebec at Montreal

February 25, 2024

Abstract

This study develops a natural language processing model that measures financial-oriented sentiment targeted toward specific firms in social media texts. First, we create a human-annotated social media targeted financial sentiment dataset. Then, we propose a prompt-based model architecture that achieves state-of-the-art performance on multiple benchmark datasets for general targeted sentiment analysis. Subsequently, we finetune this model using our annotated dataset, which allows it to measure targeted financial sentiment with high accuracy. We apply it to 23 million financial-oriented social media posts from different platforms to measure financial sentiment toward 24 meme stocks (stocks that gain frenetic attention from retail investors on social media which is often accompanied by dramatic price movement) and 30 Dow Jones constituent stocks. Our results show that the sentiment measured by our model is positively correlated with price return and negatively correlated with price volatility, and that this correlation is stronger for meme stocks than for Dow Jones stocks. We further demonstrate that our model's sentiment measurement economically outperforms other representative financial sentiment measurements by comparing the returns of the same trading strategy built upon them separately.

* We acknowledge the *Financial Market Surveillance Intelligence Centre* of ESG UQAM for its financial support.

1 Introduction

A growing number of investors are turning to social media as a source of information with the advent of mobile technology and online communities. Furthermore, a significant number of business and political influencers, as well as traditional media agencies, have been utilizing social media such as Twitter, as a primary means of communicating with their audiences. As such, their posts contain specific information and could cause a significant reaction in the market. In addition, social media popularity has combined with the increased activity of retail investors in recent years, contributing to their growing influence in the marketplace, as exemplified by the "meme stocks" frenzy. The term 'meme' refers to a stock that is receiving intensive attention on social media such as Twitter, Reddit and StockTwits forums, and concurrently experiencing dramatic price movement. The growing influence of social media leads to increased research interests in its role in the financial market. Pedersen (2022) introduces a model to explain the mechanism of information propagation through social network and the consequent affects to the market. Hirshleifer (2020) highlights the transmission bias of economic and financial signals induced by social media. Financial sentiment, long recognized as an important market impacting factor, has been one of the focuses in studying social media. Sentiment from StockTwits is shown to forecast short-term stock index return (Renault, 2017), and is used to study the source of disagreement among investors which is the foundation of trading (Cookson & Niessner, 2019), and echo chamber effect which leads to investors' confirmation bias (Cookson, Engelberg, et al., 2022). More broadly, Cookson, Lu, et al. (2022) studies the social media sentiment and attention from Twitter, StockTwits, and Seeking Alpha, finding that sentiment-induced retail imbalances predict positive returns while attention-induced ones have the opposite market outcomes. Although the importance of financial sentiment in social media is widely recognized, its accurate measurement, however, is very challenging and less studied. In this paper, we propose targeted sentiment analysis (TSA) model using advanced natural language processing (NLP) methods, to address this problem.

Targeted sentiment, defined as the sentiment targeted towards a specific entity or aspect within a text, is a more refined measurement compared to the common overall sentiment

of a text. Often, a piece of text can contain multiple entities or aspects, which may bear completely different sentiments. For example, considering the sentence:

"Morgan Stanley says Disney could surpass Netflix in the streaming market."

The sentiment expressed would be totally different depending on which firm is the target of interest: neutral for *Morgan Stanley*, positive for *Disney*, and negative for *Netflix*. A common sentiment model can only predict one sentiment for the sentence as a whole regardless of which firm is being focused on, hence has no means to distinguish the individual sentiments towards the different firms separately. By contrast, only a TSA model designed to measure the fine-grained sentiment specific to a given target, could correctly predict the different sentiments given the different firms.

Targeted sentiment is especially important when studying financial oriented social media. For many traditional financial texts such as the earnings report or the financial analyst report, usually each text explicitly concerns only one particular firm, and this targeted firm is clearly documented in the database. Moreover, a such text is often in the form of a long document, and we know for sure the majority of its content is about the known target. Thus, for those type of texts, there's little problem attributing the overall sentiment obtained by pooling the measurements of all the parts of the document to the target firm. However, for the social media texts which are usually short posts by individual users, we don't have prior knowledge of which post is about what. In practice, to study a certain firm, we have to collect all financial oriented posts mentioning this firm. But unlike a long document with clear target, we have no guarantee weather the overall sentiment of a such post is really about this firm. That's why a TSA model is crucial for accurately measuring the sentiment conveyed by social media posts.

Measuring financial-oriented sentiment towards specific firms within social media texts is a challenging problem due to several reasons.

First, there's a scarcity of labeled data and advanced models for **financial-oriented sentiment** analysis. Although the measuring of textual sentiment has been extensively studied in the field of NLP, most of the data and research focus on general sentiment,

typically customers' likes or dislikes expressed in online reviews. However, the financial sentiment, defined as view of a favorable or unfavorable prospect from an investor's perspective, is very different and is much less investigated. There's a lack of large-scale labeled financial sentiment dataset to train and evaluate models, because of the difficulty of the labeling which requires expert knowledge. Unlike the general sentiment for which there are numerous public datasets of online reviews of size up to hundreds of millions, such as the Stanford Sentiment Treebank of movie reviews by Socher et al. (2013) and the Amazon Review Data by Ni et al. (2019), to our knowledge, only two datasets are publicly available for financial sentiment. One is the Financial Phrase Bank (FPB) dataset (Malo et al., 2014) that provides ~5000 financial news sentences each labeled with financial sentiment by multiple annotators. The other is the Financial Opinion Mining and Question Answering (FiQA) dataset (Maia et al., 2018) that contains 436 news and 675 social media posts from financial web pages, each labeled with fine-grained aspect-targeted financial sentiment. In terms of modeling, researchers in finance used to rely on word-counting based on tailored dictionaries of financial sentiment keywords, the most popular of which is proposed by Loughran & McDonald (2011). Besides dictionaries, classical statistical models such as naïve Bayes were also widely adopted (Antweiler & Frank, 2004; Das & Chen, 2007; Huang et al., 2014). Those models, treating a text as a simple bag of words with their order and context disregarded, are incapable of capturing complex semantics, thus leading to inaccurate measurements of the true sentiment in many cases. More recently, large pretrained language model (PLM) based on transformers, a novel NLP deep-learning architecture, has become dominant in the NLP field and achieved significant performance breakthroughs across various NLP tasks. Riding on this trend, some researchers in the domain of finance also begin to explore adapting the transformers models to financial sentiment analysis, and reported improved performance compared to traditional models (Araci, 2019; Jiang et al., 2022; A. H. Huang et al., 2022).

Second, measuring **targeted sentiment** requires more advanced models and is less studied compared to commonly measuring sentiment at the whole sentence or document level. Classical lexicon and machine learning based models, relying heavily on hand crafted rules and feature engineering, can perform well on sentence or document level

sentiment tasks, but have difficulty measuring the fine-grained targeted sentiment. Only with the recent development of deep neural network, can the complex dependency and interaction between the target and its context be effectively modeled. Apart from more complex model, labeled targeted sentiment data is also more burdensome to get. As we mentioned before, sentence or document level sentiment data can be easily obtained by mass amount from the online reviews. But fine-grained targeted sentiment datasets are much scarcer, only few public English datasets are available. The most used are the SemEval2014 laptops and restaurants review datasets (Pontiki et al., 2014) which contain online reviews on laptops and restaurants with each review manually labeled with sentiment towards different aspects, such as "service", "staff", "food", etc. for restaurants. Another one is the Twitter dataset (Dong et al., 2014) containing tweets manually labeled with sentiment towards different targets including celebrities, products, and companies. In the domain of finance, the only public targeted financial sentiment dataset available is the aforementioned FiQA dataset.

Third, the informal nature of social media texts makes them more challenging for NLP models (Farzindar & Inkpen, 2020). Non-standard or even incorrect grammatical structure and word spelling are very common in social media posts. Also, like many web content, social media are plagued with much more noise in terms of irrelevant content compared to formal media. Moreover, being short in length and often in a conversational nature, a social media post often provides very limited contextual information that is essential for language understanding. Considering all those issues, social media requires more advanced models that are more tolerant of informal language and noise, and have better ability comprehending texts in relation to their contexts.

In this paper, we address the above challenges by developing an NLP model that measures the financial sentiment targeted towards specific firms in social media texts. First, we create a targeted financial sentiment dataset of ~3000 social media posts, each annotated by multiple people with academic background in business to ensure the quality. This dataset adds to the rare public data resources regarding both financial sentiment and targeted sentiment. Then, we propose a novel NLP model architecture based on the prompt paradigm, which functions as reformulating the TSA task to imitate the natural

language inference (NLI) task on which the backbone transformers model was well pretrained with massive data. Our model proves to achieve state-of-the-art (SOTA) performance on multiple benchmark datasets for general TSA task. Subsequently, we finetune the model based on our targeted financial sentiment dataset, which enables it to measure targeted financial sentiment with high accuracy. Finally, we apply our finetuned model to over 23 million financial-oriented social media posts between 2020 and 2022 to measure financial sentiment towards 24 meme stocks and 30 Dow Jones constituent (DJ30) stocks. We show that the sentiment measured is positively correlated with price return and negatively correlated with price volatility. Moreover, we demonstrate that this correlation between social media sentiment and price is significantly stronger for meme stocks than for DJ30 stocks. We further construct a sentiment-based trading strategy using different financial sentiment measures. The return differences demonstrate that our model's sentiment measurement economically outperforms the other two representative financial sentiment measurements.

2 Background and Literature

2.1 Advanced Natural Language Processing

NLP is the subfield of artificial intelligence that aims at enabling computers to process and analyze human language, in order to perform relevant tasks such as machine translation, sentiment analysis, document summarization, question and answering, etc. Recent years have seen huge advancement of NLP due to several key factors including vast growth in computing power, increased availability of a large linguistics data, development of highly successful machine learning algorithms, and richer understanding of the language structure and its deployment in social contexts (Hirschberg & Manning, 2015). The rest of the chapter will cover some basic concepts and the development of modern NLP.

Representation Learning

For textual data to be processed by algorithms, first they need to be represented as numeric vectors. Classical NLP methods often treat a document as a **bag of words (BOW)**, i.e., a collection of independent words (or n-grams) that can be simply represented by a vector

of the counts of each word in the vocabulary. However, words represented in this way lose their semantic meanings, they become atomic units that have no inherent relationship to one another. For instance, the concept of synonym or antonym is completely absent. Furthermore, a BOW representation has the dimensionality equivalent to the size of the entire vocabulary, resulting in a large and usually sparse vector. This can lead to significant computational inefficiency.

