# Fine-Tuning Large Language Models for Stock Return Prediction Using Newsflow

**Tian Guo** and **Emmanuel Hauptmann**
Systematic Equities Team, RAM Active Investments
Geneva, Switzerland
{tig, eh}@ram-ai.com

## Abstract

Large language models (LLMs) and their fine-tuning techniques have demonstrated superior performance in various language understanding and generation tasks. This paper explores fine-tuning LLMs for predicting stock returns with financial newsflow. Return prediction is fundamental for quantitative investing tasks like portfolio construction and optimization. We formulate the model to include a text representation and forecasting modules. We propose to compare the encoder-only and decoder-only LLMs, considering they generate text representations in distinct ways. The impact of these different representations on return forecasting remains an open question. Meanwhile, we compare two simple methods of integrating LLMs' token-level representations into the forecasting module. The experiments on real investment universes reveal that: (1) aggregated representations from LLMs' token-level embeddings generally produce return predictions that enhance the performance of long-only and long-short portfolios; (2) in the relatively large investment universe, the decoder LLMs-based prediction model leads to stronger portfolios, whereas in the small universes, there are no consistent winners; (3) return predictions derived from LLMs' text representations are a strong signal for portfolio construction, outperforming conventional sentiment scores. These findings suggest the potential of LLM fine-tuning for enhancing return prediction-based portfolio construction.

## 1 Introduction

Quantitative investing relies on extracting quantitative features or signals from various data sources including market prices, economic indicators, financial text, etc., to build and optimize investment portfolios (Fama and French, 1996; Ang, 2014). In recent years, the use of text data for quantitative investing has grown significantly, thanks to the advancement of natural language processing (NLP) techniques (Xu and Cohen, 2018; Sawhney et al., 2020; Qin and Yang, 2019). In particular, large language models (LLMs) have demonstrated superior performance on various language understanding and generation tasks (He et al., 2021; BehnamGhader et al., 2024; Jiang et al., 2023; Touvron et al., 2023; Dubey et al., 2024), and the fine-tuning technique allows for adapting the pre-trained LLMs to fit investing-related applications (Hu et al., 2021; Ding et al., 2023).

This paper[1] is focused on return prediction with financial news for stock portfolio construction. Return forecasting is useful for picking stocks with profit potentials to include in portfolios. Financial news reports on events and announcements related to companies, industries, the economy, etc., and shows notable predictive power for stock future performance in previous studies (Liu et al., 2018; Hu et al., 2018; Guo et al., 2020).

The conventional way of applying financial news data to stock picking involves a multi-step extraction-and-validation process as illustrated in Fig. 1(a), i.e., formulating the numerical features (e.g., sentiments, popularity, etc.) with the expectation that these features have a predictive relationship with stock future performance (e.g., forward return, volatility, etc.) (Allen et al., 2019; Shapiro et al., 2022), developing the feature extraction process (e.g., train a financial sentiment classification model), and validating the predictive power of extracted features by statistical analysis or building forecasting models. This process might be time-consuming and require additional data (e.g., labeled sentiment data) and continuous refinements.

LLMs generate numerical representations (or embeddings) of text that capture semantic relations, and these representations can naturally serve as features for forecasting tasks. Based on this in-

[1] A preprint version of this paper appeared at https://arxiv.org/abs/2407.18103



*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1028–1045
November 12-16, 2024 

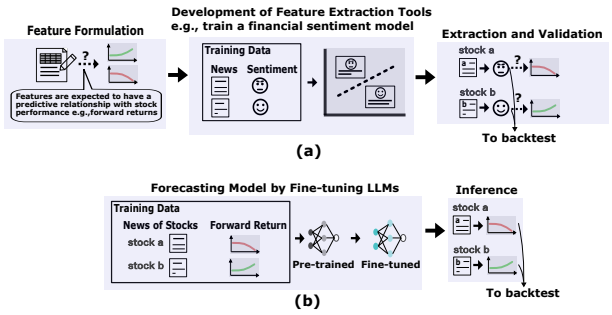


Figure 1: Comparison of different workflows of utilizing financial news for stock picking. (a) Conventional feature extraction-and-validation process, e.g., financial sentiments. (b) News-to-return forecasting by fine-tuning LLMs.

tuition, this paper explores direct news-to-return prediction through fine-tuning LLMs. Fig. 1 illustrates the difference between the conventional feature extraction-and-validation process and our LLM-based news-to-return process. Though some previous works attempted to use text embedding for forecasting (Liu et al., 2018; Wang et al., 2019; Qin and Yang, 2019; Guo et al., 2020), few works have explored the potential of fine-tuning LLMs for stock return forecasting with newsflow. Moreover, this paper has the contribution as follows:

- We design an LLM-based return prediction model comprising the text representation and the forecasting modules.
- We hypothesize that the text representations from encoder-only and decoder-only LLMs perform differently due to their distinct methods of encoding sequences in pre-training and fine-tuning; thus, we propose to compare the encoder-only (DeBERTa) and decoder-only LLMs (Mistral, Llama3) as the representation module of the prediction model.
- Considering that LLM-generated text representations are at the token level, we present two simple methods to integrate token representations into the forecasting module: bottleneck and aggregated representations.
- We perform experiments on real financial news and various investment universes. In addition to evaluating prediction errors, we assess two types of portfolios built on return predictions through backtesting in out-of-sample periods. The experimental comparison between encoder-only and decoder-only LLMs and between bottleneck and aggregated representations offers insights for identifying suitable text representations for different investing strategies and markets.

## 2 Related Work

Numerous works have investigated using financial text data for forecasting tasks. (Weng et al., 2018; Xu and Cohen, 2018) extracted the sentiment score from financial newsflow, social media, and tweets for stock price predicting. (Liu et al., 2018; Hu et al., 2018) explored learning numeric representations of financial news by attention mechanisms for modeling stock movements. (Wang et al., 2019) studied combining sentiment and text representations for return prediction.

The advent of LLMs and related techniques provides a new powerful way of using text data for forecasting tasks in quantitative investing (Zhao et al., 2023; Li et al., 2023). Encoder-only models such as BERT (Devlin et al., 2019) and DeBERTa (He et al., 2020, 2021), focus on learning contextual embeddings for input text. Decoder-only models like GPT-3 (Radford et al., 2018) and Mistral (Jiang et al., 2023) are trained to generate text by predicting the next token in a sequence.

LLMs are pre-trained on vast amounts of text data to learn general language patterns. The prompt technique is to design specific inputs to guide the pre-trained LLM to produce the desired output without modifying the LLM's parameters (Radford et al., 2019; Brown et al., 2020; Kojima et al., 2022). Fine-tuning techniques adjust the pre-trained LLM's parameters to adapt to specific tasks (Gunel et al., 2020; Wei et al., 2021; Ding et al., 2023; Chung et al., 2024). In particular, parameter-efficient fine-tuning techniques have gained popularity (Hu et al., 2021; Ding et al., 2023; Liu et al., 2024).

Some recent works use LLMs as feature extractors to obtain predictive signals from text. (Araci, 2019; Liu et al., 2021) explored the fine-tuning of pre-trained LLMs to provide more accurate financial sentiment analysis. Instead of fine-tuning LLMs, (Wang et al., 2024) extracted factors from the financial news and price history by prompts on generative LLMs. (Kim et al., 2024) used chain-of-thought prompts (Wei et al., 2022) on generative LLMs to analyze financial statements. (Li et al., 2024) fine-tuned LLMs for generating text responses of prediction and explanations.

Unlike existing works that extract features from text using LLMs, this paper focuses on fine-tuning LLMs to directly model the relationship between financial news text and numerical return values. Meanwhile, we evaluate the text representations from different types of LLMs to study their different effectiveness for the return forecasting task.

## 3 From Financial Newsflow to Stock Portfolios through LLMs

### 3.1 Problem Statement

Assume an investment universe consisting of a set of stocks $\mathcal{U} = \{s\}_{s=1}^{S}$, where $s$ represents the stock index. In quantitative investing, the stock-picking process selects a subset of the universe as the investing portfolio based on quantitative criteria. As market conditions and various information change, the stock-picking process is repeatedly performed to update or rebalance the portfolios at (regular) time intervals, e.g., weekly, monthly, etc.

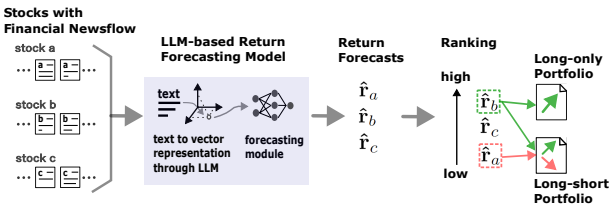


Figure 2: Illustration of the LLM-based return forecasting model for the stock-picking process. Assume an investment universe of 3 stocks denoted by $a, b, c$. Each stock has an associated list of news. Then, given the return forecasts and ranks, stocks can be selected into long-only or long-short portfolios.

