Semantic Graph Based Learning for Trend Prediction from Long Financial Documents

BOLUN (NAMIR) XIA, Computer Science, Rensselaer Polytechnic Institute, Troy, United States
APARNA GUPTA, Lally School of Management, Rensselaer Polytechnic Institute, Troy, United States
MOHAMMED J ZAKI, Computer Science, Rensselaer Polytechnic Institute, Troy, United States

Language models (LMs) have shown great promise in generative and predictive tasks. However, they do not explicitly incorporate semantic relations, and despite the progress in increasing the context size, they still struggle with long documents. Since abstract meaning representation (AMR), which is a graph-based representation of text to preserve its semantic relations, can encode semantic relationships at a deeper level, it can be beneficially utilized by graph neural networks (GNNs) for constructing effective document-level graph representations built upon LM embeddings for predictive tasks. We propose FLAG, an AMR-based framework to generate document-level embeddings via GNNs for long document classification tasks. We construct document-level graphs from sentence-level AMR graphs, endow them with finance-specific LM word embeddings, apply a GNN-based deep learning mechanism, and examine the efficacy of our AMR-based approach in predicting trends from financial documents. Extensive experiments on several different tasks are conducted on two large datasets of quarterly earnings call transcripts. We find that FLAG outperforms fine-tuning LMs directly on text in predicting stock price movement trends, as well as previous work utilizing document graphs and GNNs for text classification. Finally, we demonstrate our AMR-graph-based approach's potential for explainability via a case study.

CCS Concepts: • **Information systems → Content analysis and feature selection**; **Data analytics**; • **Computing methodologies → Information extraction**; *Semantic networks*; **Supervised learning by classification**; **Batch learning**; **Neural networks**.

Additional Key Words and Phrases: Long financial documents, Predictive analytics, Graph neural networks, Abstract meaning representation, Language models, Document representation, Stock trend prediction, Earnings calls, Conference call transcripts

1 Introduction

Textual data is an important qualitative source of information in the financial domain. Financial reports can provide valuable signals for a firm's future performance, since these reports usually contain forward-looking plans and strategies, which may not be fully captured in their financial statements. Since textual data provides greater insights into firm performance, various methods have been utilized for transforming these textual reports into numerical representations in order to define effective features for predicting target variables, such as temporal price trend, that are of value to investors.

Despite the progress that has been made in recent years, especially in the sphere of LMs, there still remains the challenge of long documents whose lengths usually exceed the maximum context length of LMs. Even with longer context large language models (LLMs), learning good representations of documents is still quite difficult: a recent benchmark work on Q&A tasks in the financial domain demonstrated that even big LLMs

Authors' Contact Information: Bolun (Namir) Xia, Computer Science, Rensselaer Polytechnic Institute, Troy, New York, United States; e-mail: xiabolun@gmail.com; Aparna Gupta, Lally School of Management, Rensselaer Polytechnic Institute, Troy, New York, United States; e-mail: guptaa@rpi.edu; Mohammed J Zaki, Computer Science, Rensselaer Polytechnic Institute, Troy, New York, United States; e-mail: zaki@cs.rpi.edu.

ACM 2158-6578/2026/4-ART
https://doi.org/10.1145/3810185

such as GPT-4 [36] have difficulties in answering questions correctly based on specific corpora of financial documents [17]. In addition, with transformer-based methods, semantic relations between word-token entities are usually constructed arbitrarily, either with full attention where each word-token attends to every other word-token, or sparse attention, where attention between word-tokens is set up arbitrarily, such as sliding window attention or randomized attention.

We propose **F**inancial **L**ong Document Classification via **A**MR-based **G**NNs (FLAG), that learns effective document-level embeddings based on specialized LM word embeddings in the finance domain through AMR [3], which is a graph representation of text that preserves semantic relations. The unique feature of AMR graphs is that they are abstracted representations of text capable of capturing the semantic meaning of sentences, rather than just verbatim word sequences. Hence, words, phrases and sentences that have the same meaning, but differ in wording or spelling, usually result in the same AMR representation. As such, AMR is more semantically detailed and represents deeper meaningful relations between semantic concepts.

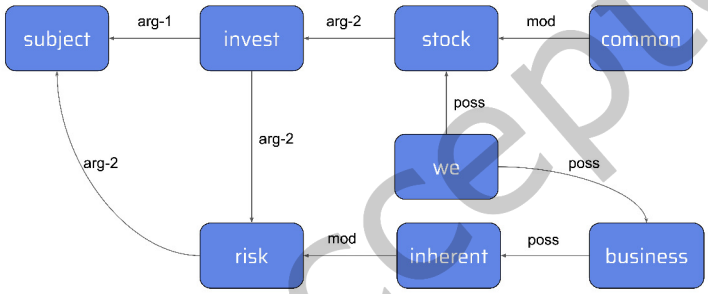

Fig. 1. An example of the AMR graph for the sentence: *an investment in our common stock is subject to risks inherent to our business.*

In order to demonstrate the abstracting nature of AMR graphs, in Fig. 1, we show the AMR graph of a sample sentence: *an investment in our common stock is subject to risks inherent to our business.* As we can see, the AMR graph is an abstracted representation of semantics. It can identify meaningful concepts within a sentence, such as "invest" and "subject", and it extracts the semantic relations between them. Moreover, we see that the two instances of "our" in the original sentence have both been abstracted as the concept of "we", and the node "we" has possessive relations with "stock" and "business."

Our approach utilizes the transition AMR parser [34] to transform each sentence of a document into an AMR graph with alignment of each node with its corresponding word in the sentence. The sentence-level graphs are then aggregated using a hierarchical approach that utilizes both document-level and sentence-level virtual nodes. We initialize each node (or word) with its contextual embedding using FinBERT [45], a specially trained LM in the finance domain. On the document-graph thus constructed, we apply GATv2 [7], a GNN that employs dynamic attention mechanism. Finally, we take the embedding of the document virtual node as the final representation for the document, and use it for downstream classification tasks. While we choose FinBERT as the base LM for FLAG, other models such as BloombergGPT [43], or open-source models such as FinGPT [44] can also be used.

In summary, our contributions are:

- We propose and implement an AMR-based deep learning framework for classification tasks geared towards long financial documents. Our FLAG approach constructs novel document-level AMR graphs from sentence-level AMR graphs and uses a GNN to learn effective document-level representations.
- We perform an extensive set of experiments on two collections of earnings call transcripts for companies from different sectors of the economy and from the S&P 1500 Composite Index to show that FLAG outperforms previous methods in predicting stock price movement trends for different time horizons, thereby achieving highly competitive results.

2 Related Works

In processing documents, traditional approaches for feature identification generate static embeddings that do not contain contextual information. Methods such as Term Frequency - Inverse Document Frequency (TF-IDF) [21], word2vec [32], and GloVe [37] belong in this category. They generate numerical vector representations that contain some semantic information, but strictly speaking, are not contextual embeddings. Recent approaches construct contextual embeddings that represent a word in view of its context. LMs, such as BERT [11] and GPT [36] belong in this category. These approaches can learn different representations for a word according to its surrounding context. The challenge with LMs, however, for using them on long financial documents, such as corporate earnings call transcripts, is the difficulty to extract document-level features, since the maximum number of word tokens these transformer LMs can handle is limited, and even if they can handle longer context windows, getting effective document-level representations still poses a big challenge. Nonetheless, we do evaluate our method against representative baselines in this category of contextual LMs, such as FinBERT [45] in the domain-specific fine-tuned models subcategory, and Longformer [5] in the long-context transformers category. Longformer employs a local windowed attention with a task motivated global attention to achieve linearly scaling attention with sequence length. We also compare against Long T5 [15], which employs a local/global attention mechanism named 'Transient Global,' and is trained for sequence-to-sequence tasks. We conclude that compared to FLAG, these long-context transformers employ arbitrary attention mechanisms that do not take into account the semantic relations between words, since their attention mechanisms are not explicitly guided by semantic structure.

On the semantic graph side, to our knowledge, AMR-based approaches have not been applied for long financial document classification tasks, such as earnings call transcripts that can exceed 7,000 words in length. However, there have been several methods in different domains that utilize AMR for textual analysis. For example, researchers have used it for text classification [39], event detection [27], profanity and toxic content detection [13], paraphrasability prediction and paraphrase generation [23], and machine translation [25]. All of these utilize only sentence-level AMR, which is unsuitable for our purpose.

Methods for AMR parsing, which is the process of transforming text into AMR graphs, are well-studied. Transition-based parsers, such as [47] and [24], provide top-tier sentence-level results, and AMR aligners, such as [12], provide reliable AMR-to-text alignments that link each node entity to its corresponding word in the original sentence. There have also been recent works on parsing multi-sentence AMRs to preserve cross-sentence information. O'Gorman et al. [35] provided a corpus of annotated multi-sentence AMRs, which was used by Naseem et al. [33] to implement a new approach to constructing multi-sentence or document-level AMR representations. Since their approach is still limited to short documents (e.g., averaging about 429 words per document), it is unsuitable for our purpose. Instead, we transform the AMR graphs into a document-level graph specially designed for long documents.

As for utilizing document graphs and GNNs to perform graph learning for text classification in the finance domain, Medya et al. [31] designed and implemented StockGNN, which constructs document graphs based on the contextual window of each unique word in the document, applies a Gated GNN [28] on the document graphs,

and concatenates the final embeddings of the document graphs with their respective doc2vec [22] embeddings to generate the final document representation for text classification. They collected a corpus of general domain earnings calls from several sectors of the economy and predicted the financial impact of earnings calls on stock price using StockGNN. We use the earnings call corpus they collected and StockGNN as a baseline method in our experiments. Importantly, StockGNN graphs are localized context graphs, and do not model semantics as we do via AMR. In addition, HiPool [26] employs a chunking approach to modeling long documents. It first creates overlapping chunks and extracts their representations (e.g., CLS token), and uses a hierarchical GNN model to aggregate the local sentence information. It further proposes an inter-hierarchy attention mechanism. Unlike our FLAG approach, HiPool does not take semantic relations within chunks into account, but since it deals with representations for long documents using GNNs, we use it as another baseline in our experiments.