A milestone for representation learning is achieved with the novel **word embeddings** methods: featurized word representations that preserve semantic information of words. Instead of merely being a numeric encoding without meaning, the new word representations can capture syntactic and semantic regularities that enable analogy reasoning. Mikolov, Yih, et al. (2013) found that using the word embedding vectors they generated, semantic relationships can be represented using simple arithmetic, e.g., "$King - Man + Woman = Queen$". The word embeddings are learnt from large unlabeled text corpuses, which are easy to obtain. The generated word embeddings can then be applied for downstream tasks, which can greatly boost their performances since more semantic information of words can be of great value to those tasks. The most influential word embedding algorithms include *word2vec* (Mikolov, Chen, et al., 2013) and *GloVe* (Pennington et al., 2014). Those advanced word embedding models have greatly boosted the performance of many NLP tasks, because of their ability to extract and preserve words' meaning by simply being pretrained on a large unlabeled corpus. This concept of gaining general knowledge from training on large unlabeled data, in order to later apply the knowledge learned to other downstream tasks, is the core idea of transfer learning, which we will introduce in the following.

Transfer Learning

A major assumption for statistical learning algorithms is that the training data and the future data to be applied on must be in the same feature space and of the same distribution. However, this can't always be satisfied for real-world applications: often we only have a small amount of labeled data for our task of interest (target task), but we may have enough data from another related task (source task) where the feature space or distribution is

different. In this case, if we can let the model "pre-train" on the source task data and then transfer the knowledge it leans to apply on the target task, it would improve the performance without expensive additional data labeling effort. This reasoning leads to the development of transfer learning.

Transfer learning was earlier popularized in the field of computer vision (CV), because during pretraining CV models can effectively gain automatic feature extraction capabilities such as detecting the edge of objects or identifying shapes. Those capabilities will benefit all sorts of downstream CV tasks. Similarly, since NLP tasks also share common knowledge about the language, transfer learning was naturally introduced into the NLP field, and has become a fundamental methodology today. According to a summary by Ruder et al. (2019), the most common process of transfer learning in NLP today consists of two phases: 1. a pretraining phase in which general representations are learned on a source task or domain; 2. an adaptation phases in which the learned knowledge is applied to a target task or domain.

As we mentioned, word embedding is a typical case of transfer learning. Word embedding algorithms can extract word vectors that preserve the semantic meanings of words from merely unlabeled corpus, which corresponds to the phase of pretraining. Then in the adaptation phase, those word vectors are applied to represent the input texts of target tasks, which greatly improves the performance compared to using representation that contain no prior semantic knowledge of the word such as BOW. Despite its huge success, word embedding is essentially a "shallow" form of transfer learning, since it can only learn and represent individual word-level knowledge. This leads to one critical flaw: the representation of a word is not context-specific, whereas a word can have very different meanings in different contexts. In order to capture and transfer deeper-level language knowledge such as contexts and interactions of words within an entire text, researchers developed the method of language model pretraining.

Pretrained Language Model (PLM)

Language model is a type of NLP model that learns to probabilistically predict the next word given any previous sequence of words. The goal is to be able to assign a probability

for any given sentence or paragraph, based on the probability distribution it learned from the training language corpus (Goldberg, 2017). The pretraining of a language model usually means training the model on large-scale unlabeled corpus, with the task of reconstructing the original text given a text that has been artificially corrupted with certain noise functions. Pretraining enables the model to gain useful insights on the language, thus learning the representation of texts that captures deeper-level language knowledge including contextualized semantics.

Given a capable PLM architecture and proper pretraining tasks design, the more diverse data the model is pretrained on, the more comprehensive language knowledge it can encode (Pérez-Mayos et al., 2021). Although labeled text data are rare, fortunately, unlabeled text data is abundant and cheap to obtain, thanks to rapid digitalization and information boom. Many public large-scale corpuses are collected from sources like Wikipedia, news, books, web crawl, etc. Nowadays, it is common for large PLMs to be pretrained on more than a hundred GB of raw text data, such as the RoBERTa model (Liu et al., 2019). PLMs empowered by deep neural network, especially large PLMs developed in recent years, have become the fundamental technologies of NLP (Li, 2022).

As a simple usage example, the PLM can generate new texts that mimic the style of those in the training corpus, e.g., a PLM trained on a corpus of Shakespeare's poems and plays can "write" Shakespeare style texts. Beyond this little funny application, language modeling is actually a critical NLP task that lays the foundation for many other higher-level tasks. As an example, when performing French-English translation, given the input sentence *"Tom est gros"*, the model may have to weigh between *"Tom is large"* and *"Tom is fat"* as for output. If the model has good language knowledge, then it should recognize "Tom" as most likely a person's name, thus the more plausible adjective that follows should be "fat" instead of "large". So, the model should evaluate that $Probability("Tom\ is\ fat") > Probability("Tom\ is\ large")$, which leads to the correct translation. From this example, we may have a glimpse of insight that evaluating the probability of a sentence implies the judgement on the semantic and contextual information, sometimes even world knowledge in that sentence. Recent large PLMs pretrained on immerse amount of textual data have demonstrated the ability to incorporate

such complex context semantics and knowledge, leading to a revolutionary development of modern NLP.

2.2 NLP Development

In its early days, NLP relied heavily on linguistics study, researchers often attempted to predefine dictionaries and rules to decipher human language for the computer. However, language has proved to be too complex for this approach: language can be ambiguous, fuzzy, context-dependent, and often requires reasoning based on common sense. Gradually, researchers turned to the machine learning approach of applying statistical models over a large amount of data so that algorithms can learn empirical language patterns and knowledge by themselves. Classical machine learning models such as naïve Bayes and support vector machines achieved notable successes in NLP. However, those models usually treat texts with BOW method, which limits the models' ability to capture the order and dependency of words which are crucial for language understanding.

Later, **recurrent neural network (RNN)** based models such as long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent unit (GRU) (Cho et al., 2014), capable of handling texts as ordered sequences of words, dominated the NLP domain by greatly pushing the performance on many tasks. RNN is a special type of neural network whose units are connected recurrently along a sequence, allowing prior output based on previous values to affect subsequent input concerning the current value. This way RNN model can relate different parts of a sentence to understand the dependencies and contexts. For certain sequence to sequence tasks like machine translation, the input sequence and output sequence may have different lengths and the relationship between the two lengths is non-monotonic. This brings a problem for RNN which is good at mapping the input sequence to the output sequence only when the alignment between them is known a priori. To address this problem, the **encoder-decoder** architecture (Fig. 6) was proposed (Cho et al., 2014; Sutskever et al., 2014). The encoder is an RNN module that takes a variable-length sequence as input and output its encoding, i.e., a fixed-size vector that encapsulates all the information of input sequence. Then this encoding is passed to the decoder, another RNN module which is essentially a PLM, that predicts the output sequence with the highest possibility conditioning on the input.

However, there are bottlenecks with RNN encoder-decoder architecture. First, it needs to compress the information of a whole sentence into a fixed length vector, which leads to bad performance for long sentences due to loss of information. Second, the sequential computation of RNN precludes computing parallelization, which greatly limits the speed and scale of model training.

To address those issues, Bahdanau et al. (2014) first introduced a novel encoder-decoder model with the **attention mechanism**: when the model predicts a word at each time step, it searches the source sentence for the most relevant words, and use information of those words in addition to the previously predicted words to generate the current word prediction. In another word, like human reading, the model leans to pay attention only to those words in the source sentence that are relevant to the target word, instead of relying on encoding the whole source sentence. Also, without the recurrent modules, attention models are significantly more efficient with parallelization, which makes them way faster to train. The novel attention-based language models are called **transformers** model. Large pretrained transformers models have brought revolutionary improvements on almost all NLP tasks, greatly pushing the boundaries of NLP.

Two representatives and pioneers of transformers are the **GPT** (Generative Pretrained Transformer) (Radford et al. 2018) from OpenAI, and **BERT** (Bidirectional Encoder Representations from Transformers) (Devlin et al. 2019) from Google. The both adopts the pretrain-finetune paradigm to perform transfer learning. The initial version of GPT (GPT-1) has 117 million parameters, consisting of a 12-layer left-to-right transformer pretrained on a diverse corpus of unlabeled text, with a standard autoregressive PLM task, i.e., to predict the next word given the previous sequence of words. After the pretraining, the model is then fine-tuned on each specific down-stream task. For classification tasks, the finetuning involves simply concatenating a classification head, usually a shallow neural network, that takes the text embeddings generated by the PLM and make classification based on it. GPT with finetuning achieves SOTA performance on a wide range of NLP tasks. Its success demonstrates that large PLMs pretrained on massive raw text data are effective transfer learner when coupled with the fine-tuning approach. Due to the auto-regressive nature of GPT, it is especially suitable text generation related tasks.

The later BERT model pushed the SOTA even further. To overcome the unidirectional constraint of classical PLM, BERT invented a novel bidirectional pretraining objective called *"masked language model (MLM)"*: randomly mask some words within a text, and let the model predict those words based on the contextual texts from both sides. In addition to the MLM, BERT also uses a *"next sentence prediction (NSP)"* task that leans sentence-level representations. Thanks to those innovations and the huge size of model and data being used, BERT is proven to be impressively effective for transfer learning. With finetuning, BERT achieved amazing success by claiming the SOTA on almost all major NLP tasks, including both text and token classifications as well as text generation. It has been widely regarded as a new milestone for NLP, triggering a wave of transfer learning using pre-trained models. Also, there have been sizable research efforts related to its variants (e.g. by diminishing model size, like Albert (Lan et al. 2019); or improving model pretraining, like RoBERTa (Liu et al., 2019)), derivatives (e.g. extending BERT for specific tasks, such as BioBERT (Lee et al., 2019)), and interpretation (e.g. investigating its internal mechanism (Tenney et al. 2019)).

Training a large transformers model on massive data takes a prohibitively huge amount of computing resources. So, instead of training a transformers model from scratch, people usually use directly the public models already pretrained on huge raw texts data (and sometimes further trained on specific large labeled datasets of source tasks), as the backbone of their own model for downstream tasks.

3 Prompt-Based NLP Model for Targeted Sentiment Analysis

3.1 Related Works

Prompting Method

Using prompts to guide a PLM to perform different tasks is becoming a novel paradigm for effectively leveraging large PLMs. A prompt is a piece of text that we add to the PLM input so that the original task can be reformulated as a task that the PLM has already been pretrained on. As the example in Figure 1 shows, with the help of prompting, a sentiment analysis task can be restructured in a way similar to the MLM task, so that a model

pretrained with MLM task can perform sentiment analysis task directly without further task-specific training.

