Abstractive Financial News Summarization via Transformer-BiLSTM Encoder and Graph Attention-Based Decoder

Haozhou Li, Qinke Peng, Xu Mou, Ying Wang, Zeyuan Zeng, and Muhammad Fiaz Bashir

***Abstract*—Financial news summarization (FNS) has been an attractive research problem in recent years, which aims to generate a shorter highlight of the news article while preserving key factual aspects, emotions, and opinions, providing significant assistance in stock trading and investment decision-making. However, FNS faces two challenges compared to the common domain. Firstly, financial news involves professional qualitative and quantitative information and salient content always scatters across long-range interactions. Secondly, financial news contains latent causal relationships, where historical information in the early generated sequence can significantly affect the subsequent decoding process. To address these difficulties, we propose an enhanced Seq2Seq model named TLGA, where the hierarchical Transformer-BiLSTM encoder can capture long-range interactions and sequential semantics while the Graph Attention-based decoder can fully utilize the historical information of decoded tokens and capture key causal relations. Moreover, we propose history-enhanced attention to concentrate on salient input content based on history semantics, guiding our decoder to generate the summary around the corresponding contents. It is also the first attempt to reuse history information of previously generated summary sequences in FNS using the idea of the Graph Attention Mechanism. Additionally, we construct the LCFNS dataset with 430,820 news-summary pairs for the lack of large-scale high-quality datasets in FNS. Experimental results on two financial datasets and two benchmark datasets indicate that our model outperforms other baselines.**

***Index Terms*—Graph attention mechanism, financial news, Seq2Seq, transformer, text summarization.**

I. Introduction

With the rapid growth of the Internet, major online portals publish an enormous volume of real-time financial news containing a wealth of information [1]. Exploiting key factual aspects, sentiment, and opinions in financial news plays a vital role in stock market prediction [2], [3], [4], sentiment analysis [5], [6], [7], and decision-making [8], [9]. However, due to the explosive growth of various online media, it is difficult for both individual and institutional investors to timely obtain salient information from such a massive amount of financial news. Thus, we need the text summarization technique to generate concise and informative summaries and alleviate information overload [10], which can significantly assist stock trading and help investors make better decisions [11].

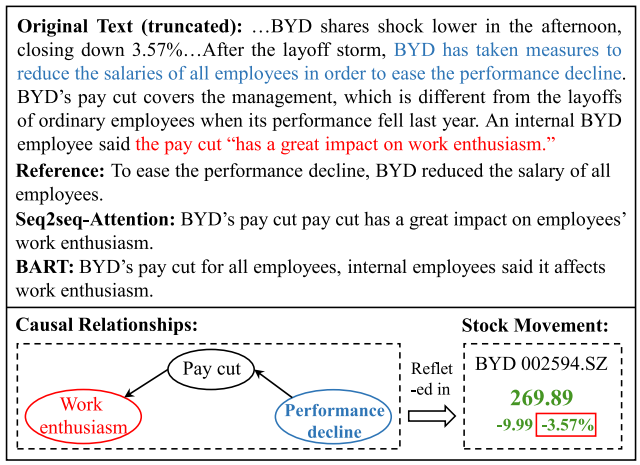

Fig. 1. News article and its reference summary. The blue words indicate crucial information in the reference summary; the red words indicate minor content focused by Seq2Seq-Attention and BART. And all causal relationships are eventually reflected in the fluctuations of BYD share price.

In recent years, financial news summarization (FNS) has been an attractive research problem, seeking to reduce the length of the original article while preserving salient events, emotions, and opinions that affect the fluctuations of financial markets [2]. However, compared with common-domain, FNS has main two difficulties: (1) financial news contains professional qualitative and quantitative information, and salient content always scatters across long-range interactions. (2) financial news contains latent causal relationships, and historical information in the early generated sequence can significantly affect the subsequent decoding process. Fig. 1 shows an example, both Seq2seq-Attention and BART ignore the key causal relation between "performance decline" and "pay cut", and all these causal relationships are eventually reflected in BYD's share price movement. For FNS, therefore, an effective model should be capable of learning long-range interactional semantics and capturing salient causal relations from redundant content, which can help predict stock movement and make investment decisions [9], [12].

Manuscript received 24 August 2022; revised 10 January 2023 and 12 June 2023; accepted 31 July 2023. Date of publication 11 August 2023; date of current version 24 August 2023. This work was supported by the National Natural Science Foundation of China under Grant 61872288. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jing Huang. *(Corresponding author: Qinke Peng.)*

The authors are with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: lihaozhou1126@stu.xjtu.edu.cn; qkpeng@xjtu.edu.cn; muxu19950916@xjtu.edu.cn; ying_wang@xjtu.edu.cn; zengzeyuan@stu.xjtu.edu.cn; mfiaz1996@gmail.com).

Digital Object Identifier 10.1109/TASLP.2023.3304473

Most existing studies on financial text summarization have relied on extractive methods, where salient sentences are directly selected as summaries. Previous study utilized the graph-based sentence ranking method to extract the summaries of stock news [8]. Recent studies appeared in the FNS 2020 shared task, where researchers focused on analyst reports from U.K. corporations and proposed series of extraction approaches [13], [14], [15]. However, these methods cannot consider similar sentences with complementary information [16] and get poor coherence. Recently, abstractive methods with encoder-decoder frameworks [17], [18] have been popular with the development of deep learning, generating novel words instead of directly extracting them. Furthermore, attention mechanisms have been proposed to enhance Seq2Seq models [19], [20], [21], [22], [23], [24], [25], [26]. For instance, [20] first selected relevant sentences using PageRank and then utilized Pointer-Generator [21] to generate the summary of financial news. However, these methods adopt LSTM or GRU as the encoder, ignoring long-range interaction information of financial news and suffering from the long-term dependence problem [27]. Nowadays, the Transformer [28] and Pre-trained Language Models (such as BART [29], UniLM [30], and BigBird [31]) have achieved promising results in open-domain. As for financial news summarization, however, the lack of large-scale domain-specific training data limits the performance of these models to some extent.

Due to their tremendous capability for encoding relational information [32], Graph Neural Networks (GNNs) have been used to enhance the encoder-decoder framework [33], [34], [35], [36], [37], [38]. Liang et al. [34] integrated Gated Graph Neural Networks to capture long-term information. [35] first extracted fact triples and then adopted an additional GAT [39] to represent them to enhance the encoder. However, they all improve the encoder via graphs, but few of these studies focus on reusing historical information in previous decoding states. As we mentioned above, financial news contains rich causal relationships, and the semantics of history summary sequences can significantly influence the subsequent steps. Unfortunately, RNN stores all history information via a fixed-size vector [40], losing salient history semantics due to the gate mechanism [41].

Moreover, the lack of large-scale high-quality datasets is also an urgent problem. Although [13] has released an English annual reports dataset, its size is still too small, only containing thousands of samples. As far as we know, there is no large-scale Chinese corpus publicly available in FNS.

To address these problems, we propose an enhanced Seq2Seq model with stacked **T**ransformer-Bi**L**STM encoder and **G**raph **A**ttention-based decoder for **F**inancial **N**ews **S**ummarization (**TLGA-FNS**). Firstly, we utilize Transformer to capture long-range interactional information and then learn sequential semantics using BiLSTM, solving the long-term dependence problem. Secondly, we propose the Graph Attention-based decoder to learn latent causal relationships in financial news. Since cause and effect often appear sequentially, once capturing either of them, our proposed decoder can reuse history information in early generated tokens and guide to decode summaries surrounding cause or effect. Unlike traditional Seq2Seq attention [20], [21], [22], we further propose history-enhanced attention, which calculates the contextual attention weights based on the historical information of all previous decoding states. Thus it can dynamically adjust the attention to the salient input content related to these history semantics. Finally, to solve the lack of large-scale datasets problem for FNS, we construct the LCFNS dataset and we evaluate TLGA on two financial datasets and two benchmark datasets, and experimental results indicate the effectiveness of our model.

The main contributions of this article are as follows.

1) A hierarchical Transformer-BiLSTM encoder is proposed to learn long-range interactions and sequential information in financial news.
2) A Graph Attention-based decoder with history-enhanced attention is proposed to fully utilize history information to learn key causal relationships.
3) A new Large-scale Chinese Financial News Summarization dataset (LCFNS) is constructed, containing 430,820 samples. Experimental results on four datasets indicate that our model outperforms other baseline models.

The remainder of this article is as follows. Section II reviews related work and Section III introduces our proposed model. In Section IV, we show experimental details, results, and discussion. Finally, Section V draws the conclusion.

II. Related Work

A. Financial Text Summarization

Financial text summarization seeks to condense the original document into a shorter version while preserving critical factual aspects, emotions, and opinions, which provides significant support and assistance in predicting stock movements [9], [11]. Previous research adopted unsupervised graph-based sentence ranking methods [8], [42] to obtain the summaries of stock news but got lower coherence. [43] encoded news articles using CNN and LSTM and then extracted summaries via a supervised sentence classifier. Recent studies on financial text summarization appeared in the FNS 2020 shared task [13]; researchers focused on the analyst reports of U.K. corporations. [14] constructed a hierarchical framework using the discourse parsing approach and the Latent Dirichlet Allocation model. [15] proposed a CNN-based classifier to select key sentences as summaries. Additionally, Agrawal et al. [9] proposed a hierarchical model with BERT [44] and BiLSTM to extract summaries and predict the stock market movement. However, most existing studies are extractive methods that directly select salient sentences from the source text as summaries, which is incapable of fully considering similar sentences with complementary information [16]. Since financial news always contains a wealth of professional information and redundant content, these methods have limited scalability and versatility.

B. Abstractive Summarization

1) RNN-Based Methods: Instead of directly extracting sentences, abstractive models usually use the encoder-decoder framework to generate novel words [17], [18]. Zhang et al. [20]

generated summaries of financial news using the Pointer-Generator Network [21], which can not only generate words from the vocabulary but also copy terms from the source text. Recent work attempts to represent texts by enhancing the encoder. [23] introduced an auto-encoder to represent the reference summaries and used adversarial learning to supervise the model. Gui et al. [26] proposed Attention Refinement Unit to concentrate on the salient information, ignoring irrelevant semantics. Yao et al. [45] proposed a dual encoding mechanism to obtain a more accurate contextual representation. However, these methods all rely on the RNN-based encoder to represent the input text, losing long-range interactions and suffering from long-term dependence problem [27].