Let $r_{s,t+\ell} \in \mathbb{R}$ be the $\ell$-step forward return of stock $s$ w.r.t. timestep $t$. The textual content of news reported at time $i$ and w.r.t. stock $s$ is denoted by $\mathbf{x}_{s,i}$, a list of text tokens. At time $t$, the news text available for predicting $r_{s,t+\ell}$ in a look-back time window $W$ is $\{\mathbf{x}_{s,i}\}_{i \in \mathcal{T}_{s,<t}}$ where $\mathcal{T}_{s,<t}$ represents the set of timesteps of available news.

Considering the large sequence length that LLMs can process nowadays (Zhao et al., 2023; Li et al., 2023), we concatenate the set of news in the look-back window into one sequence denoted by $\mathbf{X}_{s,<t} = \oplus\{\mathbf{x}_{s,i}\}_{i \in \mathcal{T}_{s,<t}}$, where $\oplus$ denotes the concatenation operation. Next, we formulate the return forecasting model as a composite structure of a text representation module and a forecasting module as defined in Eq. 1:

$$\hat{r}_{s,t+\ell} = f \circ g\left(\mathbf{X}_{s,<t}\right) \tag{1}$$

We aim to explore realizing Eq. 1 by jointly fine-tuning a pre-trained LLM as $g(\cdot)$ and training a dense layer as $f(\cdot)$. In particular, Eq. 1 is a sequence-level task requiring the text representation module $g\colon \mathbf{X}_{s,<t} \mapsto \mathbf{h}_{s,<t}$ to encode the sequence $\mathbf{X}_{s,<t}$ into a numerical vector $\mathbf{h}_{s,<t} \in \mathbb{R}^D$. Then, the forecasting module $f\colon \mathbf{h}_{s,<t} \mapsto \hat{r}_{s,t}$ transforms $\mathbf{h}_{s,<t}$ to the return forecast. We train the model using a set of data instances pooled from individual stocks and associated news, i.e., $\{(r_{s,t+\ell}, \mathbf{X}_{s,<t})\}_{s \in \mathcal{U}, t \in \mathcal{T}}$ where $\mathcal{T}$ represents the timestamps in the training period.

At test time, besides evaluating prediction errors such as the root mean square error (RMSE), we implement the return prediction-based stock picking to construct long-only and long-short portfolios which are subsequently backtested. This process is illustrated in Fig. 2.

Long-Only Portfolios are intended to include stocks with the expectation of a price rise above the universe average. In practice, it is built by ranking the stocks based on the return forecasts and selecting the top-K stocks. $K$ is usually chosen according to the decile or quantile of the universe, e.g., $10\%$ of the total number of stocks.

Long-Short Portfolios include both the stocks with the expectation of a price rise and drop. For the stocks with a price drop expectation, the portfolio can profit by selling them at the present price and repurchasing them at a lower price in the future. In this paper, the long-short portfolio is built by including the top-K and bottom-K stocks based on the forecast ranks.

### 3.2 Methodology

LLMs can be categorized into three main types: encoder-only, decoder-only, and the hybrid encoder-decoder. All these LLMs transform text into high-dimensional vector representations, however, their different pre-training objectives lead to text representations with varying implications.

In the following, we describe the text representation difference in encoder-only and decoder-only LLMs. Then, we present two simple methods of integrating the token-level representations from LLMs into the forecasting module. These methods introduce no additional parameters to learn and provide a clear comparison of the native representations of different LLMs for return forecasting.

**Encoder-only vs. Decoder-only LLMs.** Given a sequence of text tokens $\mathbf{X} = \{x_1, \cdots, x_L\}$, LLMs output a sequence of vector representations

$\{\mathbf{h}_1, \cdots, \mathbf{h}_L\}$ corresponding to the input tokens. However, as presented below, the vector representations from encoder-only and decoder-only LLMs encode the different parts of the input sequence.

Pre-training an encoder LLM is mostly based on masked-language modeling (Devlin et al., 2019; Lan et al., 2019; He et al., 2020). Concretely, it prepares a training text sequence $\mathbf{X}$ by randomly masking some tokens, leading to $\hat{\mathbf{X}} = \{x_{\text{mask}} \text{ if } i \in \mathcal{M} \text{ else } x_i \; \forall i = 1, \cdots, L\}$. $\mathcal{M} \subset \{1, \cdots, L\}$ represents the indices of tokens to mask. The mask token $x_{\text{mask}}$ is a special token without concrete meaning and plays as the placeholder. The pre-training objective is to predict masked tokens, i.e., maximizing the likelihood of masked tokens as:

$$\begin{aligned} &\log p\big(\{x_m\}_{m\in\mathcal{M}} \,|\, \hat{\mathbf{X}}\big) \\ &= \sum_{m\in\mathcal{M}} \log p(x_m \,|\, \mathbf{X}_{<m}, x_{\text{mask}}, \mathbf{X}_{>m}) \\ &\approx \sum_{m\in\mathcal{M}} \log p(x_m \,|\, \mathbf{h}_m) \end{aligned} \tag{2}$$

In Eq. 2, $\mathbf{X}_{<m} = \{x_1, \cdots, x_{m-1}\}$ and $\mathbf{X}_{>m} = \{x_m, \cdots, x_L\}$ represent the tokens before and after $x_m$. Maximizing Eq. 2 encourages the representation $\mathbf{h}_m$ to incorporate both the left and right contexts, i.e., $\mathbf{X}_{>m}$ and $\mathbf{X}_{<m}$, for predicting the masked token. Particularly, in the attention mechanism of Transformers, $\mathbf{h}_m$ is derived based on the similarities between the mask token $x_{\text{mask}}$ and the context tokens $\mathbf{X}_{>m}$ and $\mathbf{X}_{<m}$.

On the other hand, a decoder-only LLM models an input sequence autoregressively using the next-token prediction task (Radford et al., 2018; Touvron et al., 2023). The pre-training objective function is defined in Eq. 3:

$$\begin{aligned} &\log p(x_1, \cdots, x_L|\check{\mathbf{X}}) \\ &= \sum_{i=1,\cdots,L} \log p(x_i \,|\, \mathbf{X}_{<i}) \\ &\approx \sum_i \log p(x_i \,|\, \mathbf{h}_{i-1}) \end{aligned} \tag{3}$$

For modeling the first token, the practical way is to add a Beginning-of-Sequence (BOS) token, i.e., $\check{\mathbf{X}} = x_{\text{bos}} \oplus \mathbf{X}$. Similar to the mask token, the BOS token has no concrete meaning. The representation $\mathbf{h}_{i-1}$ encodes the information from already seen tokens and is derived based on the relation between $x_{i-1}$ and $\mathbf{X}_{<i-1} = \{x_1, \cdots, x_{i-2}\}$.

**Bottleneck vs. Aggregated Representations.** As LLMs output the token-level vector representations, to obtain a representation encoding the sequence, the idea of bottleneck representation is to push LLMs to compress the sequence information into a single vector representation during fine-tuning (Yang et al., 2019; Wang et al., 2023a,b).

In practice, this is achieved by appending an End-of-Sequence (EOS) $x_{\text{EOS}}$ to the input sequence, e.g., $\mathbf{X}_{s,<t} \oplus x_{\text{EOS}}$. As $x_{\text{EOS}}$ is constant across sequences, its vector representation $\mathbf{h}_{\text{EOS}}$ depends on the real tokens of the sequence. During fine-tuning, $\mathbf{h}_{\text{EOS}}$ is fed into the forecasting module as shown in Eq. 4. The backpropagation process propels $\mathbf{h}_{\text{EOS}}$ to summarize real tokens's representations through the forecasting module.

$$\hat{r}_{s,t+\ell} = f(\mathbf{h}_{\text{EOS}}) \tag{4}$$

The bottleneck representation has different implications for encoder-only and decoder-only LLMs. In encoder-only LLMs, the vector used for predicting is obtained based on the mask token and the real context tokens during the pre-training, as explained in Eq. 2. As a result, appending an EOS token (identical to the mask token used in pre-training) aligns the fine-tuning with the pre-training. This consistency might facilitate the EOS token representation to summarize sequence-level features effectively. In decoder-only LLMs, the vector representation of each token is conditioned on the already-seen tokens; thus, the last token of a sequence naturally summarizes the whole sequence, making an additional EOS token redundant.