Prompting method was first popularized with the GPT-2 model (Radford et al., 2019) for zero-shot prediction, i.e. making prediction on a downstream task without training for this specific task. The enormous 1.5 billion parameters GPT-2 model was pretrained with 40GB of raw texts. Then the model parameters are frozen, and different prompts could be used to direct the model to perform different tasks including translation, reading comprehension, etc., without further tunning the model for those tasks. Yin et al. (2019) propose a prompting approach to reformulate text classification task as NLI task. They show that a BERT model further trained on NLI task can perform zero-shot classification. Beyond the zero-shot setting, Schick & Schütze (2021) introduce PET model that proposed further finetuning the model with prompt. Instead of freezing the PLM model, they further finetune the PLM's parameters with supervised training approach. While unlike traditional supervised learning using only input texts and labels, they add prompts specifically engineered for different tasks to guide the PLM to better leverage the patterns it leant during pretraining that are relevant to different tasks. Later, Gao et al. (2021) show that prompt-based finetuning of PLM on a small amount of labeled data can dramatically outperform standard finetuning. A formal definition of prompting method and a systematic survey are presented by Liu et al. (2021).

[Insert Figure 1 here]

Targeted Sentiment Analysis Models

The development of deep neural network enables researchers to build modern TSA models that are increasingly better at detecting this fine-grained sentiment. Tang et al. (2016) propose MemNet which adopts attention mechanism with external memory. It uses attention mechanism to explicitly model the target's relatedness to different parts of the texts semantically embedded in the memory. Wang et al. (2016) propose ATAE-LSTM which combines attention mechanism with LSTM. It concatenates target embedding with the representation of each word to let the aspect embedding play a role in computing attention weight. Chen et al. (2017) propose RAM model which uses bidirectional LSTM

to build memory from embeddings. The importance of different words in a sentence is weighted by their distance to the target word(s), closer words get higher weights. It also uses recurrent attention to focus on target-related information from memory.

More recently, transformers-based PLM have brought its success to TSA. Dai et al. (2021) show that fine-tuning a PLM on TSA task forces the PLM to implicitly learn more sentiment-word-oriented dependency trees compared to classical parser-provided dependency tree. Combining the induced tree with popular TSA models proves to elevate the performance to SOTA level. Tian et al. (2021) propose BERT-based TSA enhanced with word dependencies captured by an external key-value memory network (BERT-KVMN). They firstly extract the words associated to the target by parsing the dependency information of the sentence, then use KVMN to encode and weight such information to enhance TSA accordingly.

The novel prompting method is also being applied to TSA task. Seoh et al., 2021 build two different prompt-based models, one formulates the TSA task as a language modeling task, and use pretrained BERT or GPT-2 model as backbone model; the other formulates the TSA task as a NLI task, and used BERT further trained on NLI data as backbone model. Their approach proves to outperform standard supervised finetuned models for TSA.

3.2 Prompt-Based Targeted Sentiment Analysis Model

In this section we describe our method of prompt-based TSA using pretrained transformers model. We use BART-MNLI (Lewis et al., 2020), a powerful transformers model trained a large NLI task dataset, as the backbone of our model. In order to effectively leverage the language understanding capability of the backbone model, we design a prompt-based approach to reformulate the TSA task to imitate the NLI task. We further finetune our prompt-based model with labeled TSA data, to update the model's weights to be adapted to specific TSA tasks. Our model is different from the one in Seoh et al. (2021 in several important aspects. First, the model architectures for leveraging the inference prediction are different. We modify the BART-MNLI classification head to generate binary prediction during finetuning instead of keeping the three-class NLI

classification head. This enables our model to be more adapted to the actual TSA task with finetuning. Second, correspond to the model architecture, the prompting designs are different. We construct prompts for all three sentiment categories explicitly, instead of deducing the predictions from the inference results. Third, the backbone model we choose has better generalization ability and is more suitable for NLI task.

Backbone Model

Natural Language Inference: NLI is the problem of determining whether a text ("hypothesis") can logically be inferred from another text ("premise"). We call this inference relationship "entailment" if it's true, "contradiction" if it's false, or "neutral" if it's undetermined. For example, given the premise *"A child is playing football on the muddy playground in the rain."*, the relationship is entailment if the hypothesis is *"A person is playing sport outside amid bad weather."*, neutral if the hypothesis is *"A man loves football more than reading."*, and contradiction if the hypothesis is *"A man is afraid of getting wet in the rain."*. NLI task is a perfect testing ground for an NLP model's ability to capture the linguistic meanings of sentences. Hence a model trained to excel in NLI would be capable of extracting rich semantic representations of texts, which is a crucial basis for performing other downstream tasks.

There are two notable large public datasets for NLI, which have promoted a great amount of progress in NLP. The earliest one is the Stanford NLI (SNLI) corpus (Bowman et al., 2015), that contains 570K human-written hypothesis-premise pairs. To construct a set of pairs, a human annotator is given a true description of an image as the premise, and asked to come up with the three types of hypotheses: an alternate true description as entailment, a description that might be true as neutral, and a false description as contradiction. The way SNLI dataset is constructed constraint its text genre to be image captions which are descriptions of concrete visual scenes, thus lacking many important concepts such as time, mental states, etc. Modeled on the SNLI corpus, the later Multi-Genre NLI (MNLI) corpus (Williams et al., 2018) overcomes the earlier drawbacks by covering a wider range of genres of texts with different styles, formality, and topics. Its 433K human-annotated texts-pairs include both written texts like press releases, letters, fictions, travel guides,

etc., and spoken texts like face-to-face conversation, telephone transcripts, etc. The wide coverage of MNLI makes it a valuable source for training advanced NLP models that are good at domain adaptation and transfer-learning when solving different tasks in various domains.

BART-MNLI: BART is a transformers model developed by the Facebook AI team (Lewis et al., 2020). It adopts a standard sequence to sequence architecture, while creatively combines the bidirectional encoder of BERT and the autoregressive decoder of GPT. It is pretrained with a so-called text denoising task, which involves two steps: 1) corrupting the text with a noise function by masking arbitrary spans of words and randomly permuting sentences, and 2) letting the model learn to reconstruct the original text. Thanks to its special architecture and the pretraining task, which is proven to be very effective, BART achieves great performance on various common benchmarks for both text comprehension and text generation.

We adopt the BART-MNLI, i.e., BART model further trained on the MNLI dataset, as the backbone of our model, because of its proven capability of language understanding and domain generalization, as well as its easy public access[1]. During the training, BART model takes each input from MNLI in the form of a premise-hypothesis pair, adds a special token to separate the two, and appends another special token to mark the end of the sentence. The representation of the end of sentence token, EOS, is plugged into a classification head to make prediction.

Prompt Method and Model Design

For the model to effectively harness the inference capability of the backbone PLM, we follow two basic concepts when designing the prompt. First, the prompt must be constructed in a way that mimics the NLI in both form and logic. Second, the prompt needs to aim the model to perform inference on the specific aspect that we want to capture,

[1] We use the pretrained BART(large size)-MNLI model checkpoint available freely via the HuggingFace platform: https://huggingface.co/facebook/bart-large-mnli. Due to the sizes of the model and dataset, pretraining BART model on MNLI will take an immense computational resource, which is both impractical and unnecessary to do by our own.

i.e., sentiment in our case. Those lead to a basic cloze-style prompt design similar to the ones first proposed by Schick & Schütze (2021). When we construct a prompt in this way, we are actually imitating how human read a text and respond to the question of judging the sentiment. Furthermore, in order to direct the model to focus attention on the targeted entity, we also need to explicitly embed and indicate the target in the prompt.

The resulted prompt method is as follows: For the supervised training phase, given an input text with a specified target and a corresponding sentiment label, we construct three different prompts embedded with the target and separately with the three sentiments labels (see the illustrative example in Figure 2). We then assign a binary label to the inference relationship between the text and prompt pairs. Only the one embedded with the original sentiment label is true among the three prompts. Similar to the input of NLI data, we input the text-prompt pairs joined by special tokens to mark the boundaries, and take the embedding of the EOS token as the representation of the sentence pair. So, one row from the original training data will generate up to three labeled text-prompt pairs for the training. In the prediction phase, we want the model to predict the sentiment given an input text with a target. For that, we still construct the three different text-prompt pairs same as in the training, and let the model predict their individual probabilities of being true, then we take the sentiment label in the prompt with the highest probability as the sentiment prediction of the target.

[Insert Figure 2 here]

The model architecture is illustrated in Figure 3. The original BART-MNLI model consists of a two-layer three-class classification head appended to BART PLM. Except for changing the last layer of the classifier with a layer of binary output, we keep the all the rest pretrained parameters of the BART-MNLI model, for the sake of preserving as much the pretrained knowledge as possible.

[Insert Figure 3 here]

The above model input design can fit any TSA dataset that has a clear sentiment label for each observation, like the public SemEval2014 (Pontiki et al., 2014) and Twitter (Dong

et al., 2014) datasets. But in practice, when annotating a dataset with multiple people for each observation, different annotators will often have disagreement on the label, so that the original labeling is not clear-cut. Aggregating those disagreed labels into one hard label causes information loss. Now that we have the full original data of the labels made by multiple annotators on each text, we design a slightly different input to best leverage the extra information on our data. Instead of hard labeling an observation as clearly being of one sentiment label, we attribute to each label a probability score based on the percentage of annotators that vote for this label, as shown in the example in Figure 4. The soft label is then used to calculate the loss. This soft-label input method proves to generate better performance on our data.

[Insert Figure 4 here]

3.3 Experiment and Results

To test models' performance on general TSA, we use the three most widely adopted public benchmark datasets: the SemEval2014 laptops and restaurants review datasets (Pontiki et al., 2014), and the Twitter TSA dataset (Dong et al., 2014).

To test models' performance on financial TSA, we use the multiple-platform social media financial TSA data that we gathered, as well as the public FiQA dataset (Maia et al., 2018).[2]

Results on General Targeted Sentiment Analysis

We first train and test our model for general TSA. Table 1 shows the performance comparison between our prompt-based TSA model (Prompt-TSA) and other representative SOTA TSA models. Our model outperforms all the existing models on the three benchmarks. The results prove the strong ability of our model in performing the TSA task.

[2] Please see appendix for details on the model training implementation and hyperparameters.