2) Transformer-Based Methods: As for Transformer-based methods with multi-head attention and masked mechanism [28], [29], [30], [31], Lewis et al. [29] constructed a bidirectional transformer encoder and an auto-regressive decoder, pre-trained by some noising approaches. [31] proposed the block sparse attention to process longer input and reduced the time complexity to linear. Gidiotis et al. [46] presented a divide-and-conquer strategy that first divides the lengthy article into sections and then summarizes each section, then aggregates all the sub-summaries. Su et al. [47] proposed a two-stage model to generate variable-length summaries using BERT. However, the lack of large-scale financial corpus limits their performances to some extent. Inspired by [28] and [48], we incorporate Transformer into the Seq2Seq model to capture long-range interactions. Then we construct a hierarchical Transformer-BiLSTM encoder to consider both temporal information and interactional relationships of financial news.

C. Graph-Based Summarization

Graph neural networks have been used for summarization due to their powerful capacity to encode crucial relationships between diverse words or sentences. Many studies attempt to enhance the encoder via graph-based methods. Tan et al. [22] proposed the graph-based attention mechanism to obtain attention weights based on sentence importance ratings [42]. Based on [22], Cai et al. [37] further proposed a HITS-based attentional model to consider both word- and sentence-level information. Furthermore, [33] and [34] integrated Gated Graph Neural Networks to enhance the RNN-based encoder to capture long-term semantics. Huang et al. [35] integrated the information from the extracted fact triples into the Seq2Seq model via an additional GAT [39] encoder and trained the model using reinforcement learning. [36] proposed a two-stage model in which a GNN encoder was used to identify important texts, and then a graph-to-sequence model was built to generate informative opinion summaries. However, these models all focus on improving the encoder but few efforts are made to fully utilize the salient history information [40] in the early decoded summary to enhance the RNN-based decoder.

Unlike [35] and [36], we adopt the idea of GAT to construct the Graph Attention-based decoder with history-enhanced attention to capture historical information from the previously decoded sequence, which can generate the summary around the salient history semantics and learn causal relations in financial news.

TABLE I
NOTATIONS OF FREQUENTLY USED VARIABLES IN THIS PAPER

Notations	Description
D	The dataset of financial news summarization
X	The input news article
Y	The summary of the news article
k_m	The number of tokens in a news article
k_n	The number of tokens in a reference summary
k_h	The number of heads in the multi-head attention mechanism
d	The dimension of word embedding
d_e	The dimension of LSTM hidden state
h_j	The j-th hidden state of the BiLSTM encoder
a'_{tj}	Attention weight computed by the history-enhanced attention
$\mathbf{e}_j$	Representation of the j-th token produced by the transformer
$\mathbf{c}'_t$	Vectorized representation of the news article (the context vector)
$\mathcal{G} = (\mathcal{S}, \mathcal{E})$	The graph built at each decoding step
α_{mn}	The graph attention distribution of our decoder
S	$S = \{s_1, s_2, ..., s_{t-1}\}$ is the set of decoder hidden states
S'	$S' = \{s'_1, s'_2, ..., s'_{t-1}\}$ is updated by graph attention network
s'_t	The decoding state updated by the graph attention-based decoder

III. PROPOSED METHOD

In this section, we first define the problem formulation and then describe the model TLGA-FNS in detail. We develop the Transformer-BiLSTM encoder to represent news articles and then propose the history-enhance attention mechanism to obtain the contextual representation. Finally, we develop a novel Graph Attention-based LSTM decoder to generate the summary. The model overview is depicted in Fig. 2.

A. Problem Definition

Given the financial news dataset $D = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_{|D|}, Y_{|D|})\}$, where each sample (X, Y) contains the source text $X = (x_1, x_2, \ldots, x_{k_m})$ and its reference summary $Y = (y_1, y_2, \ldots, y_{k_n})$, where k_m and k_n represent the number of tokens in X and Y ($k_m \gg k_n$), respectively. Given the input sequence X, our goal is to generate the target summary Y. Specifically, we aim to estimate the conditional probability $p(y_t|y_1, \ldots, y_{t-1}, X)$. Due to the independence of the decoding process for each token, the target summary can be generated by maximizing the following probability:

$$p(Y|X;\theta) = \prod_{t=1}^{k_n} p(y_t|\{y_1, y_2, \ldots, y_{t-1}\}, X;\theta) \quad (1)$$

As a result, the objective function f of this task can be formulated as the sum of the negative log likelihoods of target summaries on the training set:

$$\min_{\theta} f = -\sum_{(X,Y)\in D} \log p(Y|X;\theta) \quad (2)$$

where θ represents the model parameters. Table I shows the frequently used variables in this article.

B. Transformer-BiLSTM Encoder

Salient qualitative and quantitative information typically scatters across long-range interactions and sequential contexts of

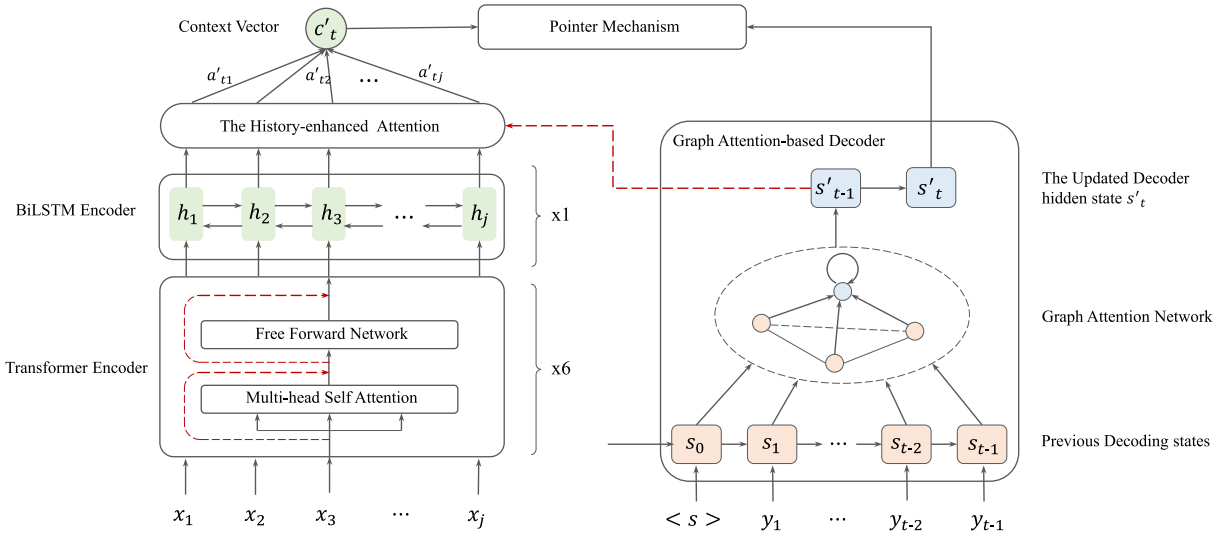

Fig. 2. Overall architecture of our TLGA-FNS model. The encoder is composed of six stacked transformer blocks and a layer of BiLSTM, and the decoder is constructed via Graph Attention Mechanism.

financial news. Previous RNN-based encoders only process the input sequentially and cannot fully consider long-term dependencies. Vanilla Transformer encoders can better model long-range interactions, but they lack the property to distinguish which side the contextual semantics come from [49]. To capture interactional relations while preventing the loss of sequential information, we stack a layer of BiLSTM on a six-layer of Transformer and propose the Transformer-BiLSTM encoder to combine the advantages of both models.

1) Transformer-Based Interactional Relationships Representation: Given the financial news document $X = (x_1, x_2, \ldots, x_{k_m})$ with k_m tokens, we utilize Transformer with self-attention mechanism to capture the long-range interactional relationship among all tokens. We first obtain the vectorized representation of X as $\mathbf{X}=(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k_m})$ by the embedding layer, where $\mathbf{x}_i = \mathbf{E}_{x_i} + \mathbf{P}_{x_i}, i \in [1, k_m]$ and $\mathbf{E}_{x_i}$ represents the $i{-}th$ token embedding. Apart from the token features, the position information of each token also plays an essential role in representing the input news article. Thus, we obtain the position embedding $\mathbf{P}_{x_i}$ via Sinusoidal Position Encoding [28] using sine and cosine functions. Then a stacked Transformer encoder equipped with multi-head self-attention is utilized to obtain inter information as follows:

$$\mathrm{Head}_l = \mathrm{softmax}\left(\frac{\left(\mathbf{Q}\mathbf{W}_l^Q\right)\left(\mathbf{K}\mathbf{W}_l^K\right)^T}{\sqrt{d}}\right)\left(\mathbf{V}\mathbf{W}_l^V\right) \quad (3)$$

where $\mathrm{Head}_l, l \in [1, k_h]$ is the $l{-}th$ head attention output. $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{d \times d_h}$ are query, key and value respectively, which are three linear transformations of the model input $\mathbf{X}$ itself. $\mathbf{W}_l^Q, \mathbf{W}_l^K, \mathbf{W}_l^V \in \mathbb{R}^{d \times d_{\mathrm{head}}}$ are learnable parameters, and $d_{\mathrm{head}} = d/k_h$. Then, all the head attentions Head_l are concatenated to compute the final attention representation as follows:

$$\mathrm{MultiHead} = \mathrm{concat}(\mathrm{Head}_1, \ldots, \mathrm{Head}_{k_h})\mathbf{W}^{\mathrm{O}} \quad (4)$$

where $\mathbf{W}^{\mathrm{O}} \in \mathbb{R}^{d \times d}$ is the parameter of a feed-forward layer.