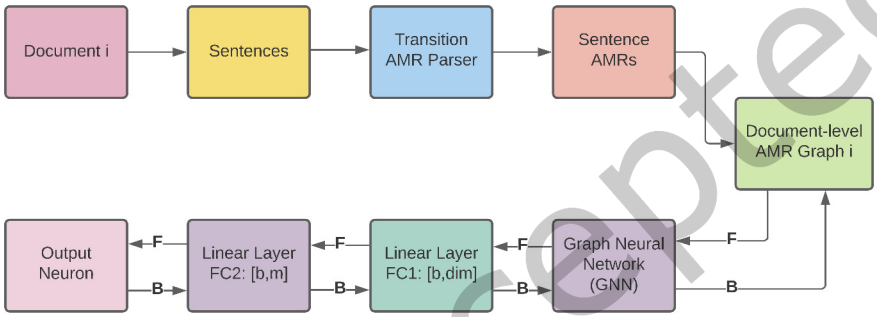

Fig. 2. FLAG Architecture: Each document is parsed into sentences, which are converted into sentence AMR graphs. Using our hierarchical approach, we combine them into the document-level graph, which is endowed with contextual LM word embeddings, and we then apply the GNN model to generate the final document virtual node embedding. This embedding is then passed through two fully connected linear layers to predict the target output. F: forward propagation; B: backward propagation.

3 The FLAG Approach

FLAG is an AMR-based GNN graph learning framework based on LM embeddings for long financial document classification. For each document, we parse all its sentences into AMR graphs, and construct a document-level AMR graph hierarchically with a sentence virtual node for every sentence and a document-level virtual node to represent the whole document. We initialize each node with the word embedding of its corresponding word, generated from a contextual LM. Next, we apply a GNN model to generate the final document representations by taking the document virtual node embeddings. As such, our approach can be split into three phases: (1) Sentence AMR Parsing, (2) Document-level Graph Construction, and (3) GNN Model Training and Fine-Tuning, with the architecture illustrated in Fig. 2. Each of the phases is discussed next.

3.1 Sentence AMR Parsing

In a corpus consisting of N documents, $L = \{d_1, d_2, \cdots, d_N\}$, let d_i represent the i-th document of the corpus. For a document d_i, we first sentencize the document into sentences, using the pre-trained Punkt tokenizer for English in the NLTK package [6]. This provides a sequence of sentences $S = \{s_1, s_2, \cdots, s_{m_i}\}$, where m_i denotes the total

number of sentences in d_i. Each sentence in S is parsed into a sentence-level AMR graph using the transition AMR parser [34], thereby generating a sequence of sentence graphs, $SG_i = \{sg_1, sg_2, \cdots, sg_{m_i}\}$.

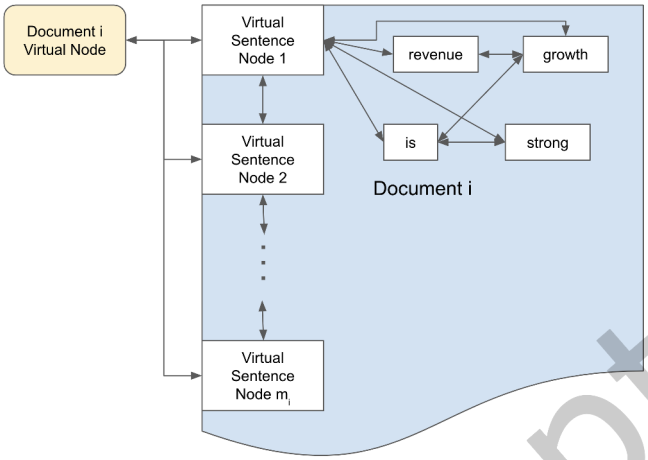

Fig. 3. Document-level graph construction.

3.2 Document-level Graph Construction

Fig. 3 shows our document-level graph construction. For d_i, we have a sequence of sentence AMR graphs $SG_i = \{sg_1, sg_2, \cdots, sg_{m_i}\}$. For each sentence graph in SG_i, we make a virtual sentence node that connects to all the nodes in the sentence graph, culminating in a consecutive sequence of virtual sentence nodes $SN_i = \{sn_1, sn_2, \cdots, sn_{m_i}\}$, and then we connect these sentence virtual nodes in a consecutive manner, such that sn_1 is connected to sn_2, sn_2 is connected to sn_3, and so on, both forming a hierarchical representation of each sentence and preserving the order of sentences in the document. In turn, each sentence node in SN_i is connected to a virtual document node dn_i, representing d_i. Overall, all the original nodes and edges in SG_i, all the virtual sentence nodes and their edges in SN_i, and the virtual document node dn_i and its edges, form the graph structure g_i for document d_i. We make every edge of the graph bidirectional, so that information can flow both ways during model training.

After constructing the graphs, we initialize the nodes with embeddings of their corresponding words in the original text, generated from the base LM method. We are able to do so, because the transition AMR parser [12] provides us with alignment between node entities in the sentence AMR graphs in SG_i and their corresponding words in the original text of the sentences.

The LM method that we chose is FinBERT [45], a specialized LM in the finance domain which generates token embeddings of size 768. We initialize the non-virtual nodes in our document-level graphs with the word embeddings of the original words in the sentence that they are aligned with. We pass each sentence of the document through the LM. A word's embedding is the average of the sum of the last 4 hidden state embeddings for each of the sub-tokens. Through this process, each non-virtual node in g_i is initialized with word embeddings of size 768. All virtual nodes, dn_i and all the nodes in SN_i, are initialized as zero vectors.

To demonstrate how sentence-level AMR graphs are aggregated into a document-level representation, a simple illustrative example is shown in Fig. 4. This document is clipped from the 2022 Q1 earnings call transcript of

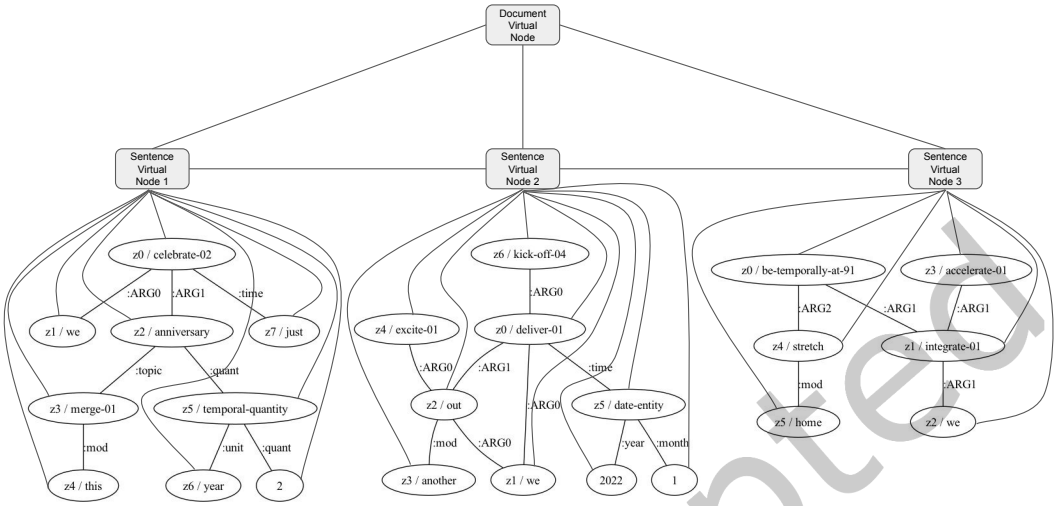

Fig. 4. An illustrative example of how sentence-level AMR graphs are aggregated into a document-level representation. The sentence-level AMR graphs (from left to right) represent the sentences: (1) We just celebrated the two-year anniversary of this merger. (2) We delivered another exciting outperformance in Q1 to kick off 2022. (3) We're in the home stretch of our accelerated integration.

T-Mobile US, and for simplicity, we only chose three sentences: i) "We just celebrated the two-year anniversary of this merger." ii) "We delivered another exciting outperformance in Q1 to kick off 2022." iii) "We're in the home stretch of our accelerated integration." As shown in the figure, each sentence-level AMR graph is anchored at the sentence virtual node, which connects to every node in its sentence graph. The sentence virtual nodes are then connected with each other to reflect the sequential order of sentences in the document, and all the sentence virtual nodes are finally connected to a document virtual node, which serves as the document-level representation. It is important to note that all the connections in the graph are modeled as bidirectional homogeneous edges without edge types, so the edge types parsed for the sentence-level AMR graphs (such as ARG0, ARG1, etc.) are not encoded into the document graph.

3.3 GNN Model Training and Fine-Tuning

Given document-level graphs constructed for every document in the corpus L, forming a collection of graphs, $G_L = \{g_1, g_2, \cdots, g_N\}$, we apply an initial MLP layer on the graph features to transform the original node embedding dimension of 768 into another hidden dimension. We then train a GNN model on the transformed graphs, specifically, GATv2 [7], an attention-based GNN model which has been theoretically proven to achieve dynamic attention. Afterwards, we take the embedding vector of the virtual document node dn_i as the document representation for the graph g_i, denoted as $\mathbf{h_i}$. We then use a linear layer:

$$\mathbf{x_i} = W_1\mathbf{h_i} + \mathbf{b_1}, \tag{1}$$

followed by another linear layer that outputs the final predicted one-hot label, $\hat{\mathbf{y}}_\mathbf{i}$, after a softmax.

$$\hat{\mathbf{y}}_\mathbf{i} = softmax(W_2\mathbf{x_i} + \mathbf{b_2}) \tag{2}$$

Finally, we use the cross entropy loss, $\mathcal{L} = -\sum_k y_k \log(\hat{y_k})$, where $\hat{y_k}$ is the k-th element of the predicted label, and y_k is the k-th element of the target one-hot label.

4 Empirical Evaluation

We now present experimental results to evaluate the efficacy of FLAG. We utilized the transition AMR parser [34] with the AMR 2.0 structured BART large model pre-trained checkpoint [12] to perform AMR parsing and alignment, and we used the Deep Graph Library (DGL) [42] for graph construction and initialization, as well as for GNN model training. Our code and datasets are publicly available on GitHub via https://github.com/Namir0806/FLAG.