[Insert Table 1 here]

Results on Financial Targeted Sentiment Analysis

We then train and test our model on two financial TSA datasets, including our own social media dataset, and the FiQA dataset. The FiQA dataset contains 436 news headlines and 675 social media posts from financial web pages, labeled with sentiment score targeted towards stocks (Maia et al., 2018). Our won social media dataset originally contains around 4K finance/investment-related social media posts from social media platforms, including StockTwits, Twitter, and Reddit. Each post is annotated by at least 5 people with adequate education background in business, as positive, negative, or neutral. Due to the ambiguous and noisy nature of the social media, the annotators often disagree on the labeling of the same sentence. To ensure the correctness of the labeling in our final sample, we only keep the posts for which over 80% of annotators agreeing on the same label. The size of our final sample is 1 355, which is comparable to that of the FiQA dataset.

Table 2 shows the performance of our Prompt-TSA model. We also tested two representative non-targeted financial sentiment models on those datasets. The first uses the Loughran & McDonald financial dictionary (LM Dictionary) (Loughran & McDonald, 2011); and the second uses FinRoBERTa (Jiang et al., 2022), financial PLM based on RoBERTa model pretrained on raw financial texts and finetuned on FPB dataset. Our model demonstrates superior performance with over 80% accuracy on both datasets, significantly outperforming the other two models.

Also, when comparing the cross test results, we notice that the model trained on our financial social media data shows better performance when applied to the FiQA dataset (acc = 68.80%) compared to the reverse (acc = 61.62%). This may indicate that our data has a higher training value that helps the model generalization.

[Insert Table 2 here]

4 Financial Implications of NLP-based Sentiment Analysis

4.1 Textual and Financial Data

We evaluate the proposed sentiment measure and its economic value empirically utilizing four different datasets for the time period between July 2020 and July 2022: 1) social media textual data from Twitter, Reddit and StockTwits for 24 meme stocks and 30 Dow Jones Industrial Average component stocks[3]; 2) traditional media articles (such as the Wall Street Journal) for the aforementioned meme and Dow Jones stocks; 3) intraday prices and volumes; and 4) daily Fama-French factors.

Using the application programming interfaces (APIs), we collect social media textual postings concerning the targeted companies from Twitter, Reddit, and StockTwits. For StockTwits and Twitter, by convention, a cashtag followed by a ticker is used as the keyword for identifying stock-related posts (e.g. $AAPL for investing-related discussion on Apple Inc.). For Reddit, however, the cashtag convention is not valid. Therefore, we first filter finance related posts by first restricting our download within selected investing-related subreddits[4], then we further filter the downloaded posts using NLP techniques to ensure the correct identification of the company names and tickers keywords[5]. In addition to the posted messages, all scraped social media postings include the date and timestamp information. Due to the unique design of Twitter and StockTwits, social postings from both platforms include the number of followers of the posters. Additionally, Twitter data includes the number of retweets for each post.

[3] See Appendix for the list of stocks we use in this paper. We select a meme stock if the stock appears in the monthly top 10 holdings of Roundhill MEME ETF at least twice during Dec 2021 (inception of the MEME ETF) to July 2022.

[4] The Reddit forum is composed of subreddits each devoted to a specific topic. We select 10 most influential finance/investing related subreddits based on the number of members and time of existence. Selected subreddits include: 'stocks', 'options', 'wallstreetbets', 'CanadianInvestor', 'SecurityAnalysis', 'InvestmentClub', 'RobinHood', 'investing', 'StockMarket', 'ValueInvesting'.

[5] When using company names and tickers as keyword to search and download posts on Reddit, the downloaded data could contain many irrelevant posts, because the simple keyword matching on Reddit API is case insensitive and superficial. For example, searching the meme stock tickers WISH will return many irrelevant posts containing the plain word "wish". As another example, searching the company name Apple with return irrelevant posts mentioning the fruit "Apple". To address this problem, first, for ticker based downloads, we filter them by requiring strict case-sensitive match of the ticker. Second, for name based downloads, we use named entity recognition to ensure that the keyword refers to a company instead of a generic meaning.

For our sample period, the meme stocks related (DJ30 stocks related) textual dataset includes 5.20 (2.34) millions Twitter posts, 11.68 (1.88) millions StockTwits posts, and 1.86 (0.44) millions Reddit posts, a daily average of 297 (107) tweets, 661 (86) stockstwits, and 106 (20) reddit posts for each meme stock (DJ30 stock), respectively. Using this dataset, we further measure the sentiment of every post using our proposed NLP learning model documented in Section 3. Generally, the sentiment of a post is expressed as a continuous numeric value between -1 (negative) and 1 (positive). We compute the daily sentiment based on the average sentiment[6] and disagreement based on the standard deviation of sentiment of all messages posted during a given period.

For the purpose of testing whether social media sentiment has additional effects on stock return and volatility beyond traditional media sentiment, we also used our proposed methodology to compute the financial sentiment embedded in the Wall Street Journal (WSJ) and use it as a control variable. The articles relating to the companies are gathered from the Factiva database. For all meme stocks (DJ30 stocks) during our sample period, we obtain 591 (2 366) articles in total, which equals to one (five) article(s) per day on average that can be used to compute the sentiment of traditional media.

In this study, we use the Trade and Quote (TaQ) dataset of the Wharton Research Data Services (WRDS) to determine the intraday price and volume. With intraday data, we calculate the daily return as the log difference between the close price of day t and that of $t-1$, and the realized volatility as the sum of squared 5-minute log return. The data of daily Fama-French 5 factors are from Kenneth French's web site[7].

[Insert Table 3 here]

From Table 3, we observe several interesting pieces of information. First, there seems to be no significant differences on both return and trading volumes between meme and DJ30 stocks. Overall, meme and DJ30 stocks combined has a return of 0.178 percent (with a standard deviation (SD) of 0.06) on daily basis. Meme stocks' daily return is positive at

[6] We also compute the followers weighted average sentiment for Twitter and StockTwits, and Retweets weighted average for Twitter.
[7] https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

0.4 percent (SD = 0.08) and DJ30 stocks' daily return is negative at -0.009 percent (SD = 0.02). The average daily trading volume for both types of stocks combined is 15.94 (millions shares, with SD = 1.10), with meme stocks traded a bit higher at 16.05 (SD = 1.28) and DJ30 stocks at 15.88 (SD = 0.95). However, the differences between both daily return and volume are not statistically significant. Second, compared to both daily return and daily trading volume, there are great variations in terms of the number of messages/posts on meme and DJ30 stocks. On average, there are 217 twitter posts (SD = 1472) each day mentioning either meme or DJ30 stocks. For meme Stocks, there are 493 (SD = 2245) posts, while there are only 27 (SD = 161) posts covering DJ30 stocks. Similarly, there are 89 (SD = 542) Reddit posts covering both types of stocks, with 207 (SD = 822) posts for meme stock and 4 (SD = 64) posts for DJ30 stocks. Considering 'meme' stock refers to a stock with dramatic price movement that is mainly caused by sentiment on social media posts, the differences on the number of posts regarding meme stock and DJ30 stock are not surprising.

4.2 Performance of the Proposed Sentiment Measure

In order to assess the performance of sentiment measures, we examine whether they can distinguish meme stocks from DJ30 stocks in market impact tests. In theory, meme stocks' return and volatility should be more sensitive to social media sentiment than DJ30 stocks' return and volatility. Table 4 and Table 5 illustrate the multivariate regression results on the impact of the proposed sentiment measurements on both stock return and stock volatility. In Table 4, the dependent variable is stock return. For each sentiment measure, we perform one regression model with control of stock trading volume, sentiment measure from traditional media WSJ, and five Fama French Factors (FF1-FF5). Models 1 to 3 use three different sentiment indexes measured from Twitter, models 4 and 5 use two different sentiment indexes measured from Stocktwits. Model 6 uses one sentiment index measured from Reddit. And model 7 adopts one sentiment index aggregating different sources using principal component analysis.

[Insert Table 4 here]

[Insert Table 5 here]

The results in Table 4 indicate that, in general, social media postings have a positive impact on stock returns. All social media sentiment indexes are positive and statistically significant at 1 percent. In term of economic value, one standard deviation of changes in social media sentiment generates 0.0038 in stock return. For meme stock, this impact is more significant. The coefficients of all social media sentiment indexes are higher for meme stocks. Economically, one standard deviation changes in social media sentiment generates 0.0286 (0.0038+0.0248) in meme stock return. Our results are consistent with our theoretical predictions that sentiment has a significant impact on stock price movement and this impact is stronger for meme stock. Besides the significant impact of social media sentiment, Table 4 also reveals some other important factors. Our results show that trading volumes are positively and statistically significantly associated with stock return. Nevertheless, sentiment generated from traditional media source WSJ is negative and statistically significant associated with stock returns.

Table 5 repeats all models in Table 4 with stock volatility as the dependent variable. Unlike the consistent and significant positive relationship found between sentiment and stock returns, the relationships between social sentiment and stock volatility are mixed, depending on social media platform. In general, results in Table 5 show a negative and statistically significant relationship between social media sentiment and stock volatility. Nevertheless, when the sentiment is measured using PCA data, our results show a positive and statistically significant relationship between social media sentiment and stock volatility. For meme stock, except for one sentiment measure from twitter, all social media sentiment shows a significantly negative relationship with meme stock volatility.

4.3 Trading Strategy Based on Social Media Sentiment

We further test if our new Prompt-TSA sentiment measure economically outperforms the measures by the other two existing representative financial sentiment models: FinRoBERTa and LM dictionary. To do so, we compare the average daily return derived from these three measures for both meme stocks and DJ30 stocks. More specifically, we conduct a daily basis trading strategy: at day t, we first compute the daily social media sentiment for Twitter, StockTwits, and Reddit. We use each social media sentiment together with other control variables to predict the next day return, and then buy (sell) if

the predicted return is greater (less) than zero [8]. We close our position at the next-day's market closing and repeat the same procedure for day $t + 1$.

We compare the average daily return of meme stocks for strategies based on sentiments measured by Prompt-TSA, FinRoBERTa, and LM dictionary. The results presented in Table 6 show that the daily average return derived from our new proposed model is statistically significantly higher than those based on FinRoBERTa and LM dictionary measures for meme stocks. When applying the same strategy to DJ30 stocks, we find that none of the 3 sentiment measurements generate positive return on average, and the results do not indicate significant difference in performance. This suggests social media sentiment as an investment signal is more applicable to meme stocks than to blue-chip DJ30 stocks.