Meanwhile, considering the recent works on the representation collapse issue of the last token in certain conditions (Barbero et al., 2024), we present a simple alternative to bottleneck representation, i.e., allowing the forecasting module to aggregate the representations of all tokens. This can be done using various methods like averaging, or sophisticated ones like attention mechanisms (Lee et al., 2024). In this paper, we choose the simple averaging method, since it introduces no additional parameters to train and enables a clear comparison with the bottleneck representation.

$$\hat{r}_{s,t+\ell} = f\left(\frac{1}{L}\sum_l \mathbf{h}_l\right) \tag{5}$$

For encoder-only LLMs, the pre-training and fine-tuning discrepancy arises when using aggregated representations, because each token's representation is based on context and itself, instead of the mask token in pre-training. For decoder-only

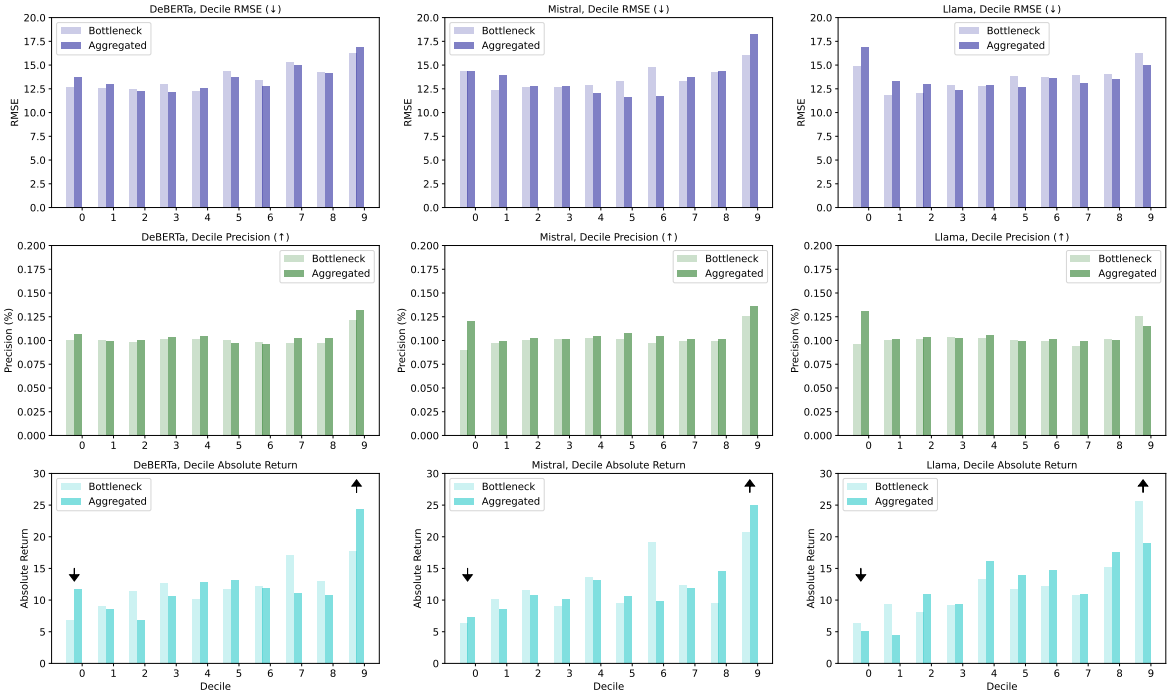


Figure 3: Decile Performance of Bottleneck and Aggregated Representations in the North American Universe (best viewed in color). Top Row: Decile RMSE. Middle Row: Decile Precision. Bottom Row: Decile Return. The up (or down) arrow indicates the higher (or lower) values are desirable.

LLMs, averaging all representations might lead to bias towards the early tokens of the input sequence. This is because, in the autoregressive setting, the early tokens are repeatedly incorporated into the representations of all subsequent ones.

**Implementations.** We experiment with one encoder-only LLM DeBERTa (He et al., 2021) and two decoder-only LLMs, Mistral-7B and Llama3-8B base models (Jiang et al., 2023; Dubey et al., 2024) and use the mean squared error (MSE) as the loss function. More details are in the Appendix.

## 4 Experiments

In this part, we present some main results, while further details and a qualitative interpretation of predictions are provided in the Appendix.

**Data.** We use company-level financial newsflow data from 2003 to 2019 provided by a financial data vendor. Each piece of news has an attribute including the company identifier(s) the news is primarily about. Meanwhile, we have three investment universe datasets of the North American (NA), European (EU), and Emerging (EM) markets.

**Setup.** The long-only portfolio is built by taking the stocks with the return predictions falling in the top (9th) decile of prediction rankings. The long-short portfolios take the stocks in the top (9th) and bottom (0th) deciles. The stocks in all portfolios are equally weighted.

We perform backtesting to evaluate the portfolios in monthly rebalancing. Besides comparing the portfolios built on return predictions by different LLMs, we also compare them with the sentiment-based portfolio construction by FinBERT (Araci, 2019) and FinVADER (Hutto and Gilbert, 2014; Korab, 2023). The sentiment-based portfolios are built using the same method but with sentiment values as the ranking criteria.

**Metrics.** As mentioned in the problem statement of Sec. 3.1, the downstream stock picking for building portfolios is based on the deciles of forecasts; thus we report three decile-wise metrics to align with downstream scenarios, i.e., decile RMSE, decile precision, and decile return. For portfolio backtesting, we report the cumulative return charts and performance statistics like annualized returns and Sharpe ratios in the testing period.

**Results.** In the following, we mainly present and discuss the results of the NA universe. The results of the EU and EM universe are in the Appendix.

Bottleneck vs. Aggregated Representations: In Fig. 3, we compare the bottleneck and aggregated representations for the three LLMs in the North American universe through the decile RMSE, pre-

Table 1: Statistics of Portfolios in the North American Universe. The Universe Equally-Weighted represents the universe performance reported under the Long-only Portfolio column.

| | Long-only Portfolio | | Long-short Portfolio | |
|---|---|---|---|---|
| | Ann. Return % (↑) | Sharpe Ratio (↑) | Ann. Return % (↑) | Sharpe Ratio (↑) |
| Universe Equally-Weighted | 9.76 | 0.68 | – | – |
| Sentiment_FinVader | 12.26 | 0.72 | 2.92 | 0.39 |
| Sentiment_FinBert | 20.64 | 1.22 | 8.81 | 0.92 |
| DeBERTa_Bottleneck | 17.47 | 0.96 | 10.83 | 0.94 |
| DeBERTa_Aggregated | 25.15 | 1.20 | 12.87 | 1.07 |
| Mistral_Bottleneck | 21.27 | 1.15 | 15.08 | 1.49 |
| Mistral_Aggregated | 25.38 | 1.12 | 18.30 | 1.26 |
| Llama_Bottleneck | 27.00 | 1.32 | 20.46 | 1.49 |
| Llama_Aggregated | 18.86 | 1.00 | 14.29 | 1.30 |

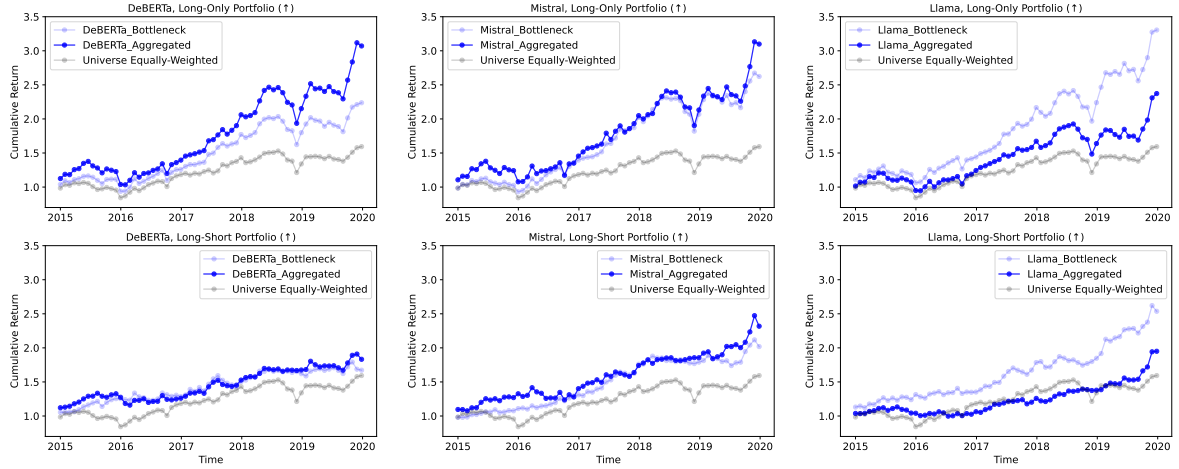


Figure 4: Cumulative Return Charts of the Portfolios based on Bottleneck and Aggregated Representation Models in the North American Universe (best viewed in color). Top Row: Long-only Portfolios. Bottom Row: Long-short Portfolios.

cision, and returns. Each column of Fig. 3 corresponds to a LLM. Meanwhile, Fig. 4 shows the cumulative return charts of portfolios and Table 1 reports the detailed performance stats of portfolios.