4.1 Dataset and Metrics

We take corporate earnings call transcript data as the subject of our analysis. An earnings call is a conference call between company executives and the financial community. It is usually held on a quarterly basis following the release of a company's earnings report. On this call, the management reviews the company's performance for a specific period, as well as potential risks and future plans, which can cause subsequent stock prices to shift dynamically [9].

To evaluate FLAG, we compare its performance with baseline methods on both the Medya et al. earnings call dataset [31], which contains earnings calls from 5 sectors of the economy during the period from 2010 to 2019, and a new dataset of S&P 1500 earnings calls that we collected, which contains earnings calls of companies from the S&P 1500 Composite Index [16] during the period from 2010 to 2023.

Table 1. Medya earnings call dataset: N is number of documents and L is average document length.

	Train/Val N	Test N	Train/Val L	Test L
Finance	14930	2036	7351.20	6668.94
Health	9869	1646	7083.55	6591.70
Materials	9240	1256	7165.85	6452.11
Service	14890	1983	7498.45	6928.90
Technology	14319	2069	7112.78	6785.09

Table 2. FLAG graph statistics for the Medya earnings call dataset: N is number of nodes, E is number of edges, and D is degree.

	Avg. N	Avg. E	Avg. D
Finance (Train/Val)	6307.34	19559.83	3.10
(Test)	5784.46	17921.15	3.10
Health (Train/Val)	6161.94	19115.60	3.10
(Test)	5775.57	17908.75	3.10
Materials (Train/Val)	6261.34	19400.57	3.10
(Test)	5640.78	17464.83	3.10
Service (Train/Val)	6464.74	20066.69	3.10
(Test)	5993.24	18598.27	3.10
Tech (Train/Val)	6192.78	19215.09	3.10
(Test)	5934.40	18407.39	3.10

Medya Earnings Call Dataset: The Medya et al. [31] earnings call dataset consists of earnings calls during the 2010 to 2019 period. It is split into five sectors: finance, health, basic materials, service, and technology. They used this dataset to predict a binary trend, which they call value-based labels. They define the label function $y_v(T_d^c)$ for an earnings call transcript T_d^c of a company c on the day d as follows:

$$y_v(T_d^c) = \begin{cases} 1, & \text{if } S_{d+1}^c > S_{d-1}^c \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where S_{d+1}^c and S_{d-1}^c denote the closing stock prices of company c on the following and preceding business day respective to day d.

Since this task is to analyze the trend, it aims to capture the immediate financial impact of an earnings call, so the target is daily and more granular than most financial impact analyses in industry. Moreover, in the real world, financial impact analyses in the context of portfolio management or asset pricing involve multiple complex factors, both qualitative and quantitative, and the textual signals from earnings calls only form a fraction of all the factors that can affect stock price trend. Therefore, the purpose of experimenting on this dataset in predicting the daily value-based label is to evaluate the effectiveness of document representations produced by different methods for predicting immediate price movements only from textual data.

All the earnings calls during the period of 2010 to 2018 are used as the training/validation set (with an 80:20 split), and all the earnings calls during 2019 are used as the test set, as is done by Medya et al. [31]. The detailed data statistics are listed in Table 1. As for the document-level graphs constructed using the FLAG approach from earnings calls for the Medya dataset, the detailed statistics are shown in Table 2.

Table 3. S&P 1500 earnings call dataset statistics, with the total number of documents (N) and average document length (L).

	N	L
Train/Val	55696	8138.38
Test	6049	7912.46

Table 4. FLAG graph statistics for the S&P 1500 earnings call dataset.

	Avg. # of nodes	Avg. # of edges	Avg. degree
Train/Val	7093.21	28883.19	4.07
Test	6512.83	26353.94	4.05

S&P 1500 Earnings Call Dataset: The S&P 1500 earnings call dataset we collected contains more recent data, and consists of earnings call transcripts of companies from the S&P 1500 Composite Index [16] for the period from 2010 to 2023. It includes companies from all sectors and represents the overall U.S. equity market. Unlike for the Medya dataset, we use this dataset to predict weekly value-based labels. For this dataset, we define the label function $y_v(T_d^c)$ for an earnings call transcript T_d^c of a company c on day d as follows:

$$y_v(T_d^c) = \begin{cases} 1, & \text{if } \frac{1}{5}\sum_{i=1}^{5} S_{d+i}^c > \frac{1}{5}\sum_{i=1}^{5} S_{d-i}^c \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $\frac{1}{5}\sum_{i=1}^{5} S_{d+i}^c$ and $\frac{1}{5}\sum_{i=1}^{5} S_{d-i}^c$ denote the average value of closing stock prices of company c from the following and preceding business week respective to day d (a business week is defined as 5 working days).

The target labels in this case are on a weekly basis, so it is closer to what an analyst would infer from a company's earnings calls about the future direction of the company in the following week. The purpose is to investigate the effectiveness of documents representations produced by different methods, but in the context of predicting a longer-term (weekly) target label.

All the earnings calls during the period from 2010 to 2021 are used as the training/validation set (with a 90:10 split), and all the earnings calls during 2022 and 2023 are used as the test set. The detailed statistics for the dataset are shown in Table 3. The document-level graphs are constructed using the FLAG approach from earnings calls in the S&P 1500 corpus, with the detailed statistics shown in Table 4.

Table 5. Market-risk adjusted returns extension of the S&P 1500 earnings call dataset statistics, with the total number of documents (N) and average document length (L).

	N	L
Train/Val	44904	8161.06
Test	5349	7907.69

Market-Risk Adjusted Returns: In addition to the weekly value-based price change labels mentioned before for the S&P 1500 dataset, we add another, more realistic, return-based price trend label that adjusts for market risk, according to the Capital Asset Pricing Model (CAPM) [38] which states that:

$$E(R_a) = R_f + \beta(E(R_m) - R_f) \tag{5}$$

where $E(R_a)$ is the expected return on the asset, R_f is the risk-free rate, $E(R_m)$ is the expected return on the market, and β is a coefficient representing the extent to which the particular asset is moved by market risk (potential movements in the stock market as a whole). Assuming that CAPM holds true, any difference between $E(R_a)$, the expected return of the asset, and R_a, the actual return of the asset, represents the idiosyncratic price trend of the particular stock, which we use to formulate the label function:

$$y(T_d^c) = \begin{cases} 1, & \text{if } R_c(d-1, d+i) > E(R_c(d-1, d+i)) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $R_c(d-1, d+i)$ represents the actual return of company c over the period from the previous day, $d-1$, to the i-th day following the earnings call, $d+i$, and $E(R_c(d-1, d+i))$ represents the expected return of company c over the same period. Note that return is defined as:

$$R_c(d-1, d+i) = \frac{S_{d+i}^c - S_{d-1}^c}{S_{d-1}^c} \tag{7}$$

where S_d^c represents the adjusted closing stock price for company c at the end of day d. In our experiments, we set i to be both 1 and 5, representing a daily label and a weekly label.

In formulating the market-risk adjusted returns labels, we use the S&P 1500 Composite Index [16] as a whole to represent the market, and we take the daily risk-free rate from the Fama-French data library [14], compounding it for the weekly label. In addition, we need to estimate β, which is the coefficient denoting how much a stock is influenced by the market. For this, we source the **daily** returns from both the market and the specific stock with a rolling 2-year window ending one week before the earnings call, to avoid any potential overlap with signals from the earnings call event. For example, if an earnings call was on 11/13/2022, we define its 2-year rolling window to

be from 11/06/2020 to 11/06/2022. Then, for this rolling window, we calculate β using the covariance/variance formula:

$$\beta = \frac{\text{Covariance}(R_a, R_m)}{\text{Variance}(R_m)} \tag{8}$$

where R_a is the return on the particular asset, and R_m is the return on the market.

This type of target labels was suggested to us by the financial industry, and represents a more realistic approximation of the actual impact of an earnings call event on a particular company's stock price trend, after excluding the portion of returns that we estimate to be affected by overall market risk.

Since these labels require more historical data, the total number of data points is smaller than that of value-based weekly labels, but we still use the earnings calls during the period from 2010 to 2021 as the training/validation set (with a 90:10 split), and the earnings calls during 2022 and 2023 as the test set. The detailed statistics for this extension of the S&P 1500 dataset are shown in Table 5.

4.2 Methods

We describe the baseline methods and FLAG variants used in our experiments.

- **FinBERT [45]**: We compare with a baseline LM model, using a domain-specific model pre-trained on financial corpora, FinBERT [45]. This baseline method truncates the long document to the maximum length that the LM can take, and applies the LM to generate document embeddings, which is then passed through two linear layers to predict the target. Overall, this serves as a representative transformer-based LM baseline.
- **Longformer [5]:** We compare with a long-context transformer LM model, using a local windowed attention with a task motivated global attention mechanism, Longformer [5]. This baseline method truncates the long document to the maximum length that the LM can take, and applies the LM to generate document embeddings, which is then passed through two linear layers to predict the target. Overall, this serves as a representative long-context transformer-based LM baseline.
- **Long T5 [15]:** We compare with another long-context transformer LM model, using a local/global attention mechanism but geared towards sequence-to-sequence tasks, Long T5 [15]. This baseline method truncates the long document to the maximum length that the LM can take, and applies the LM to generate document embeddings, which is then passed through two linear layers to predict the target (since it is a sequence-to-sequence transformer, the target is converted into binary text labels). Overall, this serves as a representative long-context sequence-to-sequence transformer-based LM baseline.
- **Instruction Fine-tuning: LLM Q&A**: We model the prediction task as a LLM Q&A task, and using different prompts coupled with contexts containing earnings call texts, we fine-tune the Llama3 8B model [1] with the low-rank adaptation (LORA) technique implemented by FinGPT [29], in order to train the model to respond with a single word to indicate a positive or negative signal. Details of the prompts are given below.
- **StockGNN [31]**: StockGNN, the method proposed by Medya et al. [31], also uses contextual graphs and the Gated GNN (GGNN) method [28] to learn from long financial documents. However, it uses only a local contextual graph and does not leverage deeper semantics. Overall, this serves as a representative previous leading baseline method for stock price trend prediction tasks.
- **HiPool [26]:** HiPool is a hierarchical method which chunks a long document and uses GNNs to model the chunks into a single representation of the document. We directly apply their method to our datasets and tasks. Overall, this baseline method serves as a representative for hierarchical-chunking graph-based methods.