Since each head pays attention to interactional information at a different distance (i.e., long-term or short-term), we can obtain a global representation of the input new article and learn long-range semantics. Finally, we further utilize layer normalizations and residual connections to obtain the output vector $\mathbf{Trm} = (\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{k_m})$ of the Transformer encoder.

2) BiLSTM-Based Sequential Information Representation: Given the Transformer's output $\mathbf{Trm}$ with the interactional information, we add a layer of BiLSTM to encode the source news article from the right and left sides and learn directionality as well as sequential contextual information, obtaining the final representation sequence. The BiLSTM consists of a forward layer and a backward layer, processing the input sequence $(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{k_m})$ bidirectionally:

$$\vec{h}_j = \mathrm{LSTM}\left(\mathbf{e}_j, \vec{h}_{j-1}\right), j \in [1, k_m] \quad (5)$$

$$\overleftarrow{h}_j = \mathrm{LSTM}\left(\mathbf{e}_j, \overleftarrow{h}_{j-1}\right), j \in [k_m, 1] \quad (6)$$

$$h_j = \left[\vec{h}_j, \overleftarrow{h}_j\right] \quad (7)$$

where $\vec{h}_j$ is the forward hidden state and $\overleftarrow{h}_j$ is the backward state. We further obtain the BiLSTM encoder hidden state h_j by concatenating $\vec{h}_j$ and $\overleftarrow{h}_j$, which contains the sequential contextual information of financial news.

C. Graph Attention-Based Decoder

Different from the common domain, salient causal relations of financial news scatter across redundant information, and content in the early decoded summary can affect the subsequent decoding process. However, the conventional attention-based RNN decoder ignores latent information of the history summary sequence, which cannot learn key causal relationships. Thus,

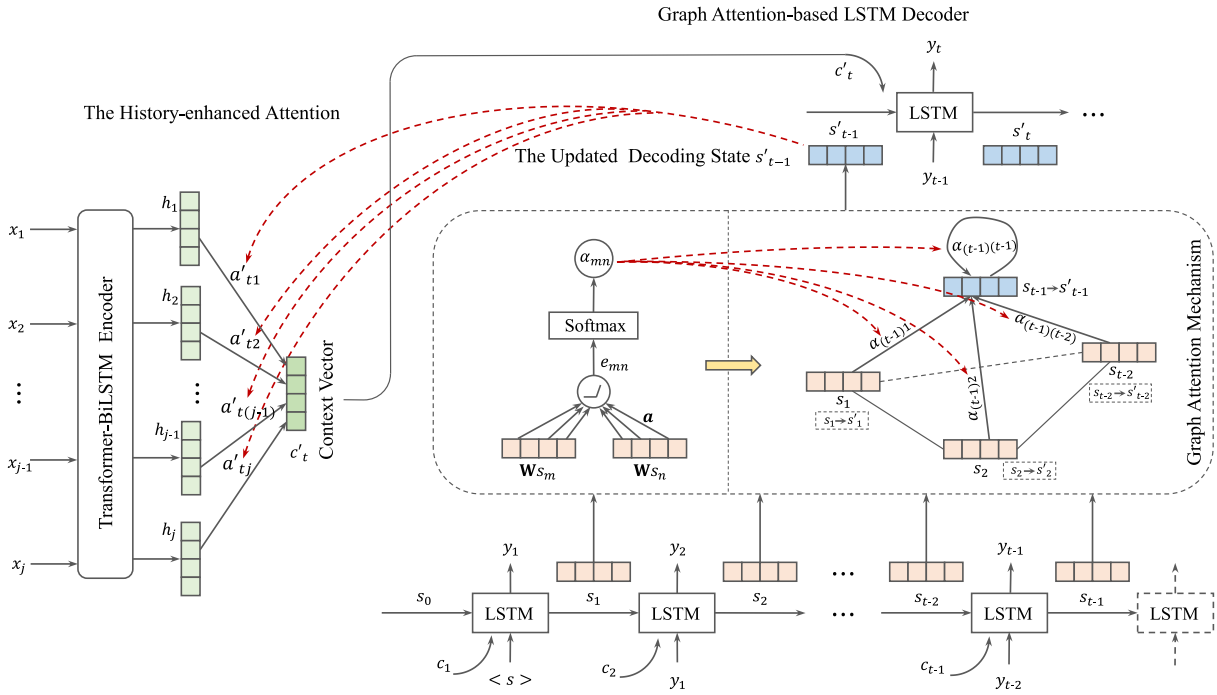

Fig. 3. Architecture of the Graph Attention-based LSTM decoder with the history-enhanced attention mechanism for summarization at the t-th decoding step. Notably, all history decoder hidden states $s_1, \ldots, s_{t-2}, s_{t-1}$ are updated to $s'_1, \ldots, s'_{t-2}, s'_{t-1}$, and we only utilizes the s'_{t-1} to compute the next state s'_t.

we propose the Graph Attention-based decoder with history-enhanced attention to solve this problem, and the architecture of our decoder is shown in Fig. 3.

1) Traditional Attention-Based Decoder: Traditional decoder uses the hidden state s_{t-1} at the last decoding step to represent the whole history information of the summary, and the current decoder's hidden state s_t is computed by:

$$s_t = \text{LSTM}\left(\left[\mathbf{c}_t, y_{t-1}\right], s_{t-1}\right) \tag{8}$$

where y_{t-1} is the last decoded token, and $\mathbf{c}_t$ is document representation computed via traditional attention machanism:

$$e_{tj} = \tanh\left(\mathbf{W}_h h_j + \mathbf{U}_s s_{t-1}\right) \tag{9}$$

$$a_{tj} = \text{softmax}\left(e_{tj}\right)) \tag{10}$$

$$\mathbf{c}_t = \sum_{j=1}^{k_m} a_{tj} h_j \tag{11}$$

where e_{tj} is the cross attention coefficient that indicates the influence magnitude of the j-th token in source text when decoding the t-th token, and a_{tj} is the attention weight.

However, only using s_{t-1} to grasp the entire history semantics of the summary is not adequate for financial news. The reason is that financial articles contain rich causal relationships, and the semantics of early generated tokens can affect the subsequent decoding steps. The conventional decoder only considers the history of the last state, inevitably losing the interactional information of all previous decoder hidden states due to the gate mechanism [41].

2) GAT-Based History Information Representation: Unlike traditional decoders, the GAT-based decoder can directly aggregate the history semantics from all previous decoding states and prevent information loss, attributed to GAT's tremendous capacity for encoding relational information [32], [39]. Specifically, at t-th decoding step, we represent the set of all historical decoder hidden states $\mathcal{S} = \{s_1, s_2, \ldots, s_{t-2}, s_{t-1}\}$ as a complete graph $\mathcal{G} = \langle \mathcal{S}, \mathcal{E} \rangle$, where $s_m \in \mathcal{S}$ is a node and $(s_m, s_n) \in \mathcal{E}$ is an unweighted edge between node s_m and s_n. We further compute the updated hidden states set $\mathcal{S}' = \{s'_1, s'_2, \ldots, s'_{t-2}, s'_{t-1}\}$ through the graph attention mechanism, which updates each hidden state from s_m to s'_m by all its first-order neighbors' features. We use the additive attention operation on the graph to aggregate the history information in all previous hidden states as follows:

$$e_{mn} = \text{LeakyReLU}\left(\mathbf{a}^{\mathrm{T}}\left[\mathbf{W}s_m \| \mathbf{W}s_n\right]\right) \tag{12}$$

$$\alpha_{mn} = \frac{\exp\left(e_{mn}\right)}{\sum_{k=1}^{t-1} \exp\left(e_{mk}\right)} \tag{13}$$

$$s'_m = \text{ELU}\left(\sum\nolimits_{n=1}^{t-1} \alpha_{mn} \mathbf{W} s_n\right) \tag{14}$$

where e_{mn} is the graph attention coefficient that indicates the importance of state s_m to state s_n, $\mathbf{a}^{\mathrm{T}}$ is the weight matrix of a single-layer feed-forward network using LeakyReLU as the nonlinear activation function. $\mathbf{W} \in \mathbb{R}^{d_e \times d_e}$ is a linear transformation, d_e is the dimension of decoder hidden state, and $[\cdot \| \cdot]$ represents the concatenation operation.

Formally, for each hidden state $s_m \in \mathcal{S}$, our decoder first computes the graph attention coefficient e_{mn} between s_m and its neighbor s_n via (12), and then obtains the graph attention weight α_{mn} through the softmax operation. Finally, it updates the new

decoder hidden state s'_m by (14). Since the graph $\mathcal{G} = \langle \mathcal{S}, \mathcal{E} \rangle$ is a complete graph, the new state s'_m is updated based on all the states in S. At t-th steps, although all nodes in $\mathcal{G}$ are updated, we only utilize the updated state s'_{t-1} to compute the next decoding state s'_t, considering the latent history information of all previous decoder hidden states.

3) History-Enhanced Attention-Based LSTM Decoder: After obtaining the updated set $\mathcal{S}' = \{s'_1, s'_2, \ldots, s'_{t-2}, s'_{t-1}\}$ at each step, we further propose a history-enhanced attention-based LSTM to compute the next decoder hidden state s'_t for the summary generation. Different from (8), we compute s'_t via the updated state s'_{t-1} with historical information and the history-enhanced context vector $\mathbf{c}'_t$:

$$s'_t = \text{LSTM}\left([\mathbf{c}'_t, y_{t-1}], s'_{t-1}\right) \tag{15}$$

$$e'_{tj} = \tanh\left(\mathbf{W}_a h_j + \mathbf{U}_a s'_{t-1}\right) \tag{16}$$

$$a'_{tj} = \frac{\exp\left(e'_{tj}\right)}{\sum_{k=1}^{M} \exp\left(e'_{tk}\right)} \tag{17}$$

$$\mathbf{c}'_t = \sum_{j=1}^{M} a'_{tj} h_j \tag{18}$$

Unlike traditional attention mechanisms in [17] and [18], our attention coefficient e'_{tj} is calculated based on each h_j and s'_{t-1} using (16), where s'_{t-1} directly aggregates historical information of all early decoded tokens via GAT. Then we obtain the history-enhanced attention distribution a'_{tj} and new contextual representation $\mathbf{c}'_t$ by (17) and (18). Therefore, our proposed attention can capture the semantics of source articles that are related to previously generated summary sequences and dynamically adjust attention distribution towards input tokens based on historical information. Then, the next output word y_t will be predicted as follows:

$$P_{\text{vocab}} = \text{softmax}\left(\mathbf{W}_{v_1}\left(\mathbf{W}_{v_2}\left[s'_t, \mathbf{c}'_t\right] + b_{v_2}\right) + b_{v_1}\right) \tag{19}$$

Notably, when sequentially predicting the summary with k_n tokens, the above process repeats k_n times to update state s'_t at each t-th iteration and the P_{vocab} is calculated based on the concatenation of s'_{t-1} and $\mathbf{c}'_t$. So the proposed attention can guide the Graph Attention-based decoder to generate a summary surrounding the salient content (such as key causal relations) by reusing the history semantics of early decoded summaries, which helps filter out unnecessary information.