In the bottom row of Fig. 3, the returns from the 0th decile to the 9th decile generally present an upward trend, implying that the return predictions are generally aligned with actual future performance. We are particularly interested in the top 9th and bottom 0th deciles as they are the main constituents of portfolios. For the top 9th decile, the aggregated representation model generates a higher return and benefits the long portfolio, except for Llama. For the EU and EM universe, as presented in the Appendix, the aggregated representation model consistently outperforms the bottleneck one.

Interestingly, the higher returns do not necessarily imply low RMSE in the 9th decile. For instance, in Fig. 3, the aggregated representation model has a higher decile return, but a higher RMSE, in the 9th decile corresponding to the long-only portfolio for DeBERTa and Mistral. An explanation is that the 9th decile is regarding predicting high-value returns and less accurate predictions of these returns might have high RMSE. But, if the return prediction still falls into the 9th decile as the true return, the corresponding decile return is retained. In this case, the decile precision is more indicative of the decile return, for instance, in Fig. 3 the outperforming representations mostly have a higher precision in the 9th decile.

As for the bottom 0th decile, a lower return is preferred as the short side of a long-short portfolio benefits from stocks with underperforming forward returns. In Fig. 3, the aggregated representation model falls short of lowering the 0th decile's return for DeBERTa and Mistral, however, Table 1 shows that the return and Sharpe ratios of long-short portfolios are mostly improved with aggregated representations compared to the bottleneck

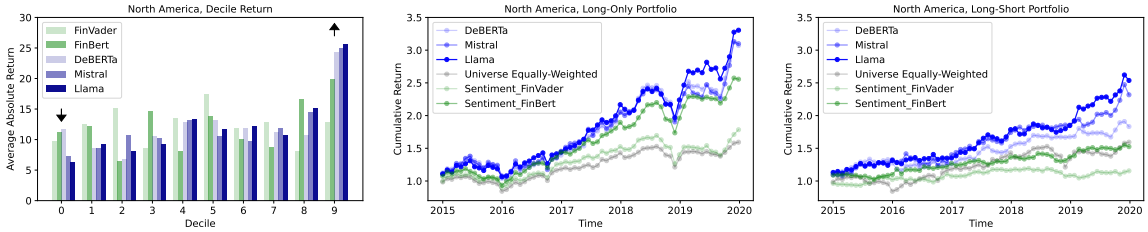


Figure 5: Comparison with Sentiment-based Portfolios in the North American Universe (best viewed in color).

representations.

Fig. 4 visualizes the cumulative return of the portfolios using the bottleneck and aggregated representation models. The performance of long-only and long-short portfolios correspond to the top and bottom deciles in Fig. 3. The return curves of the aggregated representation model are notably higher except for Llama. In the Appendix, the aggregated representation constantly outperforms the bottleneck representation for the EU and EM universes.

Encoder-only vs. Decoder-only LLMs: Fig. 5 shows the comparison of encoder-only and decoder-only LLMs with the suitable representations for the NA universe, i.e., the aggregated representation for DeBERTa and Mistral, and the bottleneck representation for Llama. For the EU and EM universes in the Appendix, the aggregated representation is favored for all three LLMs.

The decile return in Fig. 5 exhibits that decoder-only Mistral and LLama generate high returns in the top 9th decile and lower returns in the bottom 0th decile, thereby leading to the outperforming long-only and long-short portfolios as shown in the cumulative return charts. The performances of long-only portfolios are comparable among encoder and decoder LLMs, however, in long-short portfolios, the short side drags down the performance of the long side, especially for the encoder-only DeBERTa. This highlights the importance of effective stock selection on both sides of the portfolio. Meanwhile, all the prediction-based portfolios yield higher returns than the universe average.

Prediction-based vs. Sentiment-based Portfolios: In this part, we compare the prediction-based portfolios with conventional sentiment-based portfolios. Fig. 5 shows the decile returns and the return charts of portfolios, and the performance statistics are in Table 1.

In Table 1, the prediction-based long-only and long-short portfolios outperform the sentiment-based portfolios in returns and Sharp ratios. In Fig. 5, the return charts of prediction-based portfolios are above the sentiment-based portfolios. In particular, for the long-short portfolios, as shown in the return chart, the short side of the sentiment-based method negatively offsets the long side, leading to underperformance w.r.t. the universe. In contrast, the prediction-based long-short portfolios have smoother return curves than the long-only portfolios, because the short side mitigates the overall portfolio's volatility. The outperformance of prediction-based portfolios suggests that the return prediction models capture more relevant information from text representations for future stock performance, leading to effective stock picking.

## 5 Conclusion

This paper focuses on return forecasting with financial newsflow for quantitative portfolio construction. Unlike the conventional feature extraction-and-validation workflow, this paper explores fine-tuning LLMs to directly model the relationship between news text and stock forward return.

The experiment results reveal the key findings: (1) aggregated representations from LLMs' token-level embeddings generally produce the return predictions that enhance the portfolio performance; (2) in the relatively large investment universe, the decoder LLMs-based prediction model leads to stronger portfolios, whereas in the small universes, there are no consistent winners. (3) return predictions derived from LLMs' text representations are a strong signal for portfolio construction, outperforming conventional sentiment scores.

Several open questions remain for future research. For instance, it is unclear whether the underperformance of encoder-only DeBERTa in the large universe is due to the model size or other factors, and why DeBERTa has varying performance in different small universes. Evaluating recently proposed large encoder-only LLMs (Wang et al., 2023b; BehnamGhader et al., 2024) would be an interesting follow-up.

## References


David E Allen, Michael McAleer, and Abhay K Singh. 2019. Daily market news sentiment and stock prices. *Applied Economics*, 51(30):3212–3235.

Andrew Ang. 2014. *Asset management: A systematic approach to factor investing*. Oxford University Press.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Federico Barbero, Andrea Banino, Steven Kapturowski, Dharshan Kumaran, João GM Araújo, Alex Vitvitskyi, Razvan Pascanu, and Petar Veličković. 2024. Transformers need glasses! information over-squashing in language tasks. *arXiv preprint arXiv:2406.04267*.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Eugene F Fama and Kenneth R French. 1996. Multifactor explanations of asset pricing anomalies. *The journal of finance*, 51(1):55–84.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Tian Guo, Nicolas Jamet, Valentin Betrix, Louis-Alexandre Piquet, and Emmanuel Hauptmann. 2020. Esg2risk: A deep learning framework from esg news to stock volatility prediction. *arXiv preprint arXiv:2005.02527*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 261–269.

Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 216–225.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Alex Kim, Maximilian Muhn, and Valeri V Nikolaev. 2024. Financial statement analysis with large language models. *Chicago Booth Research Paper Forthcoming, Fama-Miller Working Paper*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Petr Korab. 2023. Finvader: Financial sentiment analysis. `https://github.com/PetrKorab/FinVADER`.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

Xiang Li, Zhenyu Li, Chen Shi, Yong Xu, Qing Du, Mingkui Tan, and Jun Huang. 2024. Alphafin: Benchmarking financial analysis with retrieval-augmented stock-chain framework. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 773–783.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. Large language models in finance: A survey. In *Proceedings of the fourth ACM international conference on AI in finance*, pages 374–382.

Qikai Liu, Xiang Cheng, Sen Su, and Shuguang Zhu. 2018. Hierarchical complementary attention network for predicting stock price movements with news. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1603–1606.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2021. Finbert: A pre-trained financial language representation model for financial text mining. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, pages 4513–4519.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Yu Qin and Yi Yang. 2019. What you say and how you say it matters: Predicting financial risk using verbal and vocal cues. In *57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, page 390.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, and Rajiv Shah. 2020. Deep attentive learning for stock movement prediction from social media text and company correlations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8415–8426.

Adam Hale Shapiro, Moritz Sudhof, and Daniel J Wilson. 2022. Measuring news sentiment. *Journal of econometrics*, 228(2):221–243.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2023a. Simlm: Pre-training with representation bottleneck for dense passage retrieval. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2023b. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*.

Meiyun Wang, Kiyoshi Izumi, and Hiroki Sakaji. 2024. Llmfactor: Extracting profitable factors through prompts for explainable stock movement prediction. *arXiv preprint arXiv:2406.10811*.