- **FLAG**: For the GNN model in FLAG, we configure GATv2 [7] with the optimal setting using the validation data, in terms of the number of layers, the number of attention heads per layer, and the hidden dimension that node embeddings of the original graphs are transformed to before applying the GNN model.
- **FLAG with LLM2Vec [4]**: LLM2Vec is a recent method which transforms decoder LLMs into text encoder models that can generate good contextual embeddings. In this variant of FLAG, instead of using FinBERT for node embeddings, we use two different LLM2Vec models built from Llama3 8B [1] and Mistral 7B [18].
- **GGNN on FLAG Graphs**: Since StockGNN uses only contextual graphs, we examine the effectiveness of GGNN [28] on the AMR graphs constructed in FLAG. This baseline serves to showcase the added benefits from a more semantics-based AMR graph with the same GNN as used in StockGNN.

Table 6. Best hyperparameters for all methods on the Medya earnings call dataset: (e) number of epochs, and (lr) learning rate.

	Finance	Health	Materials	Service	Tech
FinBERT (E)	7	11	7	9	3
FinBERT (LR)	10^{-5}	10^{-6}	10^{-5}	10^{-6}	10^{-5}
StockGNN (E)	717	725	316	178	187
StockGNN (LR)	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
FLAG (E)	14	9	17	10	15
FLAG (LR)	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}

Hyperparameter Tuning: For FLAG and its variants, we train the framework for up to 20 epochs; for LM baselines, we train up to 30 epochs; and for StockGNN and related methods, we train up to 1000 epochs. We vary the learning rate from 10^{-2} to 10^{-6}, and select the model with the lowest validation error as the best model. As an example, the best hyperparameters for each method on the Medya earnings call dataset are reported in Table 6.

4.3 Comparative Results

For experiments, we train the models to minimize the training loss, and because the target value is a trend metric, we evaluate the quality of the binary trend label predictions by evaluating the accuracy, precision (macro average), and recall (macro average) of the predictions, as is done in Medya et al. [31], as well as the F1 score.

Results on the Medya Dataset: Table 7 shows the detailed results of comprehensive experiments across all 5 sectors in the Medya earnings calls dataset. As we can see, FLAG outperforms FinBERT, Long T5, HiPool, and StockGNN across all sectors of the economy, and it outperforms Longformer by a significant margin in all sectors except in the materials sector, where it performs worse than Longformer, and in the service sector, where it is only doing slightly worse. Overall, the highest absolute performance is in the technology sector.

With regards to FinBERT, FLAG is able to achieve significant performance gains in accuracy over directly applying the domain-specific LM. Especially in the service and technology sectors, we see very high performance gains with FLAG (e.g., 26.2% gain for service), and in the financial sector, it outperforms FinBERT, however by a lesser margin (7.8% gain). We find the same trend for performance using StockGNN. The improvement in the service and technology sector is generally higher than in the other sectors. When it comes to comparing with Longformer, we see this trend of varying performance in different sectors as well. Generally, we do not see Longformer perform better than FLAG except in the materials and service sectors. In HiPool, we also find the same trend where HiPool performs much better in the service and technology sectors than in other sectors. We posit that this trend is due to the differences in the nature of earnings calls in different sectors of the economy and how the stock market reacts to an earnings call event in each particular sector, and we analyze this phenomenon in the case study below.

Table 7. Experiment results on the Medya earnings call dataset. Best results are marked in bold.

	Accuracy				
	Finance	Health	Materials	Service	Tech
FinBERT	0.574	0.547	0.516	0.496	0.541
Longformer	0.590	0.559	**0.616**	**0.629**	0.600
Long T5	0.561	0.484	0.469	0.508	0.493
StockGNN	0.559	0.584	0.561	0.554	0.551
HiPool	0.558	0.557	0.529	0.592	0.606
FLAG	**0.619**	**0.614**	0.597	0.626	**0.637**
	Precision				
	Finance	Health	Materials	Service	Tech
FinBERT	0.581	0.544	0.524	0.503	0.545
Longformer	0.578	0.555	**0.624**	0.629	0.639
Long T5	0.281	0.534	0.234	0.254	0.508
StockGNN	0.563	0.583	0.562	0.554	0.557
HiPool	0.536	0.554	0.538	0.592	0.610
FLAG	**0.615**	**0.616**	0.595	**0.629**	**0.641**
	Recall				
	Finance	Health	Materials	Service	Tech
FinBERT	0.521	0.543	0.523	0.501	0.543
Longformer	0.570	0.554	**0.621**	**0.629**	0.604
Long T5	0.500	0.510	0.500	0.500	0.500
StockGNN	0.564	0.583	0.562	0.553	0.553
HiPool	0.528	0.543	0.536	0.590	0.604
FLAG	**0.616**	**0.604**	0.593	0.627	**0.638**
	F1 Score				
	Finance	Health	Materials	Service	Tech
FinBERT	0.437	0.542	0.513	0.423	0.536
Longformer	0.566	0.553	**0.616**	**0.629**	0.574
Long T5	0.360	0.389	0.319	0.337	0.339
StockGNN	0.559	0.583	0.561	0.552	0.543
HiPool	0.511	0.526	0.525	0.589	0.600
FLAG	**0.615**	**0.598**	0.592	0.624	**0.635**

Earnings calls have an immediate financial impact on stock prices, and through this set of experiments, we show that generally FLAG is able to capture that better. In other words, as a more semantically meaningful approach, it is able to achieve better performance in predicting stock price movement trend for which the input document is a main contributing factor. This indicates that choosing the sparse connections in a semantically meaningful way, as done via AMR graphs, helps the model achieve better performance in trend analysis. We validate this conclusion through an ablation study of applying Gated GNN, the GNN used in StockGNN, on the AMR document-level graphs (discussed below). This finding is important for building a textual element within asset pricing model in the real world, where the textual signals' implications for stock price trend must be understood.

Time complexity of training. Beyond looking at the performance of models, it is important also to examine the time complexity of training FLAG compared to the different baseline methods. We took the two competitive baseline methods which performed above 60% in one of the sectors, namely Longformer and HiPool, and compared

Table 8. Comparison of time complexity between FLAG and competitive baseline methods.

Method	Avg. Time Per Epoch (sec)
Longformer	8009.9
HiPool	6934.3
FLAG	**854.9**

their time complexity of training with that of FLAG. Table 8 shows the average per epoch time over the first three epochs of each method, training on the financial sector of the Medya dataset. The average time per epoch is many times higher for Longformer and HiPool than for FLAG, with FLAG 9.4 to 8 times faster, respectively. This is due to FLAG's modeling of the semantic relations between words using AMR graphs. As such, we conclude that FLAG delivers superior performance compared to the competitive baselines, and does so at a fraction of the training time, which is a significant benefit in real-world usage, where models would often need to be updated as new data comes in.

Table 9. Experiment results on the S&P 1500 earnings call dataset. Best results are marked in bold.

	Accuracy	Precision	Recall	F1 Score
FinBERT	0.549	0.543	0.530	0.501
Longformer	0.588	0.593	0.573	0.557
Long T5	0.534	0.267	0.500	0.348
StockGNN	0.556	0.551	0.543	0.533
HiPool	0.552	0.557	0.527	0.468
FLAG	**0.601**	**0.598**	**0.597**	**0.597**

Results on the S&P 1500 Dataset: Table 9 shows the experimental results on the S&P 1500 dataset, in the context of predicting weekly average price trend. Once again, FLAG performs better in predicting longer-term price trend metrics at the weekly granularity when compared with FinBERT, Longformer, Long T5, HiPool, and StockGNN, with the graph-based approach of StockGNN performing better than FinBERT, Long T5, and HiPool. We also see Longformer performing the best among all the baseline methods, but not reaching the performance of FLAG in all four metrics. This dataset consists of earnings calls from all sectors in the U.S. equity market, as represented in the index. As such, the absolute performance gains of FLAG on this corpus are lower compared to the sector-specific Medya earnings call dataset, albeit the target metric scope is different: the latter is on a daily granularity. Overall, the results show that not only are there indications for weekly stock price trends contained in the soft information of the earnings calls, but that FLAG is able to extract this better with its semantically meaningful graph representations of documents, even across various different sectors of the economy.

Results on Market-Risk Adjusted Returns: Table 10 shows the experiment results on predicting the trend of market-risk adjusted returns, whether it will be positive or negative, in the context of both daily and weekly targets. As we can see, FLAG outperforms all baselines methods in the experiments, with the performance of StockGNN slightly increasing and that of FinBERT, decreasing, especially on the longer-term weekly target. Performance of HiPool and Long T5 decreased in daily targets compared to weekly targets. Longformer's performance increased slightly, rising to just a bit better than FLAG in accuracy on the weekly target, but worse in precision, recall, and most importantly, worse on the balanced metric of the F1 score. It is of note that generally transformer-based methods that take the first truncated chunk of long documents, such as FinBERT, Longformer, and Long T5, perform worse in the longer-term weekly targets compared to daily targets, and the opposite is generally true

Table 10. Experiment results on the market-risk adjusted returns extension of the S&P 1500 earnings call dataset. Best results are marked in bold.

	Daily Target			
	Accuracy	Precision	Recall	F1 Score
FinBERT	0.521	0.528	0.527	0.519
Longformer	0.586	0.585	0.585	0.585
Long T5	0.529	0.265	0.500	0.346
StockGNN	0.565	0.564	0.564	0.564
HiPool	0.524	0.489	0.498	0.388
FLAG	**0.592**	**0.612**	**0.602**	**0.586**
	Weekly Target			
	Accuracy	Precision	Recall	F1 Score
FinBERT	0.539	0.270	0.500	0.350
Longformer	**0.591**	0.587	0.581	0.578
Long T5	0.539	0.270	0.500	0.350
StockGNN	0.561	0.567	0.566	0.561
HiPool	0.555	0.547	0.535	0.513
FLAG	0.589	**0.599**	**0.597**	**0.588**

for hierarchical, graph-based approaches, such as StockGNN, HiPool, and FLAG, which can all take in the full document in their scope due to their graph modeling of the input text. This indicates both that the textual signals are more predictive of the more real-world market-risk adjusted returns target, and that FLAG has the potential to be applied in real-world use case scenarios in the financial industry, especially in asset recommendation for hedge funds and securities trading.