4) Copy Mechanism: For most existing abstractive models, the out-of-vocabulary problem is always detrimental to the readability of the generated summaries. Since financial news contains many professional economic terms and names of listed companies, making it difficult for traditional methods to process unregistered words. Therefore, we employ the copy mechanism to alleviate this problem. Formally, we compute the generation probability as:

$$p_{\text{gen}} = \sigma\left(\mathbf{W}_h \mathbf{c}'_t + \mathbf{W}_s s'_t + \mathbf{W}_y y_{t-1} + b_g\right) \tag{20}$$

where $p_{\text{gen}} \in (0, 1)$ controls whether the next token is copied from the source text based on attention weights or generated from vocabulary according to vocabulary distribution. $\mathbf{W}_h, \mathbf{W}_s, \mathbf{W}_y$, and b_g are learnable parameters, σ is the sigmoid function. Finally, we can obtain the probability of the output token w_t as follows:

$$P_{\text{out}}\left(w_t\right) = p_{\text{gen}} P_{\text{vocab}} + \left(1 - p_{\text{gen}}\right) \sum_{j:w_j = w} a'_{tj} \tag{21}$$

5) Repetition Alleviation Mechanism: Abstractive models are also susceptible to the repetition problem [50] because the traditional attention-based decoder allocates high attention weights to some input content, bringing about the redundancy information to summaries. Therefore, if one word has already appeared, we can adjust its attention weight at subsequent decoding steps to alleviate repetition.

Firstly, our Graph Attention-based decoder can obtain the importance of each token in the early decoded summary and aggregate historical information. The history-enhanced attention can compute an accurate attention distribution based on the history semantics at each decoding step t to avoid repetition. Secondly, we further integrate the coverage mechanism [50] into our decoder to alleviate repetition. The coverage vector c^*_t is defined as the sum of attention distributions of all the previous iterations as follows:

$$c^*_t = \sum_{t'=0}^{t-1} a'_{t'} \tag{22}$$

where c^*_0 is a zero vector at the beginning of decoding, then we use the coverage vector to update the attention coefficients e'_{tj}, so (16) is rewritten as:

$$e'_{tj} = \tanh\left(\mathbf{W}_a h_j + \mathbf{U}_a s'_{t-1} + \mathbf{W}_c c^*_t\right) \tag{23}$$

After that, we define a coverage loss function to penalize repeated words generated at each decoding step t:

$$\mathcal{L}_{\text{cov}} = \lambda \sum\nolimits_j \min\left(a'_{tj}, c^*_{tj}\right) \tag{24}$$

where λ represents coverage loss weight.

D. Model Training

During training, we first train the model without coverage mechanism. We define the loss function as the sum of the negative log-likelihoods of $P_{\text{out}}(w_t)$ as:

$$\mathcal{L}_{\text{sum}} = -\frac{1}{k_n} \sum_{t=0}^{k_n} \log P_{\text{out}}\left(w_t\right) \tag{25}$$

where $P_{\text{out}}(w_t)$ is computed by (20) and (21). After several iterations, we further integrate the coverage loss to obtain the final model. We define the overall loss of the TLGA-FNS as the sum of (24) and (25):

$$\mathcal{L}_{\text{final}} = -\frac{1}{k_n} \sum_{t=0}^{k_n} \log P_{\text{out}}\left(w_t\right) + \lambda \sum_j \min\left(a'_{tj}, c^*_{tj}\right) \tag{26}$$

where k_n is the target summary length, and we train our model with the Adagrad optimizer.

Our model mainly consists of Transformer, LSTM, and Graph Attention Mechanism. The encoder has a time complexity of

TABLE II
STATISTICS OF THE FIRST THREE DATASETS

Datasets	*LCFNS*	*Fin-LCSTS*	*CNN/Daily Mail*
# Samples (trianing)	420,820	207,092	287,227
# Samples (validation)	5000	1000	13,368
# Samples (testing)	5000	1000	11,490
Average article length	112.2	83.9	789.8
Average summary length	19.1	15.8	55.6

$O({k_m}^2 d + k_m d^2)$ due to the multi-head self-attention mechanism, and the time complexity of the history-enhanced attention mechanism is $O(k_m k_n d^2)$, where k_m and k_n are the length of input text X and output summary Y, d is the representation dimension. When decoding, we need a total of k_n graphs with the number of nodes ranging from 1 to k_n to update decoding states, so our decoder's time complexity is $O({k_n}^3 d + k_n d^2)$. Therefore, the total time complexity is $O(({k_m}^2 + {k_n}^3)d + (k_m k_n + k_m + k_n)d^2)$.

IV. EXPERIMENTS

A. Datasets

As far as we know, there is no publicly available corpus for Chinese financial text summarization. To evaluate the performance of TLGA-FNS, we first build a new **L**arge-scale **C**hinese **F**inancial **N**ews **S**ummarization (**LCFNS**) dataset.

LCFNS Dataset: We have crawled 498,209 news articles and their summaries from several major financial portals from January 2013 to June 2020, including East Money, Sina Finance, China Business News, etc. Using the regular expression matching method, we remove extraneous noise from the raw articles and obtain 430,820 article-title pairs. We split them into a training set (420,820), a validation set (5,000), and a testing set (5,000).

Fin-LCSTS Dataset: The LCSTS corpus [51] contains over 2.4 million Chinese microblog-title pairs, with topics ranging from finance to military, politics, and movies. We further compile a list of frequently used financial terms and then manually extract 209,092 examples related to the financial domain from the original LCSTS dataset.

CNN/Daily Mail Dataset: We evaluate TLGA using the CNN/Daily Mail benchmark news-oriented corpus, which contains news on the open domain collected from CNN and Daily Mail portals. We utilize the non-anonymous version of the corpus with lots of named entities. Table II gives the statistics of the first three datasets. Additionally, we evaluate TLGA on the **FNS English dataset**, data statistics and results are shown in Tables XI and XII.

B. Baseline Models

First, we compare our TLGA-FNS model with other strong baselines on the LCFNS and Fin-LCSTS datasets.

RNN-based methods: **RNN** [51] is composed of GRU-based encoder and decoder, and the last encoder hidden state is the input of the decoder, and **RNN Context** [51] applies the attention mechanism. **SRB** [52] computes the cosine similarity of the source text and the generated summary to measure the semantic relevance. **Seq2Seq-Attention** [18] is composed of LSTM-based encoder and decoder with attention mechanism. **PGN+coverage** [21] is equipped with coverage and copy mechanism. **SuperAE** [23] uses an additional autoencoder to enhance the Seq2Seq model and supervises the model via adversarial learning.

Transformer-based methods: **Transformer-GRU-GRU** [48] contains a transformer encoder, a GRU encoder, and a GRU decoder. **Transformer** [28] contains a transformer encoder and a mask-based decoder. **BART** [29] consists of a bidirectional transformer encoder and an auto-regressive decoder, and is pre-trained by some noising approaches. **BigBird-RoBERTa** [31] uses the block sparse attention mechanism to reduce the time complexity to linear. **UniLM** [30] is a unified pre-trained language model (LM) that can achieve unidirectional, bidirectional, and seq2seq modeling. In this part, we fine-tune UniLM-base-Chinese[1] and BART-base-Chinese[2] on the two financial datasets, respectively.

We further compare our model with the following state-of-the-art methods on the CNN/Daily Mail dataset.

RNN-based methods: **SummaRuNNer-abs** [19] uses a two-layer RNN encoder to obtain both word and sentence level semantics. **Graph-based Attention** [22] computes the attention weights based on sentence importance scores inspired by TextRanks. **ML+RL with LM** [53] integrates pre-trained LMs into sequence decoder, which is trained via reinforcement learning. **AOA** [26] proposes Attention Refinement Unit to focus on the salient information, ignoring irrelevant semantics.

Transformer-based methods: In this part, **UniLM**, **BART**, **BigBird-RoBERTa**, and **Transformer** baselines are all large versions with 24 layers. **RoBRETa2RoBRETa** and **BRET2BRET** are two pre-trained models proposed in [54], which warm-starts from the pre-trained checkpoints.

Graph-based methods: **BiLSTM-GNN** [33] incorporates GNNs into the sequence encoder to capture long-range relationships. **ASGARD** [35] uses an additional GAT encoder to represent fact triples to enhance the LSTM encoder. **FASum** [38] uses GAT to represent fact triples and integrates fact-aware information into Transformer.