Yaowei Wang, Qing Li, Zhexue Huang, and Junjie Li. 2019. Ean: Event attention network for stock price trend prediction based on sentimental embedding. In *Proceedings of the 10th ACM Conference on Web Science*, pages 311–320.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Bin Weng, Lin Lu, Xing Wang, Fadel M Megahed, and Waldyn Martinez. 2018. Predicting short-term stock prices using ensemble methods and online data sources. *Expert Systems with Applications*, 112:258–273.

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 (Volume 1: Long Papers)*, pages 1970–1979.

Linyi Yang, Ruihai Dong, Tin Lok James Ng, and Yang Xu. 2019. Leveraging bert to improve the fears index for stock forecasting. In *Proceedings of the First Workshop on Financial Technology and Natural Language Processing*, pages 54–60.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

# A Appendix

## A.1 Experiment Details

**Implementations.** The text representation module and the forecasting module are respectively initialized by a pre-trained LLM and a dense layer. Then, the training process jointly fine-tunes the LLM and learns the forecasting module to minimize the mean squared error (MSE) between the forecasts and true values. We applied Low-Rank Adaptation (LoRA) to fine-tune LLMs (Hu et al., 2021). Other techniques including gradient checkpointing, mixed precision training, and DeepSpeed are used to reduce GPU memory (Rasley et al., 2020).

We experiment with one encoder-only LLM, i.e., DeBERTa (He et al., 2021), and two different decoder-only LLMs, i.e., Mistral-7B and Llama3-8B base models (Jiang et al., 2023; Dubey et al., 2024). DeBERTa is a recent encoder-only LLM that improves upon the BERT model with disentangled content and position embeddings. Mistral-7B is a 7-billion-parameter decoder-only LLM that uses grouped query and sliding window attention to improve performance. Llama3-8B is an 8-billion-parameter decoder-only LLM pre-trained on data mixed from different sources, e.g., multilingual, codes, etc., to improve the generalization ability.

**Data.** We use company-level financial newsflow data from 2003 to 2019 provided by a financial data vendor. Each piece of news has an attribute including the company identifier(s) the news is primarily about. Meanwhile, we have three investment universe datasets of the North American (NA), European (EU), and Emerging (EM) markets, which consist of dates, stock identifiers, and the true monthly forward returns of corresponding stocks and dates. The training and validation data is from 2003 to 2014 for each universe, while the rest is for the out-of-sample testing data. Each instance is built by linking an entry in the universe data to related news through the stock identifier and a look-back time window (e.g., one week). Table 2 shows the data stats.

Table 2: Statistics of Datasets.

| Universe | # of Stocks | Average # of News per Instance | # of Training Instances | # of Validating Instances | # of Testing Instances |
|---|---|---|---|---|---|
| North America | 630 | 2.5 | 366011 | 10167 | 241367 |
| Europe | 350 | 1.9 | 100403 | 10041 | 121705 |
| Emerging Markets | 370 | 2.6 | 71610 | 10231 | 183608 |

Note that our news data is predominantly about company-specific events, e.g., earnings reports, analyst revisions, analyst ratings, earnings outlooks, management changes, etc, and is less directly about macro economy. In this case, our prediction model is primarily designed to capture the impact of these company events on stock returns, rather than to learn broader economic processes.

**Preprocessing.** Our data preprocessing follows standard procedures and primarily involves tasks like cleaning (e.g., removal of special spaces, newlines, and empty content) and joining the news articles with their corresponding target variables.

In our dataset, the longest token length is approximately 2,100, while the average token length is around 108. For the three LLMs used in the paper, we set a consistent maximum token length of 4096 during fine-tuning. This length is selected because it accommodates the longest token length in our dataset, ensuring that no truncation was required for the LLMs in our experiments.

**Setup.** We train the model only once and then apply the model to obtain the return predictions in the testing period. We conduct the model training using a batch size of 32, a learning rate of $1e$-5, and a warmup phase of 100 steps followed by a linear decay. To fine-tune LLMs, we applied Low-Rank Adaptation (LoRA) with rank 4 to all linear layers. We employ a maximum context length of 4k for all LLMs used in experiments. All models are trained for 10 epochs on 2 A100 GPUs.

The long-only portfolio is built by taking the stocks with the return predictions falling in the top (9th) decile of prediction rankings. The long-short portfolios take the stocks in the top (9th) and bottom (0th) deciles. The stocks in all portfolios are equally weighted.

We perform backtesting to evaluate the portfolios in monthly rebalancing. It stimulates the trading of monthly constructed portfolios and reports the cumulative return chart and performance statistics

like annualized returns and Sharpe ratios in the testing period. When backtesting the long-only and long-short portfolios, besides comparing the portfolios built on return predictions by different LLMs, we also compare them with the sentiment-based portfolio construction. Specifically, FinBERT is a fine-tuned BERT (Bidirectional Encoder Representations from Transformers) for financial sentiment analysis (Araci, 2019). FinVader is a dictionary-based method with a financial sentiment lexicon (Hutto and Gilbert, 2014; Korab, 2023). The sentiment-based portfolios are built using the same method but with sentiment values as the ranking criteria.

**Metrics.** As mentioned in the problem statement of Sec. 3.1, the downstream stock picking for building portfolios is based on the deciles of forecasts; thus we report three decile-wise metrics to align with downstream scenarios, i.e., decile RMSE, decile precision, and decile return. The decile return is the actual return of stocks allocated to the decile based on predictions and is directly related to the portfolio performance. Analyzing the decile return along with the decile RMSE and precision provides insights into the relation between portfolio performance and prediction accuracy.

Specifically, at each date in the testing data, we group the predictions with the true returns into deciles based on the ranking of forecasts (i.e., the highest predictions are in the top 9th decile and the lowest ones are in the bottom 0th decile). Then, with the true and predicted returns in each decile across dates, we calculate the decile RMSE, decile precision, and decile return. The decile precision is the percentage of the true returns whose decile based on the ranking of true values is equal to the current decile. It is related to the portfolio performance, because, for instance, a high precision of the top decile implies that a high proportion of stocks in this decile has a high true forward return, thereby benefiting the portfolio including stocks from the top decile.

For portfolio backtesting, we report the cumulative return charts and performance statistics like annualized returns and Sharpe ratios in the testing period.

## A.2 Additional Results of the North American Universe

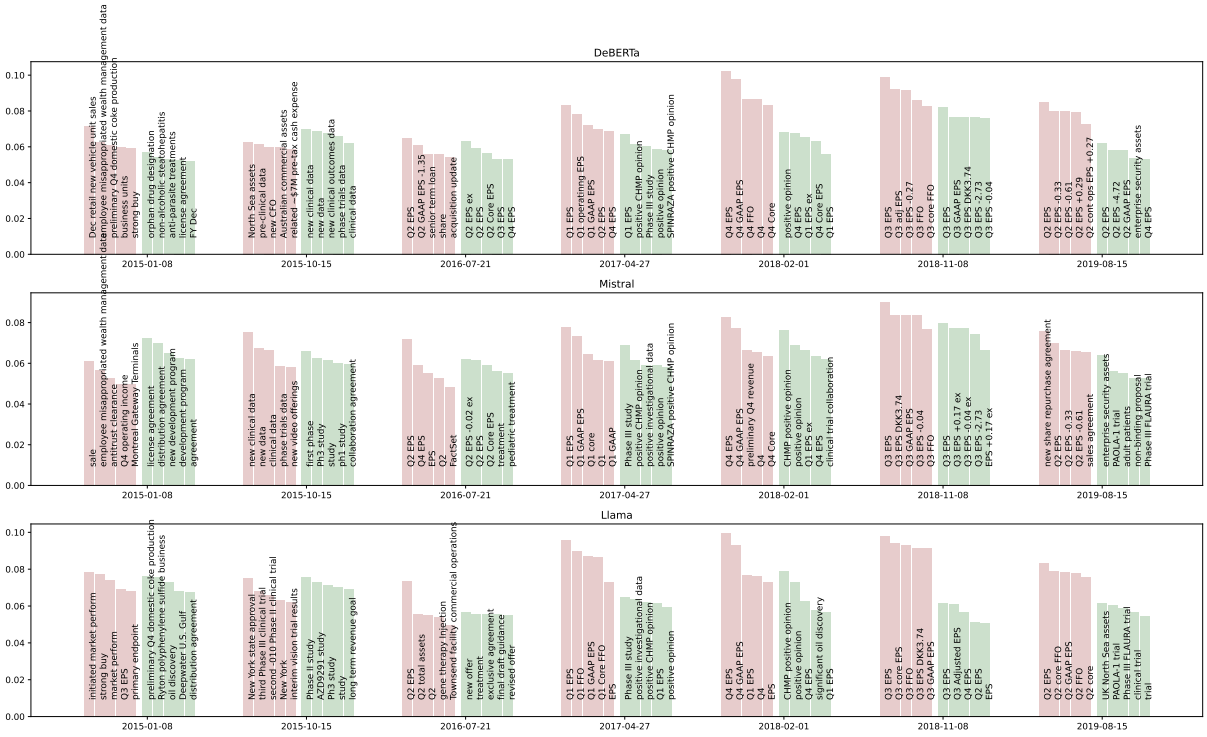


Figure 6: Qualitative Interpretation of the News Related to the Return Predictions of Bottom and Top Deciles for the North American Universe. The red and green bar charts correspond to the bottom (0th) and top (9th) deciles respectively. For each date chosen from the testing period, it shows the top 5 frequent phrases from the news leading to the prediction in the bottom/top decile. Phrases are ranked based on (Mihalcea and Tarau, 2004) as shown by the y-axis.