[Insert Table 6 here]

[Insert Table 7 here]

5 Conclusion

In this study, we develop a cutting-edge NLP model for financial-oriented social media that can measure financial sentiment targeted toward specific firms within a text. The model architecture itself is domain-agnostic and demonstrates state-of-the-art performance on multiple benchmark datasets for targeted sentiment analysis in general domains including online reviews and Twitter sentiment. Further, based on the high-quality human-annotated social media targeted financial sentiment dataset that we created, we are able to finetune our model so that it measures targeted financial sentiment with high accuracy, outperforming two other representative existing financial sentiment models (that are not as sophisticated and are not specifically designed for TSA) by a large margin. Then, we test the financial implication of our new sentiment measure using 25 million social media posts from Reddit, Twitter and StockTwits. Those posts are filtered

[8] Given that social media postings can arrive any time during the day and market operation time is between 9:30 and 16:00, we choose 16:00 as our cut-off time. Therefore, the daily return and sentiment for day t are the return and sentiment between 16:00h on day $t - 1$ and 16:00h on day t.

to be finance / investing relevant, and to concern the 24 meme stocks or Dow30 stocks. In general, the sentiment measured by our model shows predictive power for short-term future return and volatility of the targeted companies' stock prices. Higher social media sentiment forecasts higher return and lower volatility. Moreover, consistent with our hypothesis, this relationship is stronger for meme stocks than for Dow30 stocks. To further compare our model with the other two existing financial sentiment models, we construct a trading strategy using the different sentiment measures in parallel. The strategy based on our sentiment measure has a higher return compared to the others, indicating that our model outperforms the other two existing models economically.

There are several aspects that can be explored for future research. The first two aspects are about extending the model, provided that we can get proper different labeled targeted sentiment data. First, since our model architecture is agnostic to text genre or domain, we can naturally extend its application to other financial texts beyond social media such as business news; or even extend to other domain beyond finance, for example, measuring consumer sentiment towards companies for marketing research. Second, besides targeting companies, by adjusting the prompt, the model could also be modified to target different types of entities or even aspects/topics. The last aspect is about enhancing the model architecture. We could use more sophisticated technics such as automated prompting search (Shin et al., 2020) instead of empirical prompt construction.

Appendix

Implementation Details

We used Huggingface and PyTorch for our model construction and training.

Key hyperparameters for training our model include:

- Use AdamW optimizer.
- Learning rate =1e-5, with a warmup ratio = 0.1, and linear decay scheduler.
- Effective batch size = 64.
- Train for 3 epochs.

Computation platform: Compute Canada Narval HPC[9] server equipped with Nvidia A-100 GPUs.

All model performance results reported on the benchmark TSA datasets and on financial TSA datasets are based on 5-fold cross-validation tests.

[9] https://docs.alliancecan.ca/wiki/Narval/en

Stocks Lists

	DJ30 Comonente Stocks		Meme Stocks	
	Ticker	Company name	Ticker	Company name
1	AXP	American Express Co	DWAC	Digital World Acquisition Corp
2	AMGN	Amgen Inc	SOFI	SoFi Technologies Inc
3	AAPL	Apple Inc	DKNG	DraftKings Inc
4	BA	Boeing Co	BB	BlackBerry Ltd
5	CAT	Caterpillar Inc	HOOD	Robinhood Markets Inc
6	CSCO	Cisco Systems Inc	ROKU	Roku Inc
7	CVX	Chevron Corp	AFRM	Affirm Holdings Inc
8	GS	Goldman Sachs Group Inc	UPST	Upstart Holdings Inc
9	HD	Home Depot Inc	LCID	Lucid Group Inc
10	HON	Honeywell International Inc	TDOC	Teladoc Health Inc
11	IBM	International Business Machines Corp	AMD	Advanced Micro Devices, Inc.
12	INTC	Intel Corp	NET	Cloudflare Inc
13	JNJ	Johnson & Johnson	CLF	Cleveland-Cliffs Inc
14	KO	Coca-Cola Co	SQ	Block Inc
15	JPM	JPMorgan Chase & Co	RBLX	Roblox Corp
16	MCD	McDonald’s Corp	WISH	ContextLogic Inc
17	MMM	3M Co	BYND	Beyond Meat Inc
18	MRK	Merck & Co Inc	RIVN	Rivian Automotive Inc
19	MSFT	Microsoft Corp	AMC	AMC Entertainment Holdings Inc
20	NKE	Nike Inc	COIN	Coinbase Global Inc
21	PG	Procter & Gamble Co	MSTR	MicroStrategy Inc
22	TRV	Travelers Companies Inc	SNAP	Snap Inc
23	UNH	UnitedHealth Group Inc	PLTR	Palantir Technologies Inc
24	CRM	Salesforce Inc	GME	GameStop Corp.
25	VZ	Verizon Communications Inc		
26	V	Visa Inc		
27	WBA	Walgreens Boots Alliance Inc		
28	WMT	Walmart Inc		
29	DIS	Walt Disney Co		
30	DOW	Dow Inc		

References

Antweiler, W., & Frank, M. Z. (2004). Is all that talk just noise? The information content of internet stock message boards. *The Journal of Finance*, *59*(3), 1259-1294.

Araci, D. (2019). FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *arXiv preprint arXiv:1908.10063*.

Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015, September). A large annotated corpus for learning natural language inference.*Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal.

Chen, P., Sun, Z., Bing, L., & Yang, W. (2017). Recurrent attention network on memory for aspect sentiment analysis. Proceedings of the 2017 conference on empirical methods in natural language processing, Copenhagen, Denmark.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014, oct). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar.

Cookson, J. A., Engelberg, J., & Mullins, W. (2022). Echo Chambers. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3603107

Cookson, J. A., Lu, R., Mullins, W., & Niessner, M. (2022). The Social Signal. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4241505

Cookson, J. A., & Niessner, M. (2019). Why Don't We Agree? Evidence from a Social Network of Investors. *The Journal of Finance*, *75*(1), 173-228.

Dai, J., Yan, H., Sun, T., Liu, P., & Qiu, X. (2021, Jun). Does syntax matter? A strong baseline for Aspect-based Sentiment Analysis with RoBERTa.*Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online.

Das, S. R., & Chen, M. Y. (2007). Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web. *Management Science*, *53*(9), 1375-1388.

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014). Adaptive recursive neural network for target-dependent twitter sentiment classification. Proceedings of the 52nd annual meeting of the association for computational linguistics (ACL),

Farzindar, A. A., & Inkpen, D. (2020). *Natural Language Processing for Social Media* (3 ed.). Springer Cham.

Gao, T., Fisch, A., & Chen, D. (2021, August). Making Pre-trained Language Models Better Few-shot Learners.*Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Online.

Goldberg, Y. (2017). Neural Network Methods for Natural Language Processing. *Synthesis Lectures on Human Language Technologies*, *10*(1), 1-309.

Hirschberg, J., & Manning, C. D. (2015). Advances in natural language processing. *Science*, *349*(6245), 261-266.

Hirshleifer, D. (2020). Presidential Address: Social Transmission Bias in Economics and Finance. *The Journal of Finance*, *75*(4), 1779-1831.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Comput.*, *9*(8), 1735–1780.

Huang, A. H., Zang, A. Y., & Zheng, R. (2014). Evidence on the Information Content of Text in Analyst Reports. *The Accounting Review*, *89*(6), 2151-2180.

Jiang, H., Liu, P., F. Roch, A., & Zhou, X. (2022). *Social Media and Bitcoin Price Dynamics*.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2019). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234-1240.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020, July). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension.*Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online.

Li, H. (2022). Language models: past, present, and future. *Communications of the ACM*, *65*(7), 56-63.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Loughran, T. I. M., & McDonald, B. (2011). When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), 35-65.

Maia, M., Handschuh, S., Freitas, A., Davis, B., McDermott, R., Zarrouk, M., & Balahur, A. (2018). *WWW'18 Open Challenge: Financial Opinion Mining and Question Answering* Companion Proceedings of the The Web Conference 2018, Lyon, France.

Malo, P., Sinha, A., Korhonen, P., Wallenius, J., & Takala, P. (2014). Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, *65*(4), 782-796.

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

Mikolov, T., Yih, W.-t., & Zweig, G. (2013). Linguistic regularities in continuous space word representations. Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies,

Ni, J., Li, J., & McAuley, J. (2019). Justifying recommendations using distantly-labeled reviews and fine-grained aspects. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP),

Pedersen, L. H. (2022). Game on: Social networks and markets. *Journal of Financial Economics*, *146*(3), 1097-1119.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Doha, Qatar.

Pérez-Mayos, L., Ballesteros, M., & Wanner, L. (2021, November). How much pretraining data do language models need to learn syntax?*Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* The 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic.

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014, August). SemEval-2014 Task 4: Aspect Based Sentiment Analysis.*Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)* 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language models are unsupervised multitask learners*. OpenAI. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

Renault, T. (2017). Intraday online investor sentiment and return patterns in the U.S. stock market. *Journal of Banking & Finance*, *84*, 25-40.

Rietzler, A., Stabinger, S., Opitz, P., & Engl, S. (2020). Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification. Proceedings of the 12th Language Resources and Evaluation Conference, Marseille.

Ruder, S., Peters, M. E., Swayamdipta, S., & Wolf, T. (2019). *Transfer Learning in Natural Language Processing* Proceedings of the 2019 Conference of the North, Minneapolis, Minnesota.

Schick, T., & Schütze, H. (2021, April). Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference.*Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* 16th Conference of the European Chapter of the Association for Computational Linguistics, Online.

Seoh, R., Birle, I., Tak, M., Chang, H.-S., Pinette, B., & Hough, A. (2021). Open Aspect Target Sentiment Classification with Natural Language Prompts. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Punta Cana / Online.

Shin, T., Razeghi, Y., Logan IV, R. L., Wallace, E., & Singh, S. (2020, November). AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts.*Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* The 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. Proceedings of the 2013 conference on empirical methods in natural language processing, Seattle, Washington.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. Advances in neural information processing systems, Cambridge, MA, USA.

Tang, D., Qin, B., & Liu, T. (2016, nov). Aspect Level Sentiment Classification with Deep Memory Network.*Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas.

Tian, Y., Chen, G., & Song, Y. (2021). Enhancing aspect-level sentiment analysis with word dependencies. Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Online.

Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016). Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 conference on empirical methods in natural language processing, Austin, Texas.

Williams, A., Nangia, N., & Bowman, S. (2018, June). A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference.*Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans.

Huang, A. H., Wang, H., & Yang, Y. (2022). FinBERT: A Large Language Model for Extracting Information from Financial Text. *Contemporary Accounting Research*.

Yin, W., Hay, J., & Roth, D. (2019). Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong.