C. Experimental Settings

1) Data Preprocessing: We cut the summaries and articles written in Chinese into characters instead of words and then remain the characters that appear more than twice, thus reducing the vocabulary size. We use the UNK token to replace the unregistered characters and adopt the PAD token to fill input sequences into the same length. We set the maximum length of the input sequence to 512, and any article that exceeds this threshold will be truncated. The token <s> and </s> indicate the beginning and end of each summary, respectively. Moreover,

[1][Online]. Available: https://github.com/YunwenTechnology/Unilm
[2][Online]. Available: https://github.com/fastnlp/CPT

TABLE III
COMPARISON OF EXPERIMENTAL RESULTS OF VARIOUS MODELS ON THE LCFNS AND FIN-LCSTS DATASETS

Models	*LCFNS Dataset*			*Fin-LCSTS Dataset*		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
RNN	19.83	7.76	17.94	18.31	6.56	17.12
RNN Context	27.63	15.38	25.07	26.18	14.25	24.13
SRB	31.84	19.51	28.87	30.21	18.07	27.61
Seq2seq-Attention	34.01	21.46	30.95	32.29	20.10	29.99
PGN+coverage	35.32	22.62	32.02	33.81	21.45	31.24
SuperAE	37.19	22.38	33.70	35.61	21.22	32.95
Transformer-GRU-GRU	35.34	23.00	32.28	34.07	22.04	31.70
BigBird-RoBERTa	37.10	24.45	34.23	35.49	22.83	33.24
UniLM	38.03	25.29	35.14	36.27	23.64	34.04
BART	38.07	25.43	**35.21**	36.46	23.83	**34.22**
Transformer	37.05	24.23	34.07	35.48	22.61	33.03
TLGA-FNS (Ours)	**38.14**†	**25.53**†	35.18†	**36.69**†	**23.92**†	34.07†

† Represents our TLGA significantly better than the transformer baseline ($p < 0.01$).

we convert the input text into a numerical vector that can be processed by the model based on the vocabulary.

2) Implementation Details: For the model configuration, our primary encoder consists of six stacked transformer blocks, the number of heads is 8, and the dimension of the feed-forward network is 2,048. We set the dimension of LSTM hidden state to 256 and initialize the network's parameters through a uniform distribution. As for the Graph Attention-based decoder, we use a layer of GAT with without multi-head and initialize the parameters of the linear transformation $\mathbf{W}$ and the weights $\mathbf{a}$ by Xavier uniform distribution. On the two Chinese datasets, we take characters as the model input and set the vocabulary size to 6,000. We initialize the word embedding (dimension $d = 512$) randomly by a normal distribution. Additionally, the learning rate and batch size are 0.15 and 16, respectively. As for the English dataset CNN/Daily Mail, we set the vocabulary size and the word embedding to 50,000 and 128, then set the batch size to 16. For all datasets, we adopt the Adagrad optimization algorithm [55] to train our model, then we use beam search when testing, and the beam size is 4. In addition, we limit the maximum length of the decoded summaries to 20 tokens for the two financial datasets and 50 for CNN/Daily Mail. We implement the TLGA-FNS model via PyTorch[3] and train it on a machine equipped with an Intel(R) Xeon(R) CPU E5-2690 v4 and 2 NVIDIA GTX 2080Ti. Our processed data and code will be publicly available at https://github.com/lhz9999/TLGA.

D. Evaluation Metrics

We adopt ROUGE [56] to evaluate the quality of decoded summaries, which evaluates the model by calculating lexical overlap between generated and target summaries. Following previous works [21], we adopt the F1 score of Rouge-1, Rouge-2, and Rouge-L, where Rouge-1 and Rouge-2 are primarily concerned with the informativeness of summaries while Rouge-L is concerned with the readability [25]. Notably, we convert Chinese characters to numerical ids [51] through the vocabulary to compute Rouge scores based on the pyrouge package. We also conduct an extrinsic evaluation to evaluate the causality of summaries via the stock movement prediction task, and we follow [9] and use Accuracy, F1-scores, and MCC as evaluation metrics. Moreover, the BARTScore [57] is utilized to assess the informativeness and faithfulness of summaries based on the BART baseline.

E. Experimental Results and Analysis

1) Results on LCFNS and Fin-LCSTS Datasets: Table III shows our TLGA performs the best among all baselines on the LCFNS testing set, except for the Rouge-L score of BART. Among the non-transformer-based methods, SuperAE achieves the highest Rouge-1 and Rouge-L scores due to the incorporation of an additional auto-encoder. Transformer-based models generally outperform traditional RNN-based methods, except for Transformer-GRU-GRU. The reason is that Transformer-GRU-GRU still uses a traditional RNN decoder instead of a transformer decoder, which ignores key information during summary decoding. UniLM and BART, the SOTA pre-trained language models, outperform all baselines on the LCFNS dataset after fin-tuning, showing their powerful capacity for language generation. However, the lack of large-scale financial domain pre-training data limits BART's performance. Moreover, BART has been pre-trained on 200 GB data with 132.5 M parameters, whereas TLGA is solely trained on the LCFNS dataset with 24.4 M parameters. Nevertheless, TLGA still achieves higher Rouge-1 and Rouge-2 scores, demonstrating its potential as a lightweight solution with promising results.

We also evaluate our model using another financial dataset Fin-LCSTS. Similar trends are observed on Fin-LCSTS, and our TLGA still achieves the best results among all baselines, except for Rouge-L of BART. We find that all these models, including TLGA, achieve lower Rouge scores on Fin-LCSTS compared to LCFNS. The main explanation is that the average article length of the Fin-LCSTS dataset is shorter than that of LCFNS, indicating that our proposed TLGA model performs better on longer articles. Besides, the microblogs in Fin-LCSTS contain

[3][Online]. Available: https://pytorch.org

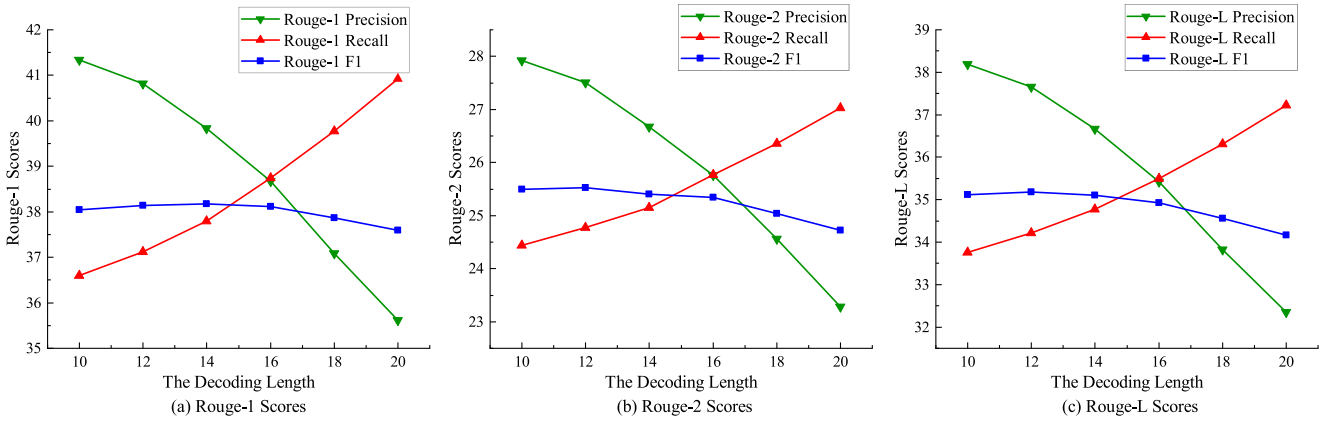

Fig. 4. Results of Rouge-1, Rouge-2, Rouge-L scores with the decoding length from 10 to 20.

TABLE IV
COMPARISON OF EXPERIMENTAL RESULTS OF VARIOUS MODELS ON THE CNN/DAILY MAIL DATASET

Models	ROUGE-1	ROUGE-2	ROUGE-L
SummaRuNNer-abs	37.50	14.50	33.40
Graph-based Attention	38.10	13.90	34.00
PGN+coverage	39.53	17.28	36.38
ML+RL with LM	40.19	17.38	37.52
AOA	40.29	17.76	36.78
BiLSTM-GNN	38.10	16.10	33.20
FASum	38.80	17.23	35.70
ASGARD	40.38	18.40	37.51
UniLM	43.08	20.43	40.34
BART	**44.16**	**21.28**	**40.90**
BERT2BERT	39.02	17.84	36.29
RoBERTa2RoBERTa	40.03	18.57	36.82
BigBird-RoBERTa	39.25	18.46	36.61
Transformer (Our impl.)	39.69	17.17	36.41
TLGA-FNS (Ours)	41.64†	18.52†	38.15†

† Represents TLGA significantly better than the transformer baseline ($p < 0.01$).

more noise, making the summarization more challenging. In this scenario, TLGA still achieves promising performance, demonstrating the effectiveness of our model. Moreover, we conduct significance tests on the two datasets, and the results indicate that TLGA performs significantly better than the Transformer baseline ($p < 0.01$).

2) Results on CNN/Daily Mail Dataset: In Table IV, we further evaluate our model on the benchmark dataset CNN/Daily Mail, which contains longer news articles written in English. Firstly, TLGA outperforms all the RNN-based methods significantly, improving Rouge scores by 1.35, 0.76, and 1.37 compared to AOA. Secondly, the vanilla Transformer baseline yields poor results and even lags behind RNN-based ML+RL with LM and AOA. Pre-trained language models like BERT and RoBERTa perform much better on Rouge-2. However, TLGA still achieves much higher Rouge scores than most transformer-based models, except for UniLM and BART. The reason is that BART has been pre-trained on 160 GB corpus with 400 M parameters for a long time. In comparison, TLGA employs fewer than 31.5 M parameters yet achieves comparable performance to most Transformer-based baselines with far fewer computational resources. Besides, we compare TLGA with three graph-based baselines which utilize GNNs to enhance encoders, and our TLGA significantly outperforms these models. It is because we enhance our decoder by using GAT to aggregate historical information in early decoded tokens and generate summaries around the previous key semantics. Finally, the significance test indicates that TLGA performs significantly better than the Transformer baseline on CNN/Daily Mail ($p < 0.01$).