While real-world predictive analysis of stock price trends is composed of various factors such as historical price data, volatility, and financial news events, through this experiment, we are able to demonstrate the value our approach can contribute to existing predictive frameworks, in supplementing additional textual signals from earnings call events or other types of text-based events, compared with purely LM-based methods and graph learning methods that do not incorporate semantics.

4.4 Ablation Studies

Table 11. Ablation results of using LLM Q&A instruction fine-tuning on the market-risk adjusted returns extension of the S&P 1500 earnings call dataset. Best results are marked in bold.

	Daily Target			
	Accuracy	Precision	Recall	F1 Score
Llama3 8B	0.572	0.569	0.567	0.565
FLAG	**0.592**	**0.612**	**0.602**	**0.586**
	Weekly Target			
	Accuracy	Precision	Recall	F1 Score
Llama3 8B	0.544	0.540	0.539	0.538
FLAG	**0.589**	**0.599**	**0.597**	**0.588**

Comparison with instruction fine-tuning LLM for Q&A:. In this ablation, we take the market-risk adjusted returns prediction task, and model it as an instruction fine-tuning task for LLM Q&A. Specifically, we used the

LORA technique to finetune the Llama3 8B model, as is done in FinGPT [29]. For both the daily and weekly targets, we constructed 3 sets of different prompts to denote the same instruction (i.e., differently worded prompts). These were as follows (for the weekly prompts we replace **day** with **week**):

- "You are a financial expert. Based on a company's earnings conference call transcript below, please provide a single word response on whether the stock returns of this company in the next **day**, relative to market returns in the next **day**, is expected to be 'Positive' or 'Negative': "
- "You are a financial analyst. Based on a firm's earnings call transcript below, please answer with a single word response on whether the stock price returns of this firm in the next **day**, relative to market returns in the next **day**, is expected to perform 'Better' or 'Worse': "
- "You are an expert in finance and trading. Based on the information released from a company below, please tell us in a single word on whether the returns of this company in the next **day**, relative to market returns in the next **day**, is expected to 'Outperform' or 'Underperform': "

We performed the training with the context window capped and input texts truncated at 2048 tokens, due to memory limits, and we trained for 3 epochs, evaluating at every 10,000 steps with an 80:20 training-validation split, using the best model in terms of validation loss at the end to judge the performance of the model.

Table 11 shows the results of this ablation. As demonstrated, for the more immediate daily target, LLM fine-tuning was able to show relatively good performance, though still worse than FLAG, and for the longer-term weekly target, it loses its edge in performance. Interestingly, Llama3 8B performs better than StockGNN for the daily target but worse for the weekly target (see Table 10). Overall, it is clear that FLAG, with the semantic structure built in, which both adds in structural knowledge in training and enables the full document to fit into the training process without truncation, still outperforms LLM Q&A instruction fine-tuning for predicting stock price return trends based on textual signals, even though we continuously trained the LLM for about a week's time for each target.

Table 12. Ablation results of using LLM2Vec [4] encoder embeddings of Llama3 8B and Mistral 7B in FLAG, experimented on the Finance and Technology sectors of the Medya dataset.

	Finance			
	Accuracy	Precision	Recall	F1 Score
FLAG (Llama3 8B)	**0.627**	**0.621**	0.604	0.600
FLAG (Mistral 7B)	0.605	0.602	0.572	0.555
FLAG (FinBERT)	0.619	0.615	**0.616**	**0.615**
	Tech			
	Accuracy	Precision	Recall	F1 Score
FLAG (Llama3 8B)	0.623	0.628	0.624	0.621
FLAG (Mistral 7B)	0.618	0.627	0.620	0.613
FLAG (FinBERT)	**0.637**	**0.641**	**0.638**	**0.635**

Comparison with using LLM2Vec [4] encoder embeddings in FLAG:. Although decoder LLMs such as Llama3 [1] and Mistral [18] are not trained to be text encoders that can generate quality textual embeddings, LLM2Vec [4] is a novel method to convert and fine-tune decoder LLMs into text encoders. We used their most fine-tuned Bi + MNTP + Supervised models to generate embeddings for the nodes in FLAG document graphs, instead of using FinBERT [45], and we experimented on the finance and technology sectors of the Medya dataset. Table 12 shows the results of this ablation. As we can see, even though the general-domain LLM2Vec embeddings were somewhat effective, for example, in accuracy and precision on data points in the finance sector, they did not

perform as well as FLAG with FinBERT embeddings as indicated by the more balanced F1 Score. Overall, they do not contribute significantly to better performance and generally do worse than FLAG graph endowed with a domain-specific fine-tuned text encoder such as FinBERT.

Table 13. Performance of StockGNN vs. Gated GNN with FLAG.

	Finance			
	Accuracy	Precision	Recall	F1 Score
StockGNN	0.541	0.513	0.509	0.489
GGNN on FLAG graphs	**0.547**	**0.540**	**0.540**	**0.540**
	Tech			
	Accuracy	Precision	Recall	F1 Score
StockGNN	0.560	0.560	0.560	0.560
GGNN on FLAG graphs	**0.573**	**0.574**	**0.574**	**0.573**

Efficacy of AMR document-level graphs –comparison between StockGNN and Gated GNN on FLAG graphs: As FLAG and StockGNN utilize different GNNs, with FLAG using GATv2 and StockGNN using Gated GNN, it is of interest to evaluate the performance of document graphs constructed with FLAG versus document graphs constructed with StockGNN. Therefore, we apply the same configurations of GGNN that StockGNN uses, but on the documents graphs constructed with FLAG, and compare its performance against StockGNN generated graphs, on the financial and technology sectors of the Medya dataset. We trained StockGNN for 100 epochs and GGNN on FLAG graphs for 50 epochs. Table 13 shows the results.

For both the sectors, there is value added in using FLAG-based document graphs, even without using an attention-based GNN, such as GATv2. This indicates that the AMR-based document graphs incorporate important semantic signals for better prediction. We see the same performance trends as the main experiments conducted on the Medya dataset, where the performance improvement in accuracy is less for the financial sector, compared to the technology sector. We delve into why this may be the case in the case study below.

Table 14. Ablation of various different GNNs applied on FLAG graphs.

	Finance			
	Accuracy	Precision	Recall	F1 Score
FLAG	**0.619**	**0.615**	**0.616**	**0.615**
GAT	0.603	0.598	0.599	0.598
GCN	0.565	0.553	0.551	0.550
GGNN	0.547	0.540	0.540	0.540
PNA	0.561	0.281	0.500	0.360
	Tech			
	Accuracy	Precision	Recall	F1 Score
FLAG	**0.637**	**0.641**	**0.638**	**0.635**
GAT	0.619	0.627	0.621	0.614
GCN	0.594	0.595	0.595	0.594
GGNN	0.573	0.574	0.574	0.573
PNA	0.511	0.525	0.516	0.462

Efficacy of attention-based GATv2 – comparison of different GNNs applied on FLAG graphs: Coupling dynamic attention offered by GATv2 with the structure of FLAG-based document graphs is an important aspect of the FLAG methodology and has been shown to achieve superior performance. On account of the large size of the document graphs produced by FLAG, as shown in the dataset metrics in Sec. 4.1, graph transformers cannot be applied on FLAG graphs. However, there are other graph convolution networks that can be applied to FLAG graphs, including conventional GCN, PNA [10], GAT [41], which does not have dynamic attention, and Gated GNN [28] (also compared above). We experiment with these GNNs and present the results in Table 14. We find that GATv2 coupled with FLAG graphs offers an edge in performance that other non-attention-based GNNs or GNNs without dynamic attention cannot match. Also interesting is that the GCN model also outperforms GGNN and PNA on the earnings call graphs.

Table 15. Ablation of various GATv2 configurations of FLAG on predicting daily value-based labels for the financial sector of the Medya dataset. This serves as the ablation for the whole dataset.

# Layers	Accuracy	Precision	Recall	F1 Score
	4 attention heads, 256 dim			
1	0.561	0.281	0.500	0.360
2	0.604	0.600	0.601	0.600
3	0.569	0.588	0.584	0.568
4	0.596	0.598	0.599	0.595
5	0.603	0.598	0.599	0.599
6	0.605	0.596	0.591	0.590
	8 attention heads, 512 dim			
1	0.561	0.281	0.500	0.360
2	0.602	0.598	0.600	0.598
3	0.611	0.607	0.608	0.607
4	**0.619**	**0.615**	**0.616**	**0.615**
5	0.604	0.598	0.598	0.598
6	0.598	0.597	0.598	0.596
	12 attention heads, 768 dim			
1	0.561	0.281	0.500	0.360
2	0.595	0.597	0.598	0.594
3	0.585	0.590	0.591	0.585
4	0.616	0.608	0.606	0.607
5	0.571	0.555	0.546	0.537
6	0.565	0.550	0.510	0.412

Various configurations of GATv2 – number of layers, number of attention heads, and different hidden dimensions: GATv2 is a GNN that has many possible configurations, including the number of layers and attention heads, and different hidden dimensions of transformed input graphs (in FLAG). In our ablations, we experimented with the number of layers ranging from 1 to 6, the number of attention heads varying from 4, 8, and 12, and hidden dimensions of 256, 512, and 768. Table 15 shows this ablation on predicting daily value-based labels for the finance sector of the Medya dataset (we followed the same approach for other datasets and target labels). This table demonstrates the effect on the performance of FLAG with different numbers of GATv2 layers, as we vary the attention heads and the hidden dimensions. Results are shown for the case when the projected dimensionality

for the attention is fixed at 64, be it 64 = 256/4, 64 = 512/8, or 64 = 768/12. The best configuration in terms of accuracy among the ablations is selected as the default for the specific dataset and target label. Generally, we found that to produce an effective result, we need at least 4 layers of GATv2. This makes sense because each layer in GATv2 reaches a node's immediate neighbors. Therefore, 4 layers are sufficient to reach from one node to all other nodes, as the diameter of the document graphs is 4. Moreover, we found that smaller dimensions and number of attention heads performing better for long-term weekly targets, and bigger dimensions and number of attention heads performing better for short-term daily targets.