List of Tables

Model	Laptop		Restaurant		Twitter	
	Acc	F1	Acc	F1	Acc	F1
BERT-ADA	80.23	75.77	86.22	79.79	-	-
FT-RoBERTa GRAT	83.33	79.95	87.52	81.29	75.81	74.91
BERT-large KVMN	80.41	77.38	86.88	80.92	75.14	73.68
BERT PLM	81.10	76.83	87.5	80.78	-	-
GPT-2 PLM	80.73	77.13	86.99	80.02	-	-
BERT NLI	77.58	73.18	85.07	77.53	-	-
Prompt-TSA	**83.86**	**80.61**	**88.30**	**81.42**	**76.59**	**75.07**

Table 1. Models' performance on general TSA datasets. The best performance for each dataset is indicated in bold. The results of the other models are taken as reported from the corresponding papers: BERT-ADA (Rietzler et al., 2020), FT-RoBERTa GRAT (Dai et al., 2021), BERT-large KVMN (Tian et al., 2021), BERT PLM, GPT-2 PLM, and BERT NLI (Seoh et al., 2021).

Model	Social Media		FiQA	
	Acc	F1	Acc	F1
Prompt-TSA (SclMd)	**83.17**	79.92	68.80	63.18
Prompt-TSA (FiQA)	61.62	59.35	**81.6**	78.3
LM Dictionary	39.48	39.32	34.10	34.12
FinRoBERTa (FPB)	40.37	40.35	50.98	51.71

Table 2. Models' performance on financial TAS datasets. The name in the parenthesis indicates the dataset on which the model is finetuned on.

	Total Stocks						Meme Stocks						DJ30 Stocks					
	Nb_Obs	Mean	Std	25%	50%	75%	Nb_Obs	Mean	Std	25%	50%	75%	Nb_Obs	Mean	Std	25%	50%	75%
Return	15,059	0.00	0.06	-0.01	-0.00	0.01	6,268	0.00	0.09	-0.02	-0.00	0.03	8,791	-0.00	0.02	-0.01	-0.00	0.01
Volume	15,059	15.95	1.10	15.14	15.85	16.65	6,268	16.05	1.28	15.11	15.99	16.98	8,791	15.88	0.95	15.16	15.79	16.47
Sentiment_Twitter_1	15,059	0.16	0.28	-0.05	0.13	0.38	6,268	0.38	0.19	0.27	0.41	0.51	8,791	-0.00	0.21	-0.13	-0.00	0.12
Sentiment_Twitter_2	15,059	0.16	0.44	-0.12	0.18	0.47	6,268	0.39	0.31	0.21	0.42	0.61	8,791	-0.01	0.44	-0.27	-0.01	0.24
Sentiment_Twitter_3	15,059	0.15	0.37	-0.08	0.16	0.41	6,268	0.36	0.26	0.19	0.37	0.54	8,791	0.00	0.37	-0.21	-0.00	0.21
Nb_Message_Twitter	15,059	217.53	1,472.59	3.00	34.00	154.00	6,268	493.04	2,245.87	74.00	164.00	332.00	8,791	21.10	161.08	-9.00	7.00	27.00
Sentiment_Stocktwits_1	15,059	0.07	0.26	-0.09	0.07	0.24	6,268	0.19	0.19	0.06	0.18	0.31	8,791	-0.02	0.28	-0.18	-0.02	0.14
Sentiment_Stocktwits_2	15,059	0.18	0.47	-0.12	0.21	0.54	6,268	0.45	0.32	0.27	0.50	0.68	8,791	-0.02	0.47	-0.29	-0.02	0.25
Nb_Message_Stocktwits	15,059	498.06	3,063.64	2.00	31.00	189.00	6,268	1,159.58	4,661.88	82.00	216.00	599.50	8,791	26.39	221.75	-5.00	5.00	21.00
Sentiment_Reddit_1	15,059	0.04	0.50	-0.19	0.03	0.30	6,268	0.12	0.34	-0.07	0.10	0.30	8,791	-0.01	0.59	-0.33	-	0.30
Nb_Message_Reddit	15,059	88.62	542.34	-	5.00	26.00	6,268	207.14	822.73	9.00	26.00	96.00	8,791	4.11	64.18	-3.00	1.00	5.00
Price range	15,059	0.04	0.05	0.02	0.03	0.05	6,268	0.06	0.06	0.04	0.05	0.07	8,791	0.02	0.01	0.01	0.02	0.02
Disagreement_Twitter_1	15,059	0.74	0.11	0.67	0.74	0.81	6,268	0.77	0.10	0.70	0.77	0.84	8,791	0.71	0.11	0.65	0.72	0.79
Disagreement_Twitter_2	15,059	0.63	0.17	0.51	0.63	0.75	6,268	0.66	0.16	0.56	0.67	0.79	8,791	0.60	0.17	0.50	0.59	0.72
Disagreement_Twitter_3	15,059	0.67	0.17	0.56	0.69	0.80	6,268	0.72	0.15	0.63	0.74	0.83	8,791	0.63	0.18	0.52	0.66	0.77
Disagreement_Stocktwits_1	15,059	0.80	0.15	0.73	0.85	0.91	6,268	0.88	0.07	0.85	0.90	0.93	8,791	0.75	0.16	0.65	0.77	0.87
Disagreement_Stocktwits_2	15,059	0.59	0.20	0.48	0.59	0.74	6,268	0.64	0.18	0.52	0.65	0.79	8,791	0.56	0.20	0.45	0.54	0.70
Disagreement_Reddit_1	15,059	0.78	0.32	0.76	0.89	0.94	6,268	0.84	0.23	0.83	0.90	0.94	8,791	0.74	0.37	0.67	0.88	0.95
Price volatility	15,059	0.00	0.02	0.00	0.00	0.00	6,268	0.00	0.03	0.00	0.00	0.00	8,791	0.00	0.00	0.00	0.00	0.00

Table 3. Descriptive statistics. This table presents the descriptive statistics of all variables used in this paper for 30 DowJones and 24 meme stocks between July 2020 and March 2022. *Return* and *Volume* are stocks' daily return and trading volume, respectively. *Sentiment_Twitter_1*, *Sentiment_Twitter_2*, and *Sentiment_Twitter_3* are equally-weighted, followers-weighted, and retweets-number weighted Twitter sentiments. *Sentiment_Stocktwits_1* and *Sentiment_Stocktwits_2* are equally-weighted and followers-weighted Stocktwits sentiments. *Sentiment_Reddit_1* is for equally-weighted Reddit sentiment. *Nb_Message* is the corresponding number of messages for each source. *Price_Range* and *Price_volatility* are daily price range and 5minute realised volatility based on intraday transactions. 25%, 50%, and 75% relate to the first, the second, and the third quartile, respectively.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)
S_Twitter_1	0.0498***						
	(13.2455)						
MmS_Twitter_1	0.1327***						
	(14.6748)						
S_Twitter_2		0.0112***					
		(12.0159)					
MmS_Twitter_2		0.0266***					
		(24.3864)					
S_Twitter_3			0.0113***				
			(6.5413)				
MmS_Twitter_3			0.0416***				
			(16.2421)				
S_Stocktwits_1				0.0229***			
				(6.4512)			
MmS_Stocktwits_1				0.1983***			
				(17.5822)			
S_Stocktwits_2					0.0058***		
					(6.3975)		
MmS_Stocktwits_2					0.0087***		
					(5.2196)		
S_Reddit_1						0.0029***	
						(3.8774)	
MmS_Reddit_1						0.0212***	
						(13.7659)	
S_PCA							0.0038***
							(5.9399)
MmS_PCA							0.0248***
							(15.0278)
Volume	0.0219***	0.0226***	0.0208***	0.0221***	0.0213***	0.0224***	0.0227***
	(38.9710)	(61.4118)	(23.5073)	(28.2971)	(22.7930)	(38.2882)	(33.3196)
WSJ	-	-	-	-	-0.0013**	-	-
	0.0019***	0.0019***	0.0017***	0.0017***		0.0020***	0.0018***
	(-2.7806)	(-4.5665)	(-3.8770)	(-3.6724)	(-2.4670)	(-5.4811)	(-4.0568)
FF1	-	-	-	-	-	-	-
	0.0026***	0.0019***	0.0018***	0.0020***	0.0014***	0.0014***	0.0017***
	(-11.7955)	(-14.1803)	(-10.5128)	(-12.9558)	(-6.1174)	(-6.3102)	(-7.9092)
FF2	0.0031***	0.0034***	0.0029***	0.0029***	0.0030***	0.0032***	0.0034***
	(12.1840)	(15.1271)	(8.5825)	(12.1932)	(9.9798)	(11.9348)	(15.8107)
FF3	-	-	-	-	-	-	-
	0.0033***	0.0035***	0.0032***	0.0026***	0.0026***	0.0031***	0.0035***
	(-9.0942)	(-11.8679)	(-8.3225)	(-5.2025)	(-4.9995)	(-7.8322)	(-11.0296)
FF4	-	-	-	-	-	-	-
	0.0069***	0.0078***	0.0068***	0.0064***	0.0066***	0.0071***	0.0074***
	(-15.5544)	(-18.2520)	(-10.9025)	(-10.6967)	(-9.5207)	(-13.4346)	(-15.0101)
FF5	0.0136***	0.0147***	0.0134***	0.0131***	0.0132***	0.0132***	0.0143***
	(15.7694)	(19.5966)	(15.2149)	(12.0934)	(13.0858)	(13.4211)	(18.9979)
Constant	-	-	-	-	-	-	-
	0.3595***	0.3614***	0.3324***	0.3580***	0.3390***	0.3557***	0.3587***
	(-38.7835)	(-60.8446)	(-23.1281)	(-27.9950)	(-22.5001)	(-38.9730)	(-30.8695)
Observations	11,372	11,372	11,372	11,372	11,372	11,372	11,372
Num of stockid	54	54	54	54	54	54	54

Table 4. Marginal effect of social media sentiment on return. The table presents the marginal impact of social media on meme stocks' daily return. $S\_Twitter(Stocktwits/$