Fig. 6 provides an interpretative analysis of news driving the return predictions in the top and bottom deciles. It reports the prominent phrases across sampled portfolio rebalancing dates to capture key topics from the news.

A comparison between high and low return phrases (green vs. red bars) reveals that earnings-related events (e.g., EPS, Adjusted EPS) are commonly relevant for both. However, topics contributing to low return predictions are more varied, including issues such as clinical trials and antitrust matters.

Meanwhile, LLMs exhibit different focuses when generating predictions. For instance, on 2016-07-21, both DeBERTa and Mistral were more influenced by EPS-related news for high return predictions. In contrast, Llama's predictions on the same date were driven by other events such as guidance and revised offers. This highlights the different ways LLMs prioritize and process financial events when making predictions. The observation suggests potential avenues for future research on the underlying mechanism of the focus difference as well as aligning LLMs' focuses, aiming for more consistent and structured predictions across different LLMs.

### A.3 Results of the European Universe

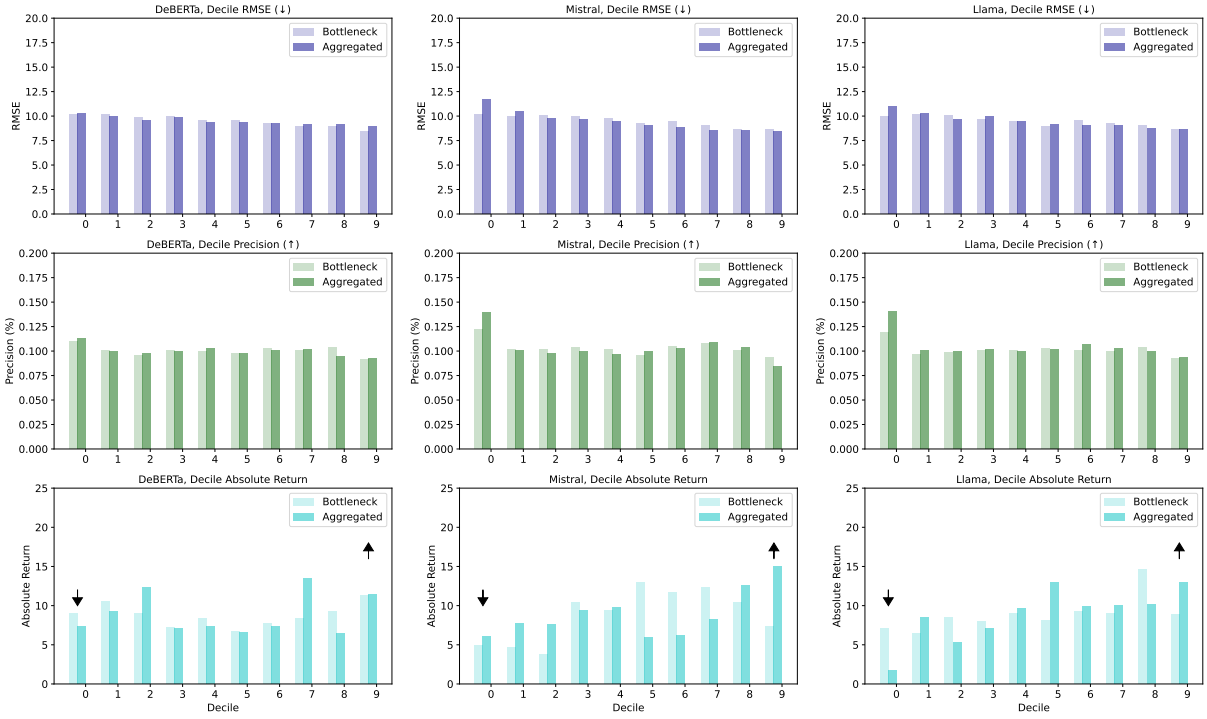


Figure 7: Decile Performance of Bottleneck and Aggregated Representations in the European Universe (best viewed in color). Top Row: Decile RMSE. Middle Row: Decile Precision. Bottom Row: Decile Return. The up (or down) arrow indicates the higher (or lower) values are desirable.

Table 3: Statistics of Portfolios in the European Universe. The Universe Equally-Weighted represents the universe performance reported under the Long-only Portfolio column.

| | Long-only Portfolio | | Long-short Portfolio | |
|---|---|---|---|---|
| | Ann. Return % (↑) | Sharpe Ratio (↑) | Ann. Return % (↑) | Sharpe Ratio (↑) |
| Universe Equally-Weighted | 9.75 | 0.74 | – | – |
| Sentiment_FinVader | 10.25 | 0.70 | 3.40 | 0.45 |
| Sentiment_FinBert | 8.17 | 0.57 | -0.36 | 0.00 |
| DeBERTa_Bottleneck | 11.04 | 0.81 | 2.11 | 0.31 |
| DeBERTa_Aggregated | 11.11 | 0.81 | 3.84 | 0.52 |
| Mistral_Bottleneck | 6.40 | 0.48 | 1.94 | 0.26 |
| Mistral_Aggregated | 15.12 | 1.02 | 9.07 | 1.04 |
| Llama_Bottleneck | 8.20 | 0.62 | 1.25 | 0.17 |
| Llama_Aggregated | 12.76 | 0.90 | 11.47 | 1.27 |

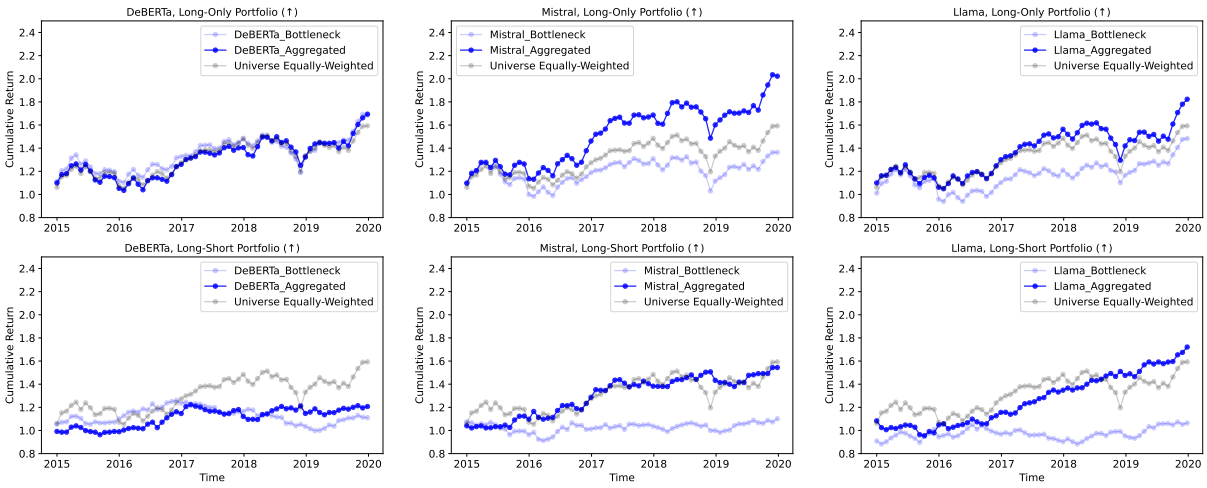


Figure 8: Cumulative Return Charts of the Portfolios based on Bottleneck and Aggregated Representation Models in the European Universe (best viewed in color). Top Row: Long-only Portfolios. Bottom Row: Long-short Portfolios.