4.5 Case Study

In order to analyze how the FLAG framework makes its predictions and explore more into the insights it can provide us, we conducted a case study specifically on the daily market-risk adjusted return labels of the S&P 1500 dataset. We chose this more realistic task with the daily timeframe so that we could delve more into the performance of FLAG in a real-world use case scenario.

Table 16. Confusion matrix of FLAG experiment results on predicting next-day market-risk adjusted return trends in the S&P 1500 dataset.

Prediction\Actual	Positive	Negative
Positive	1253	602
Negative	1579	1915

Prediction result analysis: The confusion matrix shown in Table 16 for the market-risk adjusted returns on the S&P 1500 dataset gives us more insights into how the FLAG framework predicted next-day returns. As we can see, the framework is more accurate in correctly predicting negative labels, with much less false positives than false negatives. Therefore, if the model predicts positive returns, we can be fairly certain that there would actually be positive returns. This phenomenon could be different in each sector, as indicated by previous analyses on main experiment results, so we delve into the sector-wise results more below.

Table 17. Sector-wise result quality analysis of FLAG on daily market-risk adjusted returns trend prediction. Best metrics marked in bold. Worst metrics underlined.

Sector\Prediction metric	Accuracy	Precision	Recall	F1 Score
Consumer Discretionary	0.616	0.628	0.621	0.612
Industrials	0.585	0.606	0.600	0.583
Technology	0.660	0.664	**0.666**	**0.660**
Health Care	0.560	0.581	0.562	0.534
Finance	0.559	0.577	0.566	0.547
Energy	0.570	0.578	0.580	0.569
Consumer Staples	0.576	0.640	0.613	0.566
Real Estate	.522	.554	.538	.493
Utilities	0.567	0.582	0.563	0.539
Telecommunications	**0.683**	**0.685**	0.655	0.655

Table 18. Sector-wise percentages of each classification category for FLAG on daily market-risk adjusted returns task. Highest percentages marked in bold. Lowest percentages underlined.

Sector\Percentages	TP	TN	FN	FP
Consumer Discretionary	25.94%	35.67%	26.10%	12.29%
Industrials	25.38%	33.17%	29.90%	11.56%
Technology	**34.80%**	31.25%	1.45%	12.50%
Health Care	16.35%	39.62%	34.23%	9.81%
Finance	19.80%	36.08%	32.23%	11.88%
Energy	31.00%	6.00%	28.00%	**15.00%**
Consumer Staples	21.20%	36.41%	36.41%	.98%
Real Estate	4.04%	38.18%	**38.92%**	8.87%
Utilities	15.97%	40.76%	33.19%	10.08%
Telecommunications	19.80%	**48.51%**	22.77%	8.91%

Sector-wise result analysis: To examine the performance of FLAG in a sector-wise perspective on the market-risk adjusted extension of the S&P 1500 dataset, we identified the sectors of most of the data points in the testing set, and picked 10 major sectors in terms of number of data points. They are: consumer discretionary, industrials, technology, health care, finance, energy, consumer staples, real estate, utilities, and telecommunications. We analyzed both the quality metrics of classification in each sector and the percentages of each classification category for all sectors. Tables 17 and 18 show the results of this analysis.

As we can observe, FLAG performed the best in digital sectors like technology and telecommunications. It had medium performance in manufacturing and retail sectors such as consumer discretionary, industrials, and consumer staples. And the worst performance we see in service sectors such as health care, finance, energy, and real estate. In fact, the real estate sector was the worst performing sector both judging from quality of classification and the percentage of false negatives, which is the major category of false classifications.

Analysis of sample documents using GNN Explainer [46]: Since we observe varying degrees of performance in different sectors, we sampled 3 randomly selected representative documents from each of the 4 classification categories, keeping in mind the spectrum of performances we saw in different sectors. To generate individual instance-level explanations, we trained GNN Explainer [46] with 3 hops for 1000 epochs to explain each graph, generating an edge mask for each graph, which contains a score for each and every edge in the graph, $EM = \{S(e_1), S(e_2), ..., S(e_n)\}$, where n is the number of edges in the graph and $S(e_i)$ is a numerical score representing the importance of edge e_i. Based on this edge mask, we looked specifically at the document virtual node that represents the whole document, and the sentence virtual node that represents each sentence. For each sentence j in document i, we calculate the importance of the sentence j to be:

$$I(s_j) = S(e(sn_j, dn_i)) \tag{9}$$

where $I(s_j)$ denotes the importance score of sentence j, sn_j denotes the sentence virtual node j, dn_i denotes the document virtual node of document i, and $e(u, v)$ denotes the edge going from node u to node v. With this scoring method, we then extracted the top 30 sentences from each document that were considered to be most important for the final classification, and analyzed them according to its classification category.

Analysis of positively classified sample documents: First, we look at the documents that were predicted to generate positive next-day market-risk adjusted returns, shown in Tables 19 and 20.

For **true positive** documents, in the important sentences, we observe that the C-suite is commenting on the strong earnings and concrete favorable developments that quarter in a very positive manner, mentioning the

Table 19. Sampled important sentences that FLAG deemed to be important from earnings calls predicted correctly to generate positive next-day market-risk adjusted returns.

True Positives		
TechTarget Inc., Q2 2022 (Telecommunications)	**T-Mobile US Inc.**, Q2 2022 (Telecommunications)	**Motorola Solutions Inc.**, Q2 2022 (Technology)
And we expect that if you look at our total addressable market that we've identified in about customers, potential customers and prospects, we see that there's a lot of upside on that.	Altogether , we see as another year of profitable growth and free cash flow expansion as we continue to invest in our network and the business.	First , Q2 was exceptional across the board with revenue and earnings per share both coming in above our expectations.
So what does that do? It provides really good content for us.	So again, we're in those of POPs where we already have a right to win, we are winning.	Sequentially, backlog was down, driven primarily by unfavorable FX and revenue recognition for the Airwave and ESN contracts, partially offset by record second quarter orders in Command Center Software.
Clearly , you're not seeing any of that impacting the fundamentals .	Because we've been very clear eyed about this strategy for many years, and I think that consistency is something that people should acknowledge.	We think there's great potential there.
And we're actually seeing some of our overall revenue, which was historically only Priority Engine, these integrated online content syndication, QSO, branding and Priority Engine turn into long term contracts.	And we're eating into that. We're a couple of thousand in, and we'll be continuing to build as we move through this year and next.	And I think that the demand for that broad portfolio continues to be strong and maybe perhaps is even getting stronger.
Maybe first just on the upside to the quarter and to the increased guidance for the year.	I think this product has been a fantastic showcase of what the leading 5G network can really do.	We had record Q2 product orders at higher prices, including our single largest quarter ever for PCR orders , and we ended the quarter with record Q2 ending backlog.

"upside", "profitable growth", and things being "above our expectations", etc., focusing on historical performance as evidence for the firm's financial fortitude and expected future performance, and FLAG is picking up on that and considering those statements to be important. And generally, we don't see too much forward-looking statements, and even if we do observe them make forward-looking statements, it is usually an add-on in the context of positive past performance, and with quite strong confidence levels. From a sector's perspective, it makes sense that FLAG is able to pick up on those positive signals coming from the technology and telecommunications industries, as positive developments in software and connectivity directly translates to creation of value for companies in the short term.

As for **false positive** documents, in the important sentences, we see the C-suite mostly commenting on past developments in a positive manner as well, which is expected. However, the main topics that they focused on were not the actual earnings or new developments in company growth, but soft and conceptual ideas. In the

Table 20. Sampled important sentences that FLAG deemed to be important from earnings calls predicted incorrectly to generate positive next-day market-risk adjusted returns.

False Positives		
Biogen Inc., Q3 2022 (Health Care)	**Whitestone REIT**, Q4 2022 (Real Estate)	**KKR Real Estate Finance Trust Inc.**, Q1 2022 (Real Estate)
And obviously, we'll not speculate on lecanemab, but it's a very important event.	Importantly, we've improved morale throughout the company by giving employees more effective tools for success and clear objectives that were aligned with the management team, which is resulting in greater productivity.	So we are a much more relevant counterparty for banks, for brokers in the market for operating partners, and that translates, I think, into better looks and just that connectivity with the overall market.
This is, of course, post Clarity AD readout should be positive.	We view remerchandising as a critical component of ensuring the center is connected to the ceramic community.	This broader product suite helped drive over $14 billion in total KKR Real Estate credit originations in 2021, which has led to broader and deeper relationships.
It is a well powered, well designed trial.	I'll just start by mentioning that the vast majority of our leases are triple net leases, meaning that we recover our common area expenses, real estate taxes, and insurance costs from our tenants.	In January, we raised approximately $155 million of gross proceeds through a follow-on issuance of our Series A preferred shares at a fixed for life cost of 6.5%.
The BLA of mosunetuzumab for the same indication was recently granted priority review by the FDA, and we look forward to a potential approval in the US Our progress across these areas in addition to the recent advancements we have made in R&D have the potential to help drive growth over time.	And I'll remind you that our lease structures are strategically designed with shorter lease terms, allowing for more frequent lease rate increases.	I think what it mostly translates back into well, first and foremost, is market connectivity.
In conclusion, we executed well against our R&D objectives in the second quarter and continue to prioritize our efforts across both therapeutic areas and programs.	I think when we think about the dispositions and the investment in Pillarstone, I think that is just available capital that we can redeploy in a way that creates more value.	Post-COVID obviously us and all of our peers have increased the CECL reserve significantly as a result of the impact of COVID on the macro environment.

health care context, this means trials in R&D that can have positive results or favorable sentiments, but it does not directly translate to increased tangible short-term value growth for the company. And in the real estate context, this means concepts like "morale" and "market connectivity", which can sound positive, but perhaps are veneering troublesome underlying conditions. The issue with this phenomenon is that without numerical signals coming from quantitative factors extracted from quarterly earnings, it is difficult to decipher the textual signals by themselves into a precise positive/negative prediction. This is certainly exemplified by the relatively worse performance that FLAG has in sectors like health care and real estate.