3) Effect of Decoding Length: It is essential to restrict the length of summaries when decoding, as an appropriate length can enhance performance. In this section, we investigate the impact of decoding length on TLGA's performance on the LCFNS testing set. As shown in Fig. 4, the Rouge precision drops while the Rouge recall increases when increasing the decoding length from 10 to 20, and the Rouge F1 grows initially and later declines. We observe that TLGA gets higher Rouge F1 scores when the decoding length falls within the range of 12 to 16, with the precision and recall scores reaching equilibrium. The main reason is that short decoded summaries may lose salient information, while excessively long summaries may include irrelevant or even fabricated content. Besides, once the number of nodes in Graph $\mathcal{G}$ increases, the time consumption of our Graph-Attention decoder will also increase. From Table II, we know the average length of all reference summaries in LCFNS is 19, and our model achieves the best performance when the decoding length is 12-16, indicating that TLGA can generate shorter summaries without sacrificing the critical information of original articles.

4) Effect of Graph Attention-Based Decoder Versus Transformer Decoder: Graph Attention-based Decoder in our TLGA model can fully exploit history semantics of the output sequence at each step, which is similar to Transformer Decoder equipped with mask technique [32]. However, the attention of these two decoders is calculated in different ways. Compare (3) with (12), Transformer uses dot product operation while our decoder uses the additive attention ((12) can be rewritten as $e_{mn} = \mathrm{LeakyReLU}(\mathbf{a}_1{}^{\mathrm{T}}\mathbf{W}s_m + \mathbf{a}_2{}^{\mathrm{T}}\mathbf{W}s_n)$). Different ways of computing attention can lead to different performance. To verify the performance gap between these two decoders, we

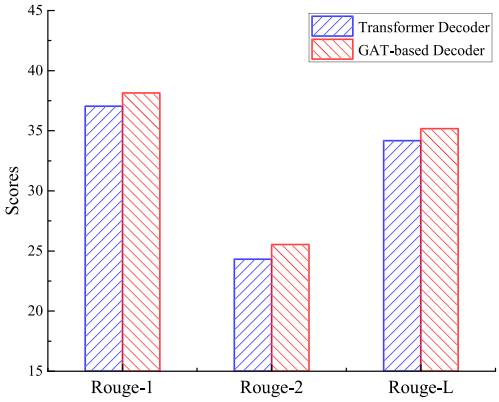

Fig. 5. Comparisons of Graph Attention-based Decoder versus Transformer Decoder.

TABLE V
RESULTS OF ABLATION STUDY IN TERMS OF ROUGE CRITERIA

Models	ROUGE-1	ROUGE-2	ROUGE-L
TLGA-FNS	**38.14**	**25.53**	**35.18**
w/o Trm Encoder	36.78↓1.36	23.91↓1.62	33.46↓1.72
w/o BiLSTM Encoder	37.20↓0.94	24.55↓0.98	34.17↓1.01
w/o GAT Decoder	35.64↓2.50	23.12↓2.41	32.69↓2.49
w/o Copy Mechanism	36.02↓2.12	23.64↓1.89	33.08↓2.10
w/o Coverage	37.37↓0.77	24.75↓0.78	34.53↓0.65

'w/o' means the removal of the module.

design an additional experiment on the LCFNS dataset by replacing our decoder in TLGA with Transformer decoder. As shown in Fig. 5, just using the Transformer decoder leads to performance degradation, indicating the superiority of our decoder.

The main reason is that the ranking of graph attention coefficients in our decoder is unconditioned on the query node, shared among all nodes in the graph [58]. Thus, our proposed decoder can globally identify salient content in the early decoded sequence, then generate the summary surrounding this key content and capture latent causal relations in financial news. As for the Transformer decoder, each Query gets a different attention coefficients ranking of the Keys, so it cannot generate summaries by concentrating on historical information with global importance during the whole decoding process.

5) Ablation Study: To highlight the effectiveness of different modules in our proposed model, we conduct ablation studies on the LCFNS dataset using TLGA, and the results are shown in Table V. We construct five ablation models by removing one of the following components each time: Transformer (Trm) encoder, BiLSTM encoder, Graph Attention-based decoder, copy and coverage mechanism. For a fair comparison, all the ablation models share the same experimental settings.

First, TLGA without (w/o) the Transformer encoder suffers noticeable performance degradation. The reason is that Transformer can better model long-range interactional relations, and removing it can cause the long-term dependence problem. Besides, when removing the BiLSTM encoder, the Rouge scores also decrease, which indicates that sequential semantics are also important. The possible reason is that only using the Sinusoidal Position Encoding in Transformer may be unaware of directionality, while the incorporation of BiLSTM allows for bidirectional processing of inputs and facilitates the learning of sequential contextual information, which can address the lack of directionality problem.

Secondly, without the Graph Attention-based decoder, the Rouge scores decrease significantly by 2.50, 2.41, and 2.49, demonstrating that our proposed decoder plays the most essential role in TLGA. The main reason is that our decoder can reuse history information in the output sequences and dynamically adjust the consideration to different input content, filtering out irrelevant content and capturing salient information. Then, we remove the pointer mechanism, and the performance also degrades significantly. The main reason is that financial news contains named entities, abbreviations, and stock codes, and the copy mechanism can directly retrieve these unregistered tokens and solve the OOV problem. Thus, TLGA's performance is achieved when the GAT-based decoder and pointer network are acting simultaneously. Notably, the Rouge scores only have a minor decline without the coverage mechanism. We speculate that our decoder can also alleviate repetition by adjusting the attention distribution based on history information of previous decoding states.

TABLE VI
RESULTS OF HUMAN EVALUAITON ON INFORMATIVENESS (INFO.), FACTUALITY, FLUENCY, AND RELEVANCY

Models	Info.	Factuality	Fluency	Relevancy
Reference	**4.04**	**4.02**	**4.20**	**4.16**
Transformer	3.26	3.16	3.32	3.36
BART	3.73	3.64	3.82	3.78
TLGA-FNS	3.93†	3.87†	3.88†	3.92†

† Marks the highest score except for reference summaries.

6) Human Evaluation: Apart from Rouge metrics, we further conduct the human evaluation to analyze the quality of summaries generated by TLGA in comparison with two other models (Transformer and BART). We randomly select 100 article-summary samples from the LCFNS test set. For each sample, the source article, golden summary, and generated summaries of different models are provided to 3 annotators who specialize in NLP and financial information analysis. We ask them to manually evaluate each summary based on four factors (informativeness, factuality, fluency, and relevancy) by assigning scores from 1 (worst) to 5 (best). Each model's score is the average of scores given by 3 annotators on each factor, and we also compute Fleiss' kappa-ratio to evaluate the agreement among the annotators. The kappa-ratio results of four factors are all between 0.4 and 0.5. The results are shown in Table VI. We can observe that TLGA outperforms Transformer and BART. Although the Rouge-L F1 score of BART is a little higher in Table III, the superiority is not maintained in the human evaluation. Our proposed TLGA achieves significantly higher scores on all four aspects, especially on informativeness (3.93) and Factuality (3.87). The quality of summaries generated by TLGA closely

TABLE VII
RESULTS OF THE EXTRINSIC EVALUATION USING SUMMARIES DECODED BY VARIOUS SUMMARIZATION MODELS

Models	Accuracy%	F1.Score%	MCC%
PGN+cov	81.41	81.22	63.74
Transformer	84.42	84.88	69.03
BigBird-RoBERTa	84.92	85.29	70.11
BART	85.43	86.38	70.71
TLGA-FNS	**86.93**	**87.61**	**73.81**

TABLE VIII
BARTSCORE METRIC FOR FAITHFULNESS AND INFORMATIVENESS

Models	*Average BARTScore on LCFNS*	
	Faithfulness	Informativeness
PGN+cov	-1.80551	-2.93010
Transformer	-1.50987	-2.52724
BigBird-RoBERTa	-1.47306	-2.48491
UniLM	-1.31087	-2.21739
TLGA-FNS	**-1.26552**	**-2.18192**

resembles the reference summaries, indicating that TLGA can generate more informative and faithful expressions by utilizing history semantics in early decoded tokens.

7) Extrinsic Evaluation for Causal Information: Financial news contains implicit causal relationships, which can affect the fluctuations of stock markets. In this part, we design a domain-specific extrinsic evaluation (predict buying or selling labels) to assess how well our TLGA models the causal relationships and generates informative summaries.

Intuitively, summaries containing more salient causal information are more valuable for making investment decisions (buy or sell). Thus, we conduct a stock movement prediction task to evaluate the causality and informativeness of predicted summaries. Specifically, we have collected 2,137 financial news items about listed companies in the A-share market, each containing an investment rating provided by experienced experts (1,044 for sell and 1,093 for buy). We employ different models to predict summaries of all samples and then only adopt summaries (no extra features) to decide on selling or buying shares of listed companies. Summaries are split into the training (80%), validation (10%), and testing (10%) sets and are represented via BERT. The [CLS] representation is utilized for classification. Following prior works [9], [12], Accuracy, F1-score, and Matthews Correlation Coefficient (MCC) are considered metrics for evaluation. For a fair comparison, we constrain all decoded summaries into the same length.

As shown in Table VII, summaries predicted by TLGA outperform all the baselines. The summaries generated by Transformer and BigBird achieve close results in this task, consistent with the Rouge scores of these two models. When compared to BART, TLGA improves Accuracy by 1.5%, F1 by 1.23%, and MCC by 3.1%, proving that the graph-attention decoder can capture latent causal relations by reusing history information and guide to generate summaries surrounding salient content. Moreover, higher informativeness of the summaries corresponds to higher evaluation scores. Thus, our extrinsic evaluation also demonstrates that TLGA can capture more useful information to help make investment decisions.

8) BARTScore Metric for Faithfulness and Informativeness: In this section, we utilize average BARTScore [57] to assess the faithfulness and informativeness of summaries generated by various models on the LCFNS testing set, which computes the weighted log probability of converting the predicted text to/from a reference output or the source text via the fin-tuned financial BART. According to [57], faithfulness is evaluated by computing BARTScore from the source news to the predicted summary, while informativeness is evaluated by considering both directions between the decoded and reference summary. Higher scores indicate more faithful and informative summaries. For a fair comparison, the summaries generated by BART are not evaluated. Table VIII reveals that TLGA performs the best, UniLM outperforms Transformer and BigBird, and Pointer-Generator achieves the worst performance, indicating that TLGA can generate more informative summaries while ensuring better faithfulness.