$Reddit$). means the sentiment measured from Twitter (Stocktwits/Reddit). Prefix Mm means the corresponding sentiment multiplied by a dummy variable indicating if the stock is a meme stock. Models (1) - (3) are about equally-weighted, followers-weighted, and retweets-number weighted Twitter sentiments, models (4) - (5) are about equally-weighted and followers-weighted Stocktwits sentiments. Model (6) is for equally-weighted Reddit sentiment. Finally, model (7) reports the results of main component of all sentiments from different sources. *Volume* is the daily trading volume. *WSJ* and *MarketRet* are the Wall Street Journal sentiment and market return, respectively. t-statistics are in parentheses. ***, **, and * represent statistical significance at the 1%, 5%, and 10% levels, respectively.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)
STwitter_1	-0.0073***						
	(-13.3516)						
MmS_Twitter_1	-0.0092***						
	(-13.9704)						
STwitter_2		-0.0191***					
		(-118.7260)					
MmS_Twitter_2		0.0153***					
		(97.0768)					
STwitter_3			-0.0020***				
			(-6.8683)				
MmS_Twitter_3			-0.0026***				
			(-8.0058)				
SStocktwits_1				-0.0032***			
				(-10.9896)			
MmS_Stocktwits_1				-0.0027***			
				(-6.1723)			
SStocktwits_2					-0.0010***		
					(-8.7874)		
MmS_Stocktwits_2					-0.0005***		
					(-4.3333)		
SReddit_1						-0.0002	
						(-1.6361)	
MmS_Reddit_1						-0.0016***	
						(-10.0668)	
S_PCA							0.0047***
							(44.2295)
MmS_PCA							-0.0067***
							(-68.6025)
Volume	0.0116***	0.0013***	0.0116***	0.0116***	0.0116***	0.0116***	0.0013***
	(305.8988)	(920.2016)	(201.7366)	(340.6712)	(384.5871)	(260.7405)	(772.5265)
WSJ	0.0066***	-0.0050***	0.0063***	0.0064***	0.0064***	0.0065***	0.0212***
	(74.2768)	(-111.9606)	(23.5152)	(43.5483)	(26.1650)	(26.1458)	(460.4797)
MarketRet	0.0005***	0.0009***	0.0004***	0.0005***	0.0004***	0.0004***	0.0011***
	(39.4674)	(71.5498)	(39.1871)	(31.8869)	(34.6334)	(35.1245)	(99.1528)
Constant	-0.1829***	-0.1945***	-0.1831***	-0.1829***	-0.1835***	-0.1836***	-0.1687***
	(-264.2760)	(-11.9023)	(-178.3267)	(-304.9551)	(-342.6001)	(-232.2836)	(-7.4360)
Observations	11,372	11,372	11,372	11,372	11,372	11,372	11,372
Number of stockid	54	54	54	54	54	54	54

Table 5. Marginal effect of social media on volatility. The independent variables are the same as in table 4.

	Prompt-TSA						FinRoBERTa						LM Dictionary					
	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit
DWAC	1.53	0.47	0.81	0.91	0.17	0.33	0.24	0.05	-0.55	-0.31	-0.46	0.41	1.38	-0.24	0.14	0.33	0.47	-0.20
SOFI	0.31	-0.14	0.17	-0.02	0.02	0.10	0.28	-0.07	-0.30	0.21	0.11	0.26	-0.19	0.06	0.03	0.27	2.72	0.52
DKNG	-0.22	-0.26	-0.16	0.41	0.01	-0.14	-0.35	-0.34	-0.33	-0.24	-0.25	0.29	-0.28	-0.32	-0.26	-0.04	-0.23	-0.20
BB	0.97	-0.05	0.09	0.17	0.60	-0.12	-0.27	-0.38	-0.27	-0.24	-0.10	0.00	0.00	-0.31	-0.32	0.03	-0.09	-0.24
HOOD	-0.06	-0.54	0.17	0.48	0.48	0.28	0.75	-0.38	-0.46	-0.30	0.08	2.20	-0.39	-0.17	-0.35	-0.46	-0.36	-0.29
ROKU	0.42	0.18	0.07	-0.02	-0.17	0.76	-0.19	-0.20	0.04	-0.35	-0.28	1.02	-0.05	-0.19	-0.18	-0.11	-0.21	-0.13
AFRM	1.77	0.44	-0.20	0.01	0.14	1.89	-0.26	-0.41	-0.16	-0.04	-0.16	1.95	-0.39	-0.26	-0.40	-0.16	0.75	-0.05
UPST	-0.22	0.29	0.99	-0.12	-0.38	0.63	0.65	-0.24	-0.24	-0.17	-0.22	-0.27	-0.32	-0.35	-0.29	-0.23	-0.30	-0.18
LCID	-0.16	-0.34	0.37	-0.22	-0.19	-0.42	1.21	0.88	1.14	1.57	0.16	0.50	-0.47	-0.16	-0.31	0.47	0.16	0.27
TDOC	0.07	0.14	-0.14	0.17	-0.17	0.31	-0.40	-0.18	-0.36	-0.22	-0.37	-0.26	-0.11	-0.04	-0.24	-0.19	0.01	-0.18
AMD	-0.14	0.17	0.25	0.13	0.24	-0.14	0.00	-0.16	-0.18	-0.21	-0.29	-0.25	-0.01	-0.24	-0.23	0.07	-0.03	-0.21
NET	-0.21	-0.27	0.28	0.57	-0.11	0.07	-0.22	-0.27	-0.34	-0.31	-0.28	0.10	0.01	-0.30	0.11	-0.03	-0.24	-0.23
CLF	-0.07	0.58	-0.14	-0.11	-0.12	1.05	-0.07	-0.27	-0.08	-0.19	0.13	0.33	0.06	-0.26	0.17	1.01	0.18	-0.06
SQ	0.45	-0.03	-0.13	-0.04	-0.11	0.02	0.09	-0.08	-0.29	-0.19	-0.20	0.22	-0.02	-0.07	-0.20	-0.28	-0.18	-0.15
RBLX	0.57	-0.15	-0.19	-0.16	-0.22	0.64	-0.12	-0.09	-0.30	-0.15	0.25	-0.25	-0.28	-0.23	-0.21	0.26	-0.20	-0.24
WISH	0.26	-0.29	-0.24	-0.19	-0.26	-0.07	-0.10	-0.25	0.12	-0.34	-0.34	0.18	0.15	0.00	-0.06	-0.21	-0.01	-0.27
BYND	-0.14	0.27	1.33	-0.19	0.17	0.10	-0.17	0.01	-0.16	-0.38	-0.33	-0.14	-0.28	-0.32	-0.32	0.06	-0.22	-0.22
RIVN	-0.32	0.49	-0.27	-0.57	1.84	-0.65	-0.62	-0.33	-0.24	-0.07	2.08	1.14	-0.47	0.96	0.30	-0.20	-0.51	-0.06
AMC	-0.03	-0.19	-0.03	-0.32	-0.27	-0.33	-0.31	-0.35	-0.35	-0.55	-0.30	0.83	-0.35	-0.35	-0.35	-0.35	-0.99	-0.31
COIN	-0.15	-0.19	0.26	-0.11	-0.03	0.38	0.23	0.04	-0.05	0.08	0.06	0.05	0.07	0.20	0.03	-0.05	0.06	-0.19
MSTR	-0.17	0.22	0.00	0.41	-0.15	0.74	-0.83	-0.80	-1.01	1.55	1.23	0.03	-0.39	-0.36	-0.39	1.19	-0.27	-0.28
SNAP	0.30	-0.15	1.69	1.07	0.26	-0.17	-0.17	-0.04	-0.24	0.60	0.58	0.11	0.25	0.04	-0.26	-0.03	0.40	0.45
PLTR	-0.15	-0.12	0.15	0.23	0.32	-0.28	0.14	-0.09	0.04	0.63	0.37	0.00	-0.22	-0.12	-0.28	-0.17	-0.21	-0.32
GME	2.41	-0.25	0.25	-0.31	-0.09	-0.33	-0.35	-0.35	-0.35	-0.52	-0.35	-0.35	-0.34	-0.48	-0.36	-0.36	11.22	-0.36
Average	0.29	0.01	0.22	0.09	0.08	0.19	-0.04	-0.18	-0.21	-0.01	0.05	0.34	-0.11	-0.15	-0.18	0.03	0.50	-0.13
p_value_1	0.082	0.055	0.002	0.466	0.696	0.348												
p_value_2	0.010	0.056	0.002	0.592	0.418	0.016												

Table 6. Comparison of returns derived from different sentiments for meme stocks. The table compares the average daily return (in percentage) of meme stocks for strategies based on sentiments issued from Prompt-TSA, FinRoBERTa, and LM dictionary. *Twitter_1*, *Twitter_2*, and *Twitter_3* are strategies based on equally-weighted, followers-weighted, and retweets-number weighted Twitter sentiments. *Stocktwits_1* and *Stocktwits_2* are strategies based on equally-weighted and followers-weighted Stocktwits sentiments. *Reddit_1* is strategy based on equally-weighted Reddit sentiment. *p_value_1* is the p_value for the test of mean equality between Prompt-TSA and FinRoBERTa. *p_value_2* is the *p_value* for the test of mean equality between Prompt-TSA and LM Dictionary.