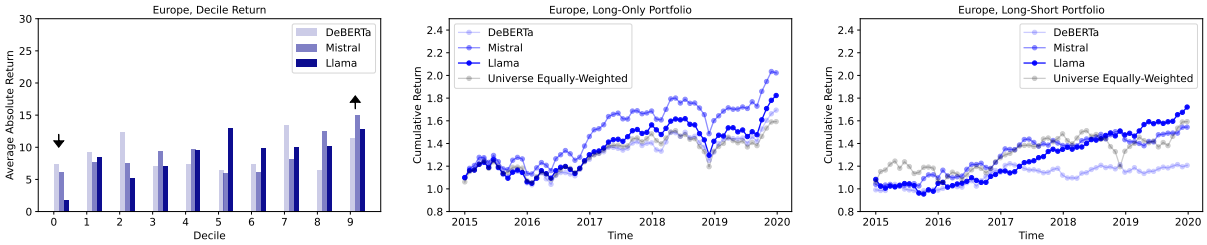


Figure 9: Comparison of Encoder-only and Decoder-only LLMs with the Suited Representations in the European Universe (best viewed in color).

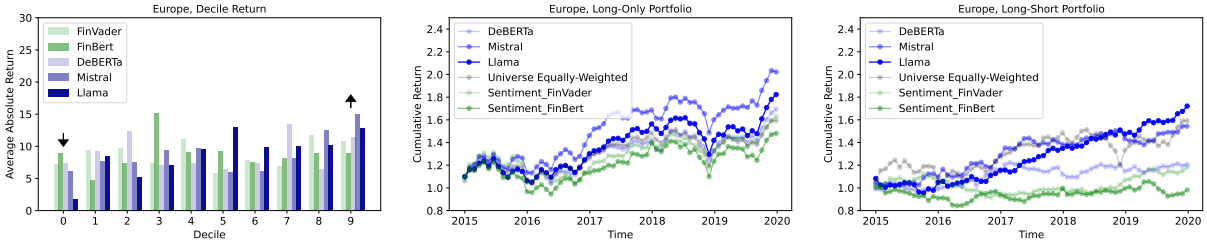


Figure 10: Comparison with Sentiment-based Portfolios in the European Universe (best viewed in color).

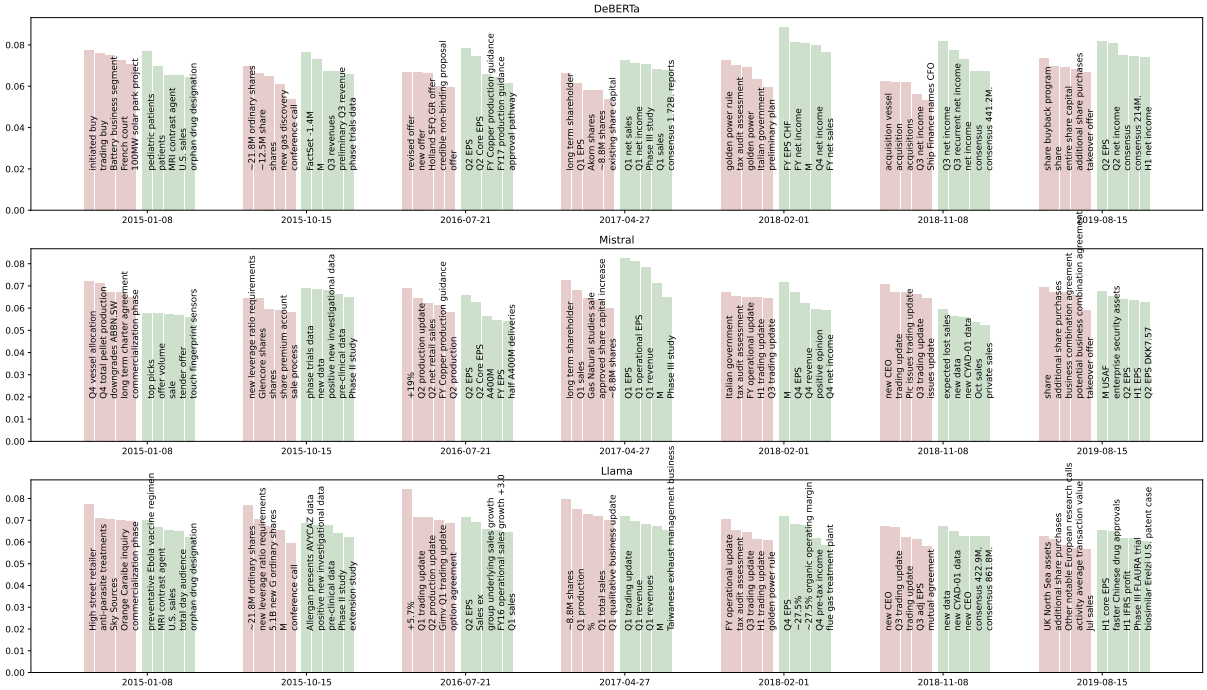


Figure 11: Qualitative Interpretation of the News Related to the Return Predictions of Bottom and Top Deciles for the European Universe. The red and green bar charts correspond to the bottom (0th) and top (9th) deciles respectively. For each date chosen from the testing period, it shows the top 5 frequent phrases from the news leading to the prediction in the bottom/top decile. Phrases are ranked based on (Mihalcea and Tarau, 2004) as shown by the y-axis.

## A.4 Results of the Emerging Markets Universe

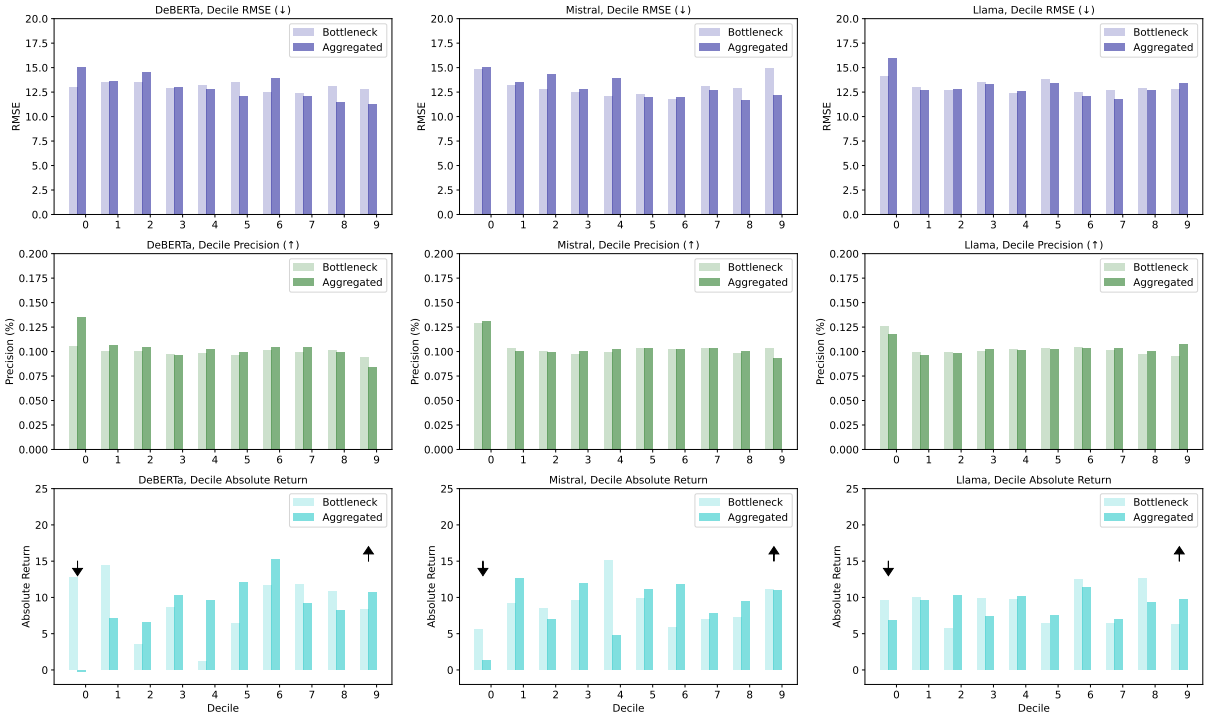


Figure 12: Decile Performance of Bottleneck and Aggregated Representations in the Emerging Markets Universe (best viewed in color). Top Row: Decile RMSE. Middle Row: Decile Precision. Bottom Row: Decile Return. The up (or down) arrow indicates the higher (or lower) values are desirable.