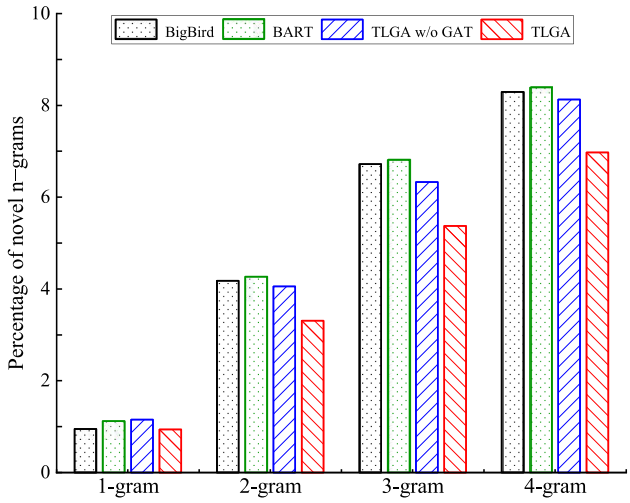

Fig. 6. Percentage of novel n-grams for decoded summaries in the LCFNS testing set.

9) Abstractiveness of Generated Summaries: Inspired by prior works [38], [40], we compute the ratio of novel n-grams that appear in summaries but not in the financial news to explore the abstractiveness of generated summaries on the LCFNS dataset. Fig. 6 shows that BigBird and BART exhibit similar novel n-gram ratios, slightly higher than TLGA. The reason is that financial news contains many named entities, abbreviations, and stock codes, and the pointer network in TLGA can directly copy some entities and unregistered tokens from the source text into the summary. Moreover, comparing the results of TLGA w/o GAT with TLGA, we observe that our Graph Attention-based decoder reduces the abstractiveness of summaries, indicating that our decoder can improve copy performance by retrieving more out-of-vocabulary tokens and effectively handling the OOV problem. This improvement is primarily achieved by

TABLE IX
COMPARISON OF SUMMARIES DECODED BY THREE DIFFENRENT MODELS ON FINANCAIL NEWS ARTICLES

Source text 1 (truncated): ...中石化董事长王玉普与俄罗斯石油公司总裁谢钦签订了《共同开发鲁斯科耶油气田和尤鲁勃切诺-托霍姆油气田合作框架协议》。根据协议，中石化集团有权收购俄罗斯石油公司所属东西伯利亚油气公司和秋明油气公司这两家子公司49%的股份。 Wang Yupu, chairman of Sinopec, and Sechin, President of Rosneft signed the cooperation framework agreement on the joint development of ruskeye oil and gas field and yulubboceno tohom oil and gas field. According to the agreement, Sinopec Group has the right to acquire 49 % shares of the two subsidiaries of Rosneft, East Siberia oil and gas company and Qiuming oil and gas company.	
Reference: 中石化收购俄油两家子公司49%股权	(Sinopec acquired 49% of shares of two subsidiaries of Rosneft)
PGN+cov: 中石化收购俄石油两家子公司49%股权获批	(Sinopec's acquisition of 49% of the shares in two subsidiaries of Rosneft was approved)
UniLM: 中石化与俄罗斯油气公司达成合作协议	(Sinopec and Russian oil and gas company reach cooperation agreement)
BART: 中石化与俄石油公司签订油气田合作框架协议	(Sinopec and Rosneft signed oil and gas field cooperation framework agreement)
TLGA: 中石化收购俄石油两家子公司49%股权	(Sinopec acquired 49% of shares of two subsidiaries of Rosneft)
Source text 2 (truncated): ...据中国经营报报道，继裁员风波之后，为缓解业绩下滑的困局，比亚迪又采取了全员降薪措施。与去年业绩下滑时的裁员对象多为普通员工不同，比亚迪此次降薪举动的范围包括了管理层。比亚迪一位内部员工表示，降薪"十分影响工作积极性。" According to China Business Journal, following the layoff storm, BYD has taken measures to reduce the salaries of all employees in order to ease the dilemma of declining performance. BYD's pay cut covers the management, which is different from the layoffs of ordinary employees when its performance fell last year. An internal BYD employee said the pay cut "has a great impact on work enthusiasm."	
Reference: 比亚迪为缓解业绩下滑采取全员降薪	(To ease the performance decline, BYD reduced the salaries of all employees)
PGN+cov: 比亚迪又采取全员降薪措施与去年不同	(Different from last year, BYD took measures agian to reduce the salary of all employees)
UniLM: 比亚迪全员降薪被指十分影响工作积极性	(Salary cuts for all BYD employees are accused of greatly affecting work enthusiasm)
BART: 比亚迪全员降薪，内部员工称影响工作积极性	(BYD's pay cut for all employees, internal employees said it affects work enthusiasm)
TLGA: 报道称缓解业绩下滑比亚迪又采取全员降薪	(The report said that, to ease the performance decline, BYD reduced the salaries of all employees again.)

The blue words represent salient information in source news and the red words indicate key semantics generated by TLGA but ignored by UniLM and BART.

dynamically adjusting p_{gen} and P_{vocab} based on the historical information from early decoded tokens, as described in (19) and (20).

F. Case Study

For further comparison, we report the output summaries of TLGA, PGN+cov, UniLM, and BART in Table IX. As for source text 1, all four models produce smooth and readable summaries. However, only TLGA captures the essential information (Sinopec acquires 49% shares in two subsidiaries of Rosneft). PGN+cov suffers from factual fabrication by including the phrase 'was approved,' which does not appear in the original text and alters the semantics. Although UniLM and BART achieve high Rouge scores, they all focus on the 'cooperation agreement' rather than the salient detail of '49% shares of two subsidiaries.'

As for the previous example in Fig. 1, all models capture the key information. However, UniLM and BART emphasize the minor point again (pay cut affects work enthusiasm), ignoring the reason behind the salary reduction. TLGA's output summary not only contains salient information 'pay cut' but also identifies the reason for the salary reduction, that is 'performance decline.' The reason is that our graph attention-based decoder successfully captures latent semantics from early generated summaries and learns key causal relations dispersed among the informative content.

G. Attention Visualization

1) Histroy-Enhanced Attention Versus Traditional Attention: Firstly, we compare our history-enhanced attention with the traditional attention mechanism by visualizing their attention weights. As shown in Fig. 7, traditional Seq2Seq attention is relatively scattered, focusing on the minor relation between "pay cut" and "work enthusiasm." In contrast, our proposed attention pays more attention to the reason for salary reduction "ease the performance decline" and produces the summary accurately. The main reason is that our history-enhanced attention can dynamically adjust the attention to the input content based on the history semantics in the early decoded summary and guide our decoder to ignore irrelevant noise and generate the summary surrounding the key content.

2) Graph Attention Visualization: To better demonstrate the effectiveness of our decoder, we visualize the graph attention weights during the whole decoding process of aggregating historical information. As shown in Fig. 8, each row reflects the graph attention weights of all previous hidden states at each decoding step. Notably, the attention mechanism in Fig. 7 is different from that in Fig. 8. The former computes the cross-correlations between the source text and summary while the latter calculates intra-correlations among all the previous decoding states. We can observe that tokens ""缓解"" and ""业绩下滑"" have been weighted higher during the whole decoding process. After decoding "缓解业绩下滑 (ease performance decline)", our decoder concentrates on this salient content and captures the relationship between performance decline and the pay cut. Consequently, our Graph Attention-based decoder can identify key causal relationships in financial news by aggregating history semantics based on all the previous decoding states.

H. Model Parameters and Efficiency of TLGA

We further conduct new experiments on LCFNS to analyze the model parameters and efficiency of TLGA. For comparison, we construct BERT-TLGA, which incorporates BERT into TLGA's Transformer-BiLSTM encoder. The only difference in settings is that the batch size reduces to 8. The results in Table X show

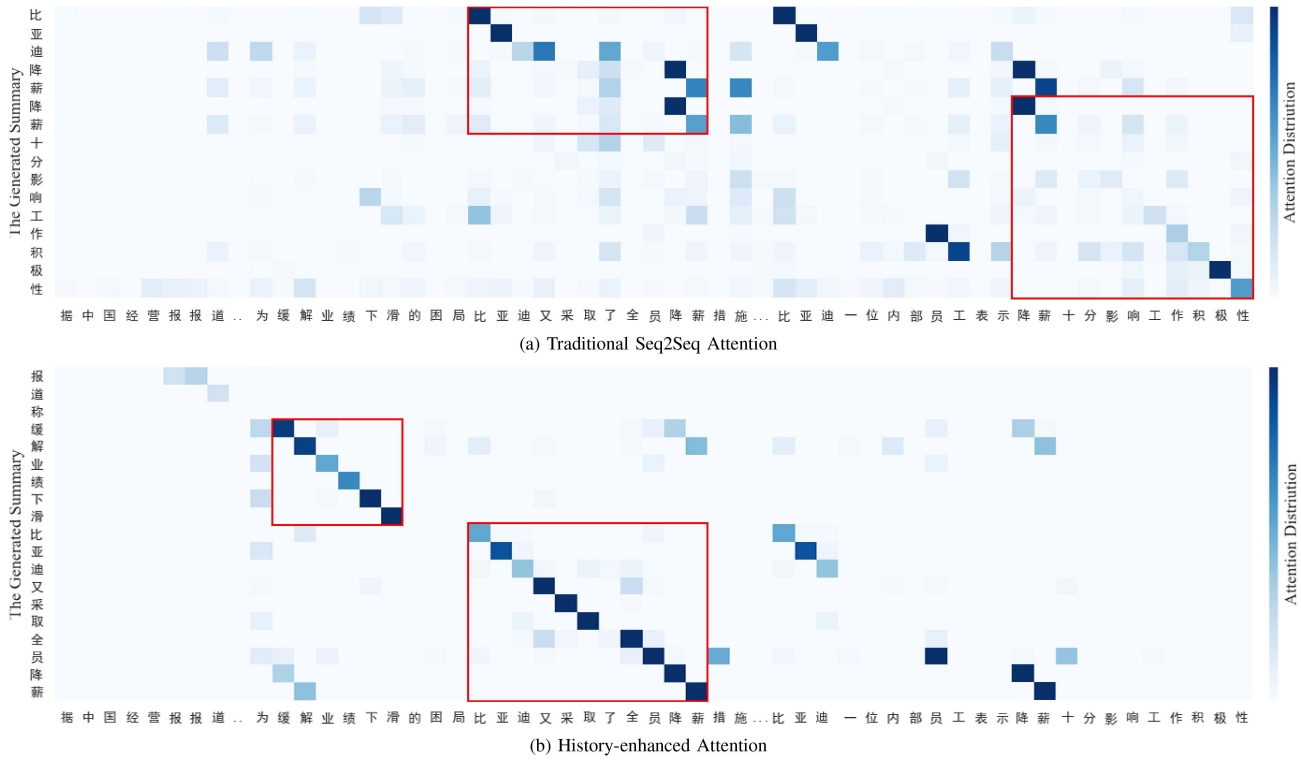

Fig. 7. Comparisons of histroy-enhanced attention versus the traditional attention mechanism. The horizontal axis shows the original text, and the vertical axis shows the generated summary. Darker color indicates higher attention weights.