Table 21. Sampled important sentences that FLAG deemed to be important from earnings calls predicted correctly to generate negative next-day market-risk adjusted returns.

True Negatives		
Liquidity Services Inc., Q3 2022 (Consumer Discretionary)	**Teledyne Technologies Inc.**, Q2 2022 (Industrials)	**Adeia Inc.**, Q4 2022 (Technology)
When you are looking at businesses like GovDeals, the Capital Assets Group, Machinio, that's very high margin business as a percentage of GAAP revenue.	In Digital Imaging now, we expect it to be lower by 130 basis points from what - for the full year.	As of today, much of the market remains a significant opportunity for us.
We are making excellent progress in executing our strategic plan, and we're very excited about the opportunity that lies ahead to expand our market leadership in the $100 billion circular economy.	There are different reasons for it, for example, in our health care, Digital Imaging because the COVID things went soft.	And so the negotiations and discussions with the other parties are going well, but we just weren't ready to close them out at that time.
We are doing a lot of tech integration to support self-directed sales, but we are responding to client needs.	Two, we were dealing with fairly successfully, and that would be overall inflation and basically parts shortages. We seem to be rolling $60 million every quarter over and over.	There currently could be a situation where we're negotiating a contract in general commercial terms that we need to. And as Paul said, we need to put up over the line and get the best turn to begin, and that might slip a month or something like that.
I think that the infrastructure that we have today certainly could take us another $0.5 billion of sales, so let's say, up to $2 billion of annual GMV.	If you create a vacuum eventually, air comes in, right? So you got these brokers that are doing pretty good work and making a lot of money. When that happens, supply chain is going to change eventually, and it is.	Our media business represents more than 90% of our baseline revenue and several aspects of the media business represent important areas of growth for us.
This supplemental operating data includes gross merchandise volume and should not be considered a substitute for or superior to GAAP results.	I think what we're looking at is improving our revenue there in the fourth quarter - in the third and fourth quarter better than we have in the first two quarters and mostly in the fourth quarter.	As we have mentioned before, we remain committed to getting the most beneficial deals done for Adeia, which can sometimes lead to a pushout in the timing.

Analysis of negatively classified sample documents: Next, we look at the documents that were predicted to generate negative next-day market-risk adjusted returns, shown in Tables 21 and 22.

For **true negative** documents, when we look at the important sentences, we see two main types of statements: defensive statements about past earnings, and positive forward-looking statements about the future, without basis in past performance. Usually, this occurs when the actual earnings are negative, and the C-suite has to defend the value of their firms and try to change the market's outlook for its future performance. Of course, without solid good performance in the past, they tend to make more forward-looking statements to "promise"

Table 22. Sampled important sentences that FLAG deemed to be important from earnings calls predicted incorrectly to generate negative next-day market-risk adjusted returns.

False Negatives		
SLM Corporation, Q2 2022 (Finance)	**Equity Residential**, Q2 2022 (Real Estate)	**Xencor Inc.**, Q1 2022 (Health Care)
And we are very well reserved for the outlook we are discussing here this morning.	Let me start with a huge shout out to the entire Equity Residential team for their continued dedication and hard work.	We fully expect Genentech to expand beyond what they're already doing with atezolizumab combos and now dara combos, and you'll hear more from Xencor.
As we are all aware, base interest rates and credit spreads have increased significantly since our last loan sale, and capital markets have been extremely volatile in a difficult economic and political environment.	And so that should help us and hopefully offset other line items that might return to more normal growth.	Remember, the BMS program when they looked at the check mate, I believe, it was the 650 study in combination of nivo and ipi, the combo was better.
In addition, the **drag** on our NIM from our liquidity portfolio **declined meaningfully** as we invested our cash and medium term treasuries as interest rates have increased rather than leaving cash on deposit in low yielding federal reserve balances.	On the expense side, we left our already strong guidance unchanged with June year-to-date expense growth of 2.8% and a number of innovation initiatives underway, we feel confident in our ability to deliver [indiscernible] **low expense growth** for the full year.	We are hopeful that, that allows us to have a differentiated profile that may be toxicity limited the efficacy because they couldn't get to enough dose, perhaps of the historical molecules.
And so even as some of those trends shift, we're well positioned to take advantage of those.	After significant impact from the pandemic, New York same-store revenue and NOI are now fully recovered back to 2019 levels.	For tidutamab, our SSTR2 x CD3 bispecific antibody and XmAb841, our CTLA-4 x LAG-3 bispecific antibody, after reviewing the data generated to date, we believe that neither program has a competitive enough clinical profile in their respective areas, particularly when compared to the programs we are currently advancing.
With that said, we will always continue to evaluate the market, the market for our equity, the market for our loans.	So I guess our sense is the supply demand balance is pretty good for us, and it will likely be pretty good unless a recession if a recession occurs, is very severe .	Also, for our CD3 platform, soon, we anticipate dosing the first patient in a Phase I study evaluating our ENPP3 targeting CD3 bispecific antibody, XmAb-819 in patients with renal cell carcinoma.

certain things for the future, without necessarily historical evidence to back them up. And we see FLAG being able to pick up these features and make the correct negative prediction. From a sector's perspective, we see that in industries like consumer discretionary, industrials, and technology, the market is able to focus on more tangible things, such as specific sub-businesses like "GovDeals", "Digital Imaging", and the media business. Therefore,

the C-suite has to get in front of those issues and both defend their values and project positive outlooks going forward.

As for **false negatives**, which are more prevalent than false positives, when we examine important sentences that FLAG picked up, even though we do not see the same level of strongly defensive patterns we have observed in the true negative documents, where the C-suite needs to provide explanations for negative past performance, we do see the same trend of more forward-looking statements being made, without being an add-on to positive past performance or developments. This may have contributed partially to false classifications for these documents and caused FLAG to "think" that the C-suite is trying to paint a rosy picture without tangible evidence to back it up. Moreover, we see FLAG picking up sentences that we do not expect it to "understand". In the finance and real estate sectors, this can be the "drag" on NIM (net interest margins) declining, or delivering "low expense growth", which are positive things but sound negative to a model without explicitly encoded reasoning capabilities. In the health care sector, this can be technical details regarding different clinical trials and their expected values to the company.

Table 23. Forward-looking statements analysis results across **all** documents in each of the 4 classification categories. "Spec." stands for "specific".

True Positives	Not FLS	Spec. FLS	Non-spec. FLS
Total count of sentences	32983	3187	2033
Percentage of sentences	86.336	8.342	5.322
Avg. score of classification	0.926	0.634	0.490
True Negatives			
Total count of sentences	49797	5213	3322
Percentage of sentences	85.368	8.937	5.695
Avg. score of classification	0.925	0.644	0.499
False Negatives			
Total count of sentences	41092	4232	2743
Percentage of sentences	85.489	8.804	5.707
Avg. score of classification	0.925	0.644	0.499
False Positives			
Total count of sentences	15816	1522	1017
Percentage of sentences	86.167	8.292	5.541
Avg. score of classification	0.925	0.627	0.491

Analysis of forward-looking statements from all documents in the test set: From our previous analyses, we have observed that one facet of earnings calls that FLAG has been trained to look at is forward-looking statements. Within the sentences it deemed to be important, too much focus on forward-looking statements indicates an attempt by the C-suite to "guide" the market to have a positive outlook for its future, without having good enough earnings or positive developments in the past to talk about. And this can contribute to FLAG casting a negative outlook on financial performance. To statistically analyze this hypothesis on forward-looking statements, we took **all** the documents in the test set, trained a GNN Explainer [46] (for 100 epochs only) to explain the document virtual node's 1-hop neighbors, which are essentially all the sentence virtual nodes, and scored each sentence according to the same Equation 9 as in the analysis of sample documents, and extracted the top 30 sentences FLAG considers to be important. Then, we applied the forward-looking statement (FLS) classifier of FinBERT [45] on all of the important sentences across each of the 4 classification categories, and Table 23 shows the result of this analysis.

The classifier labels each sentence as one of the three: not FLS, specific FLS, or non-specific FLS. Specific FLS means that the sentence is projecting into the future but is still grounded in specific topics. This could be about the company, the sector, or the economy as a whole. Non-specific FLS means that the sentence is projecting into the future without grounding in any specific topic. Instead, the sentence is generic. This can be important as they can indicate different signals to the market regarding future financial performance. However, this can also depend on the context within which a forward-looking statement is made. As shown, for important sentences picked up by FLAG from the documents that it predicted positive returns for, including both true and false positives, the percentage of non-FLS is consistently higher, the percentage and confidence level (demonstrated by the average score of classification) of both specific and non-specific FLS are consistently lower, than those of the documents for which FLAG predicted negative returns, including both true and false negatives. This substantiates the forward-looking statement hypothesis that gives us insights into how FLAG makes its decisions.

5 Conclusion and Future Works

This work presents the first use of AMR-based graphs in a predictive framework for long document classification tasks, namely, financial trend prediction. As demonstrated by the extensive experiments performed, in predicting both immediate and longer-term price trends based on textual signals from earnings calls, FLAG with underlying FinBERT-generated embeddings is able to show superior performance against both graph-based and LLM/LM (both encoder modeled as classifier and decoder modeled as Q&A instruction fine-tuning) baselines. This demonstrates the beneficial value added by incorporating semantics in predictive NLP tasks usually performed by LLM/LMs.

For future work, as we have seen the performance of FLAG vary from sector to sector, which warrants further analysis into the characteristics of earnings calls in different sectors, their interactions with the market, and how FLAG learns from them, it is of interest to utilize model-level explanation methods to identify document-level patterns that the model considers as important to its prediction, whether positive or negative, both in specific sectors and across all sectors. Compared with the instance-level explanations that we analyzed in the case study of this paper, model-level explanations seek to uncover a generic graph pattern or motif that maximizes a particular prediction. Thus, they have the potential to offer more high-level insights into what FLAG has learned from these semantically meaningful document graphs.