	Prompt-TSA						FinRoBERTa						LM Dictionary					
	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit	Twitter_1	Twitter_2	Twitter_3	Stocktwits_1	Stocktwits_2	Reddit
AXP	-0.01	0.07	0.03	0.10	-0.01	0.09	0.05	-0.05	-0.05	0.18	0.13	0.21	-0.02	0.05	-0.07	-0.05	-0.04	0.05
AMGN	0.21	-0.04	0.11	-0.07	0.02	0.00	-0.01	0.00	-0.12	0.05	-0.09	-0.19	-0.05	-0.10	-0.04	-0.07	-0.18	-0.05
AAPL	0.03	-0.18	0.14	0.54	-0.08	0.21	-0.13	-0.07	-0.04	0.00	-0.11	-0.04	-0.09	-0.10	-0.08	-0.14	-0.09	-0.13
BA	-0.18	0.02	0.02	-0.08	0.09	-0.03	-0.11	-0.16	-0.04	-0.14	-0.13	-0.04	-0.09	-0.06	-0.11	-0.12	-0.01	-0.07
CAT	-0.07	-0.09	-0.03	-0.12	-0.08	0.05	-0.18	-0.15	-0.11	-0.09	-0.11	0.04	-0.06	-0.11	-0.07	-0.11	-0.11	0.00
CSCO	-0.12	-0.07	-0.13	-0.13	-0.02	-0.09	-0.05	-0.03	-0.04	-0.10	-0.12	-0.08	-0.01	-0.07	-0.06	-0.10	-0.01	0.01
CVX	-0.06	0.11	-0.06	-0.08	-0.09	-0.08	0.01	-0.02	0.06	-0.13	-0.14	-0.05	-0.05	0.01	-0.08	-0.05	-0.06	-0.01
GS	-0.14	0.12	-0.07	0.05	-0.15	-0.05	-0.07	-0.04	-0.11	-0.07	-0.08	-0.03	-0.08	-0.08	-0.08	-0.03	-0.07	-0.07
HD	-0.15	-0.14	-0.15	0.01	-0.10	0.00	-0.11	-0.02	-0.11	-0.04	-0.03	-0.08	-0.09	-0.12	-0.14	-0.05	-0.10	-0.11
HON	0.04	-0.03	-0.03	-0.02	-0.07	-0.06	-0.08	-0.01	-0.14	0.00	0.06	-0.14	-0.04	-0.09	-0.05	-0.02	-0.07	-0.09
IBM	0.04	-0.05	-0.11	0.00	-0.03	0.04	-0.05	-0.11	-0.08	-0.06	0.01	-0.08	-0.09	-0.10	-0.11	0.01	-0.10	-0.02
INTC	-0.05	0.06	0.14	-0.14	-0.01	-0.04	-0.16	-0.16	-0.15	-0.09	-0.08	-0.13	-0.10	-0.06	-0.15	-0.05	-0.12	0.02
JNJ	-0.19	-0.16	-0.16	-0.17	-0.16	-0.14	-0.04	-0.02	-0.05	0.01	0.02	-0.06	-0.03	-0.07	-0.06	-0.03	-0.06	-0.07
KO	-0.05	-0.06	-0.05	-0.13	-0.11	-0.08	-0.08	-0.11	-0.07	0.00	-0.04	0.05	-0.05	-0.05	-0.05	-0.07	-0.11	-0.08
JPM	-0.01	-0.13	0.14	-0.12	0.09	0.06	-0.12	-0.04	-0.06	0.00	-0.05	-0.07	-0.12	-0.07	-0.11	-0.07	0.00	-0.08
MCD	-0.12	-0.17	-0.14	-0.18	-0.11	-0.10	-0.11	-0.11	-0.10	-0.05	-0.05	-0.02	-0.05	-0.06	-0.07	-0.07	-0.04	-0.06
MMM	-0.01	-0.08	-0.14	-0.13	-0.14	-0.13	-0.13	-0.10	-0.09	-0.08	-0.09	-0.02	-0.09	-0.12	-0.09	-0.08	-0.05	0.03
MRK	-0.04	-0.04	-0.07	-0.02	-0.09	-0.09	-0.12	-0.07	0.01	-0.03	-0.03	0.03	-0.08	-0.08	-0.12	-0.06	-0.04	0.02
MSFT	-0.14	-0.10	0.02	-0.05	-0.15	-0.09	-0.09	-0.04	-0.06	-0.09	-0.07	-0.07	-0.03	-0.07	-0.04	-0.11	-0.03	-0.09
NKE	-0.01	-0.09	0.01	0.11	-0.03	-0.03	-0.09	-0.15	-0.11	-0.02	-0.05	-0.06	-0.09	-0.06	-0.07	-0.16	-0.09	-0.10
PG	-0.10	-0.03	0.00	-0.03	-0.13	-0.08	-0.08	-0.10	-0.10	0.01	0.03	0.07	-0.04	-0.08	-0.08	-0.03	-0.03	-0.07
TRV	-0.42	-0.17	-0.47	-0.36	-0.49	-0.32	-1.08	0.02	-0.38	-0.36	-0.36	0.27	-0.30	-0.08	-0.25	0.09	-0.09	0.00
UNH	0.06	0.04	-0.12	0.01	-0.01	-0.05	0.12	0.18	0.20	-0.03	-0.02	0.04	-0.06	-0.09	-0.08	-0.02	-0.07	0.02
CRM	0.29	0.04	-0.07	0.19	0.08	0.16	-0.19	0.09	-0.11	-0.18	-0.09	0.10	-0.11	-0.09	-0.04	-0.09	-0.11	-0.05
VZ	-0.06	-0.13	-0.08	-0.06	-0.03	-0.08	-0.02	-0.02	0.02	0.04	0.00	0.00	-0.07	-0.04	-0.05	-0.07	-0.01	-0.04
V	-0.06	-0.04	-0.15	0.04	-0.07	-0.05	-0.02	-0.07	-0.05	0.01	0.02	-0.03	-0.07	-0.01	-0.13	-0.10	-0.10	-0.05
WBA	0.00	0.03	0.03	-0.06	0.08	0.03	0.06	0.08	-0.01	-0.04	0.00	0.03	-0.10	-0.07	-0.06	-0.14	-0.13	-0.16
WMT	-0.17	-0.12	-0.11	-0.17	-0.13	-0.01	-0.09	-0.09	-0.10	-0.05	-0.09	-0.12	-0.11	-0.11	-0.08	-0.10	-0.02	-0.08
DIS	-0.08	-0.18	0.10	0.00	-0.07	0.00	-0.10	-0.08	-0.05	-0.09	-0.06	-0.03	-0.06	0.00	0.02	-0.08	0.04	-0.08
DOW	-0.09	-0.06	-0.02	-0.12	-0.19	-0.01	-0.23	-0.17	-0.29	-0.25	-0.27	-0.20	-0.10	-0.06	0.01	-0.02	-0.07	-0.13
Average	-0.06	-0.06	-0.05	-0.04	-0.07	-0.03	-0.11	-0.05	-0.08	-0.06	-0.06	-0.02	-0.08	-0.07	-0.08	-0.07	-0.07	-0.05
p_value_1	0.087	0.955	0.221	0.562	0.583	0.764												
p_value_2	0.308	0.435	0.108	0.409	0.771	0.405												

Table 7. Comparison of returns derived from different sentiment for DJ30 stocks. The table compares the average daily return (in percentage) of DowJones30 stocks for strategies based on sentiments issued from Prompt-TSA, FinRoBERTa, and LM dictionary. The same as in table 6, *Twitter_1*, *Twitter_2*, and *Twitter_3* are strategies based on equally-weighted, followers-weighted, and retweets-number weighted Twitter sentiments. *Stocktwits_1* and *Stocktwits_2* are strategies based on equally-weighted and followers-weighted Stocktwits sentiments. *Reddit_1* is strategy based on equally-weighted Reddit sentiment. *p_value_1* is the p_value for the test of mean

equality between Prompt-TSA and FinRoBERTa. *p_value_2* is the p_value for the test of mean equality between Prompt-TSA and LM Dictionary.

List of Figures

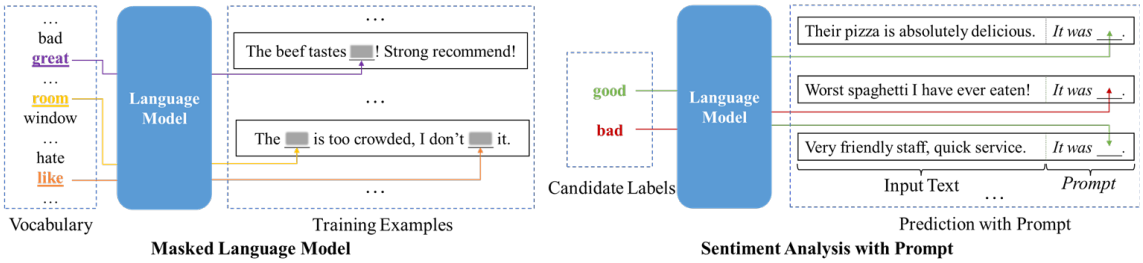

Figure 1. Example of prompting method. The left illustrates the masked language model (MLM) pretraining: we randomly mask words for a given training example, and let the PLM predict which words in the vocabulary are most likely to be the masked words. The right illustrates performing sentiment analysis with prompt based on the PLM: for an input text, we append a prompt *"It was ___."* to its end, and let the PLM fill in the empty slot with the most probable word from the candidate labels. The word chosen can be converted to the prediction about the sentiment of the input text.

Text	Target	Label
Apple beats Microsoft.	Apple	positive

Original Data

Text	Prompt	Label
Apple beats Microsoft.	The sentiment towards *Apple* is **positive**.	True
	The sentiment towards *Apple* is **neutral**.	False
	The sentiment towards *Apple* is **negative**.	False

Text-Prompt Pairs with Binary Label

Figure 2. Construction of prompts from the original labeled training data. The prompt is formatted as an explicit statement of targeted sentiment, "The sentiment towards [*target*] is [**label**].", with clozes filled by the target (shown in italic) and the candidate sentiment labels (shown in bold).

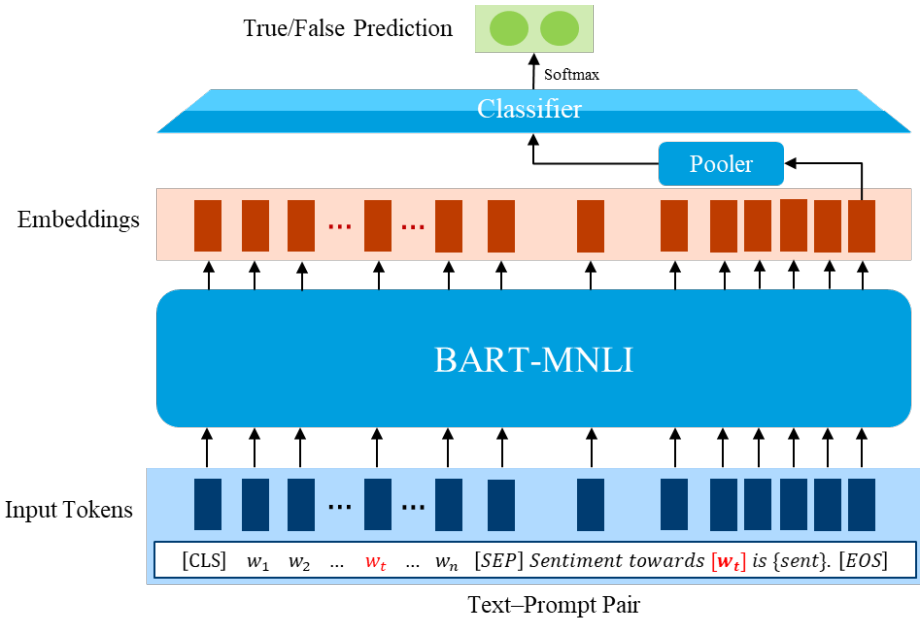

Figure 3. Illustration of the prompt-based targeted sentiment model. The backbone model is BART-MNLI. The last layer of the classifier is modified to make binary instead of 3-class classification. The input of to the model is in the form of text-prompt pair joined by special tokens, displayed in token level. The token in red color represents the targeted word / phrase, which appears in both the input text and the prompt formed. The embedding of the EOS token is inputted into the classifier to make prediction.

Text	Target	Label Vote
Apple beats Microsoft.	Apple	positive: 4 neutral: 1 negative: 0

Original Data

Text	Prompt	Probability
Apple beats Microsoft.	The sentiment towards *Apple* is **positive**.	0.8
	The sentiment towards *Apple* is **neutral**.	0.2
	The sentiment towards *Apple* is **negative**.	0

Text-Prompt Pairs with Soft Label

Figure 4. Prompts construction with soft-label based on the full original annotation data. The "Label Vote" column denotes how many people voted for each label. The probability score is calculated as the number of votes divided by total number of annotators.