Table 4: Statistics of Portfolios in the Emerging Markets Universe. The Universe Equally-Weighted represents the universe performance reported under the Long-only Portfolio column.

| | Long-only Portfolio | | Long-short Portfolio | |
|---|---|---|---|---|
| | Ann. Return % (↑) | Sharpe Ratio (↑) | Ann. Return % (↑) | Sharpe Ratio (↑) |
| Universe Equally-Weighted | 3.91 | 0.32 | – | – |
| Sentiment_FinVader | 6.18 | 0.43 | -0.08 | 0.04 |
| Sentiment_FinBert | 9.76 | 0.70 | 1.69 | 0.21 |
| DeBERTa_Bottleneck | 7.32 | 0.50 | -5.00 | -0.36 |
| DeBERTa_Aggregated | 9.88 | 0.64 | 10.96 | 0.97 |
| Mistral_Bottleneck | 10.12 | 0.63 | 4.94 | 0.47 |
| Mistral_Aggregated | 10.11 | 0.64 | 9.16 | 0.68 |
| Llama_Bottleneck | 4.94 | 0.36 | -3.99 | -0.28 |
| Llama_Aggregated | 8.82 | 0.58 | 1.83 | 0.19 |

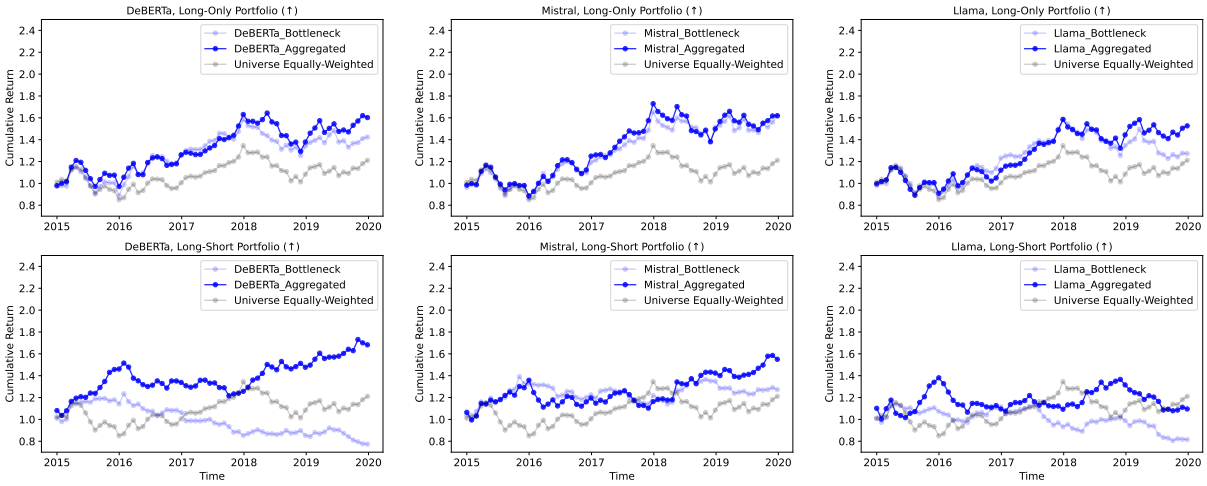


Figure 13: Cumulative Return Charts of the Portfolios based on Bottleneck and Aggregated Representation Models in the Emerging Markets Universe (best viewed in color). Top Row: Long-only Portfolios. Bottom Row: Long-short Portfolios.

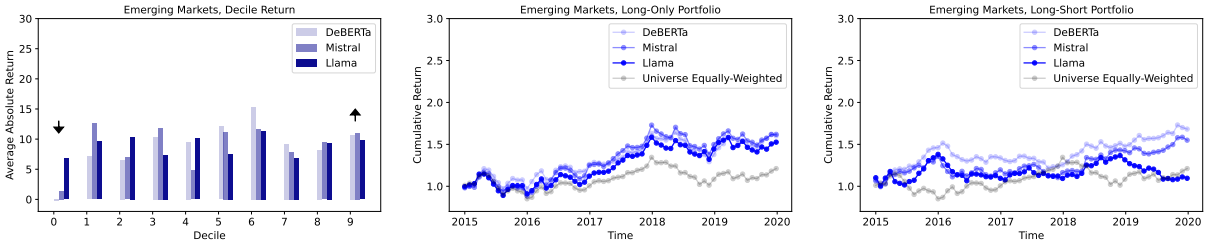


Figure 14: Comparison of Encoder-only and Decoder-only LLMs with the Suited Representations in the Emerging Markets Universe (best viewed in color).

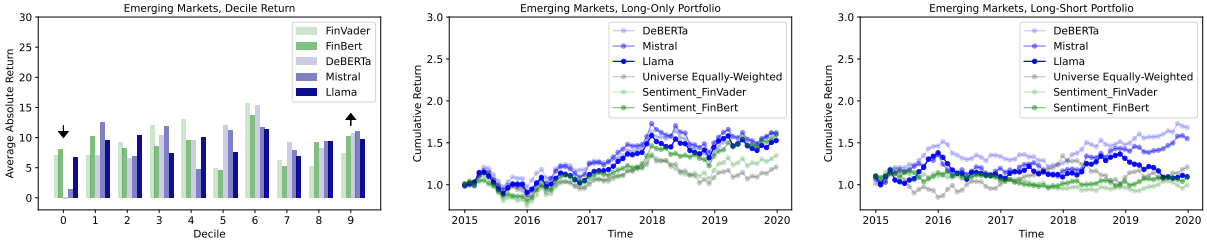


Figure 15: Comparison with Sentiment-based Portfolios in the Emerging Markets Universe (best viewed in color).

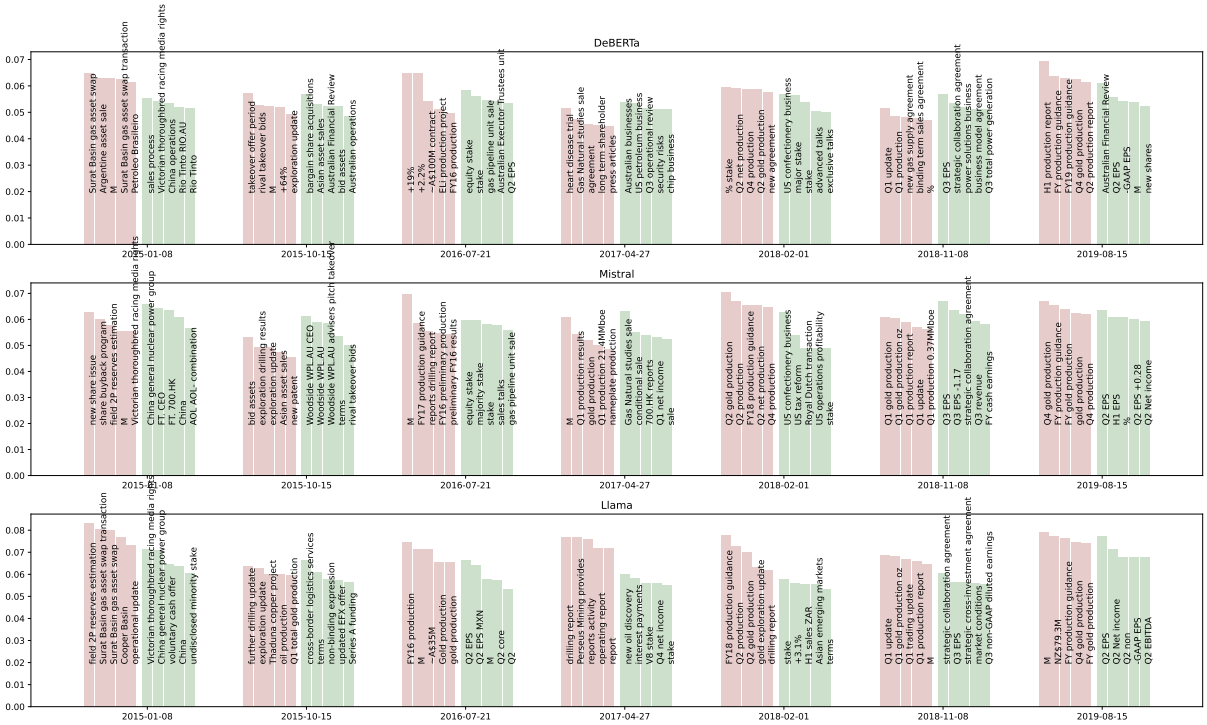


Figure 16: Qualitative Interpretation of the News Related to the Return Predictions of Bottom and Top Deciles for the Emerging Markets Universe. The red and green bar charts correspond to the bottom (0th) and top (9th) deciles respectively. For each date chosen from the testing period, it shows the top 5 frequent phrases from the news leading to the prediction in the bottom/top decile. Phrases are ranked based on (Mihalcea and Tarau, 2004) as shown by the y-axis.