Moreover, in the real-world use case scenario of price trend analysis, financial analysts make predictions from an aggregation of a variety of quantitative factors, and not only from textual signals. Therefore, we want to find ways to incorporate the textual signals offered by the FLAG framework into real-world quantitative methods used in the financial industry, so that it can add value to actual investment recommendation use cases.

Beyond predictive financial document analysis, there is potential applicability for long document analysis in other domains as well. In the legal domain, FLAG can potentially be applied to legal judgment prediction (including predicting case outcome, identifying violated articles, and estimating prison terms), where there are many datasets such as LexGLUE [8], as well as to contract review (including risk assessment, clause classification, and anomaly detection), using datasets such as LEDGAR [40]. In the medical domain, FLAG can be used to automate ICD coding (i.e., predicting the code to classify diseases, injuries, and other health conditions for various administrative and clinical purposes) and patient outcome prediction, for which there are extensive datasets, such as MIMIC-III [20] and MIMIC-IV [19]. In the scientific text domain, FLAG can be applied to document classification tasks, such as ArXiv category prediction [2]. FLAG can also be adapted for information extraction tasks, such as keyphrase extraction [30]. Overall, the FLAG framework is a flexible approach that tackles the challenge of analyzing long documents in a novel manner, and in addition to what we mentioned above, its applicability can be explored in many other domains involving long document analysis.

Finally, as we have seen, the benefits of incorporating structured data, such as the FLAG semantic graphs, to LLM/LM tasks dealing with unstructured text data have been substantiated in this work, and we believe its

potential does not stop at only predictive tasks. As generative AI is being adopted more and more in the financial industry, the problems of LLMs with reliability, consistency, and explainability are becoming more and more apparent. Therefore, it is of great interest to investigate the potential of incorporating structured data such as knowledge graphs (KGs) in the financial domain, in various LLM Q&A and generative AI use case scenarios spanning from financial analytics to compliance, customer service, market research, and much more. As we advance towards this goal, we continue to engage in dialogues and collaborations with stakeholders in the financial industry to explore relevant use cases and to identify processes where structured knowledge can add value.

Acknowledgments

This work was supported by an industry funded award from the RPI-Stevens NSF IUCRC Center for Research toward Advancing Financial Technologies (NSF Award #: 2113850).

References

[1] AIMeta. 2024. Llama 3 Model Card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.
[2] arXiv.org submitters. 2024. arXiv Dataset. doi:10.34740/KAGGLE/DSV/7548853
[3] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *7th linguistic annotation workshop and interoperability with discourse*. 178–186.
[4] Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders. In *First Conference on Language Modeling*. https://openreview.net/forum?id=IW1PR7vEBf
[5] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *CoRR* abs/2004.05150 (2020). arXiv:2004.05150 https://arxiv.org/abs/2004.05150
[6] Steven Bird and Edward Loper. 2004. NLTK: The Natural Language Toolkit. In *ACL Interactive Poster and Demonstration Sessions*. 214–217.
[7] Shaked Brody, Uri Alon, and Eran Yahav. 2022. How Attentive are Graph Attention Networks?. In *International Conference on Learning Representations*.
[8] Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. LexGLUE: A Benchmark Dataset for Legal Language Understanding in English. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Dublin, Ireland, 4310–4330. https://aclanthology.org/2022.acl-long.297
[9] James Chen. 2021. Earnings call. https://www.investopedia.com/terms/e/earnings-call.asp
[10] Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. 2020. Principal Neighbourhood Aggregation for Graph Nets. In *Advances in Neural Information Processing Systems*.
[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805
[12] Andrew Drozdov, Jiawei Zhou, Radu Florian, Andrew McCallum, Tahira Naseem, Yoon Kim, and Ramón Astudillo. 2022. Inducing and Using Alignments for Transition-based AMR Parsing. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1086–1098.
[13] Ermal Elbasani and Jeong-Dong Kim. 2022. Amr-CNN: Abstract meaning representation with convolution neural network for toxic content detection. *Journal of Web Engineering* (2022). doi:10.13052/jwe1540-9589.2135
[14] Fama-French. 2025. https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/ftp/F-F_Research_Data_Factors_daily_CSV.zip.
[15] Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient Text-To-Text Transformer for Long Sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (Eds.). Association for Computational Linguistics, Seattle, United States, 724–736. doi:10.18653/v1/2022.findings-naacl.55
[16] S&P Dow Jones Indices. 2025. https://www.spglobal.com/spdji/en/indices/equity/sp-composite-1500/#overview.
[17] Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. FinanceBench: A New Benchmark for Financial Question Answering. *arXiv 2311.11944* (2023).
[18] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut

Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. arXiv:2310.06825 [cs.CL] https://arxiv.org/abs/2310.06825

[19] Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2021. MIMIC-IV. doi:10.13026/s6n6-xd98

[20] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data* 3 (2016), 160035. doi:10.1038/sdata.2016.35

[21] Dan Jurafsky and James H. Martin. 2024. *Speech and Language Processing* (3 (draft) ed.). https://web.stanford.edu/~jurafsky/slp3/

[22] Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *31st International Conference on International Conference on Machine Learning*.

[23] John Sie Yuen Lee, Ho Hung Lim, and Carol Webster. 2022. Unsupervised Paraphrasability Prediction for Compound Nominalizations. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3254–3263.

[24] Young-Suk Lee, Ramón Astudillo, Hoang Thanh Lam, Tahira Naseem, Radu Florian, and Salim Roukos. 2022. Maximum Bayes Smatch Ensemble Distillation for AMR Parsing. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 5379–5392.

[25] Changmao Li and Jeffrey Flanigan. 2022. Improving Neural Machine Translation with the Abstract Meaning Representation by Combining Graph and Sequence Transformers. In *2nd Workshop on Deep Learning on Graphs for Natural Language Processing (DLG4NLP 2022)*. 12–21.

[26] Irene Li, Aosong Feng, Dragomir Radev, and Rex Ying. 2023. HiPool: Modeling Long Documents Using Graph Neural Networks. In *Proceedings of the Association for Computational Linguistics (ACL)*.

[27] Xiang Li, Thien Huu Nguyen, Kai Cao, and Ralph Grishman. 2015. Improving event detection with abstract meaning representation. In *1st workshop on computing news storylines*. 11–15.

[28] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard S. Zemel. 2016. Gated Graph Sequence Neural Networks. In *4th International Conference on Learning Representations*.

[29] Xiao-Yang Liu, Jie Zhang, Guoxuan Wang, Weiqin Tong, and Anwar Walid. 2024. Efficient Pretraining and Finetuning of Quantized LLMs with Low-Rank Structure. In *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*. 300–311. doi:10.1109/ICDCS60910.2024.00036

[30] Debanjan Mahata, Navneet Agarwal, Dibya Gautam, Amardeep Kumar, Swapnil Parekh, Yaman Kumar Singla, Anish Acharya, and Rajiv Ratn Shah. 2022. LDKP: A Dataset for Identifying Keyphrases from Long Scientific Documents. arXiv:2203.15349 [cs.CL]

[31] Sourav Medya, Mohammad Rasoolinejad, Yang Yang, and Brian Uzzi. 2022. An Exploratory Study of Stock Price Movements from Earnings Calls. In *Companion Proceedings of the Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. 20–31.

[32] Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *International Conference on Learning Representations*.

[33] Tahira Naseem, Austin Blodgett, Sadhana Kumaravel, Tim O'Gorman, Young-Suk Lee, Jeffrey Flanigan, Ramón Astudillo, Radu Florian, Salim Roukos, and Nathan Schneider. 2022. DocAMR: Multi-Sentence AMR Representation and Evaluation. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3496–3505.

[34] Tahira Naseem, Abhishek Shah, Hui Wan, Radu Florian, Salim Roukos, and Miguel Ballesteros. 2019. Rewarding Smatch: Transition-Based AMR Parsing with Reinforcement Learning. In *57th Annual Meeting of the Association for Computational Linguistics*. 4586–4592. doi:10.18653/v1/P19-1451

[35] Tim O'Gorman, Michael Regan, Kira Griffitt, Ulf Hermjakob, Kevin Knight, and Martha Palmer. 2018. AMR Beyond the Sentence: the Multi-sentence AMR corpus. In *27th International Conference on Computational Linguistics*. 3693–3702.

[36] OpenAI. 2024. GPT-4 Technical Report. *arXiv2303.08774* (2024).

[37] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Conference on Empirical Methods in Natural Language Processing*.

[38] William F. Sharpe. 1964. CAPITAL ASSET PRICES: A THEORY OF MARKET EQUILIBRIUM UNDER CONDITIONS OF RISK. *The Journal of Finance* 19, 3 (1964), 425–442. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1964.tb02865.x doi:10.1111/j.1540-6261.1964.tb02865.x

[39] Ziyi Shou, Yuxin Jiang, and Fangzhen Lin. 2022. AMR-DA: Data Augmentation by Abstract Meaning Representation. In *Findings of the Association for Computational Linguistics: ACL 2022*. 3082–3098.

[40] Don Tuggener, Pius von Däniken, Thomas Peetz, and Mark Cieliebak. 2020. LEDGAR: A Large-Scale Multi-label Corpus for Text Classification of Legal Provisions in Contracts. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (Eds.). European Language Resources Association, Marseille, France, 1235–1241. https://aclanthology.org/2020.lrec-1.155/

[41] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018).
[42] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. 2019. Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks. *arXiv preprint arXiv:1909.01315* (2019).
[43] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. BloombergGPT: A Large Language Model for Finance. *arXiv 2303.17564* (2023).
[44] Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. FinGPT: Open-Source Financial Large Language Models. *arXiv 2306.06031* (2023).
[45] Yi Yang, Mark Christopher Siy UY, and Allen Huang. 2020. FinBERT: A Pretrained Language Model for Financial Communications. *arXiv 2006.08097* (2020).
[46] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. 2019. GNNExplainer: generating explanations for graph neural networks. In *33rd International Conference on Neural Information Processing Systems*.
[47] Jiawei Zhou, Tahira Naseem, Ramón Fernandez Astudillo, Young-Suk Lee, Radu Florian, and Salim Roukos. 2021. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. In *Conference on Empirical Methods in Natural Language Processing*. 6279–6290.

Received 20 February 2025; revised 13 April 2026; accepted 5 April 2026