TABLE X
COMPARISON OF BERT-TLGA AND VANILLA TLGA ON THE LCFNS DATASET

Models	*LCFNS Dataset*					
	ROUGE-1	ROUGE-2	ROUGE-L	Model Params	Training Speed	Predicting Speed
BERT-TLGA	37.86	25.02	34.77	134.61 M	1.38 steps/sec	8.31 samples/sec
Vanilla TLGA	**38.14**	**25.53**	**35.18**	24.38 M‡	**2.70** steps/sec	**11.63** samples/sec

‡ Indicates the minimum model parameters.

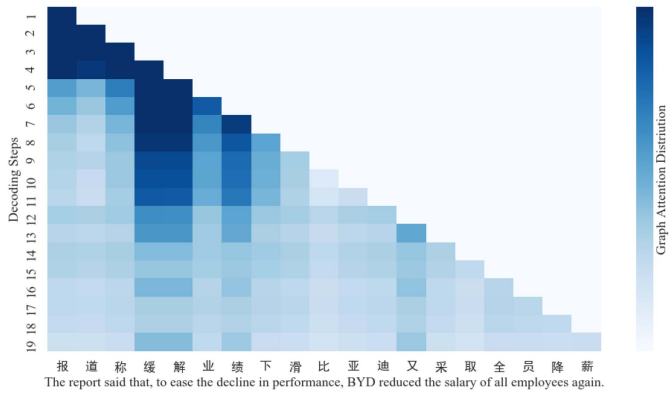

Fig. 8. Visualization of the graph attention distribution at every decoding step. Darker color indicates higher attention weights.

that BERT-TLGA achieves lower Rouge scores. The most likely reason is that common BERT cannot be effectively applied to financial data due to the large differences in vocabulary and expression between the financial corpus and general domain texts. On the other hand, TLGA achieves higher training and predicting speed with only 24.38 M parameters, indicating its greater efficiency. Thus, TLGA can enable investors to obtain salient information promptly by reducing the time required to generate summaries for a large volume of financial news.

I. *Performance of TLGA on the FNS 2022 Shared Task*

We further validate the effectiveness of TLGA on the benchmark English dataset from the FNS 2022 shared task [59], aiming to condense lengthy financial reports (each comprising around 45,000 tokens) into 1,000-word summaries (evaluated using Rouge-2 officially). Table XI provides statistics on the English FNS dataset, where each report contains 3 to 7 golden summaries extracted from the article in a continuous fashion (primarily at the document's beginning). Following the approach in [60], we locate the summary's starting point in the original report and select 4,000 words around it as the new source text for building article-summary pairs. Inspired by [61], we initially pre-train TLGA on the arXiv dataset, then fine-tune it on the

TABLE XI
STATISTICS OF THE FNS 2022 ENGLISH DATASET

Data Type	*FNS 2022 English dataset*		
	Training	validation	Testing
Report full text	3,000	363	500
Gold summaries	9,873	1,250	N/A

TABLE XII
PERFORMANCE OF VARIOUS MODELS ON THE FNS ENGLISH VALIDATION SET IN TERMS OF ROUGE CRITERA

Models	Type	*FNS English validation dataset*		
		ROUGE-1	**ROUGE-2**	ROUGE-L
TextRank	Ext	28.40	7.10	-
TFIDF-SUM-3	Ext	43.30	20.90	37.40
UoBNLP	Ext	48.00	25.00	40.00
MACQUARIE-3	Abs	44.30	30.20	43.20
LSIR-1(mBERT)	Abs	40.59	25.70	38.75
LSIR-1(mT5)	Abs	44.02	30.14	42.36
SSC-AI-RG-1(BART)	Abs	41.70	23.20	-
SSC-AI-RG-1(Top-k)	Ext	**50.80**	**34.50**	**-**
TLGA-FNS (Ours)	Abs	44.53	33.59	**43.61**

Ext indicates that the model is extractive, while Abs indicates that the model is abstractive.

FNS training set, and finally evaluate it on the FNS validation set. Experimental results are shown in Table XII, TLGA outperforms all the abstractive baselines (Longformer-based **MACQUARIE** [61], BART-based **SSC-AI-RG** [62], and **LSIR** [63]), improving Rouge-2 by 3.45 compared with the winning model **LSIR-1(mT5)** in FNS 2022. Although TLGA is abstractive, it can copy salient content into summaries via Pointer Network, outperforming most extractive methods like **TextRank** [42], **TFIDF-SUM** [64], and **UoBNLP** [65] on Rouge-2.

V. CONCLUSION

We propose an enhanced Seq2Seq model TLGA-FNS for financial news summarization. Unlike traditional encoder-decoder models, our Transformer-BiLSTM encoder captures both long-range interactions and sequential information in financial news and alleviates the long-term dependence problem. Furthermore, with the history-enhanced attention mechanism, our graph attention-based LSTM decoder can fully exploit history information in the early decoded summary and generate summaries around the salient content. To address the lack of large-scale high-quality datasets for FNS, we construct a Large-scale Chinese Financial News Summarization (LCFNS) dataset containing 430,820 pairs of news-summary items. Experimental results across four datasets show that our model outperforms other strong baselines. The ablation study indicates that our proposed decoder plays the most significant role in TLGA because it can identify key causal relationships in financial news by aggregating history semantics from all previous decoding states.

REFERENCES

[1] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New avenues in opinion mining and sentiment analysis," *IEEE Intell. Syst.*, vol. 28, no. 2, pp. 15–21, Mar./Apr. 2013.

[2] J. Duan, X. Ding, Y. Zhang, and T. Liu, "TEND: A target-dependent representation learning framework for news document," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 2313–2325, Dec. 2019.

[3] W. Nuij, V. Milea, F. Hogenboom, F. Frasincar, and U. Kaymak, "An automated framework for incorporating news into stock trading strategies," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 4, pp. 823–835, Apr. 2014.

[4] Q. Li, J. Tan, J. Wang, and H. Chen, "A multimodal event-driven LSTM model for stock prediction using online news," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 10, pp. 3323–3337, Oct. 2021.

[5] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *J. Comput. Sci.*, vol. 2, no. 1, pp. 1–8, 2011.

[6] X. Li, H. Xie, L. Chen, J. Wang, and X. Deng, "News impact on stock price return via sentiment analysis," *Knowl. Based Syst.*, vol. 69, pp. 14–23, 2014.

[7] X. Li, P. Wu, and W. Wang, "Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong," *Inf. Process. Manage.*, vol. 57, no. 5, 2020, Art. no. 102212.

[8] K. Filippova, M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Company-oriented extractive summarization of financial news," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2009, pp. 246–254.

[9] Y. Agrawal, V. Anand, S. Arunachalam, and V. Varma, "Hierarchical model for goal guided summarization of annual financial reports," in *Proc. Companion World Wide Web Conf.*, 2021, pp. 247–254.

[10] C. Chootong, T. K. Shih, A. Ochirbat, W. Sommool, and Y.-Y. Zhuang, "An attention enhanced sentence feature network for subtitle extraction and summarization," *Expert Syst. Appl.*, vol. 178, 2021, Art. no. 114946.

[11] X. Li, H. Xie, Y. Song, S. Zhu, Q. Li, and F. L. Wang, "Does summarization help stock prediction? A news impact analysis," *IEEE Intell. Syst.*, vol. 30, no. 3, pp. 26–34, May/Jun. 2015.

[12] Y. Xu and S. B. Cohen, "Stock movement prediction from tweets and historical prices," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1970–1979.

[13] M. El-Haj, A. AbuRa'ed, M. Litvak, N. Pittaras, and G. Giannakopoulos, "The financial narrative summarisation shared task (FNS 2020)," in *Proc. 1st Joint Workshop Financial Narrative Process. MultiLing Financial Summarisation*, 2020, pp. 1–12.

[14] M. Litvak, N. Vanetik, and Z. Puchinsky, "Hierarchical summarization of financial reports with RUNNER," in *Proc. 1st Joint Workshop Financial Narrative Process. MultiLing Financial Summarisation*, 2020, pp. 213–225.

[15] A. Ait Azzi and J. Kang, "Extractive summarization system for annual reports," in *Proc. 1st Joint Workshop Financial Narrative Process. MultiLing Financial Summarisation*, 2020, pp. 143–147.

[16] J. Zhang, Y. Zhou, and C. Zong, "Abstractive cross-language summarization via translation model enhanced predicate argument structure fusing," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 10, pp. 1842–1853, Oct. 2016.

[17] R. Nallapati, B. Zhou, C. Dos Santos, Ç. Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and Beyond," in *Proc. 20th SIGNLL Conf. Comput. Natural Lang. Learn.*, 2016, pp. 280–290.

[18] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 1–15.

[19] R. Nallapati, F. Zhai, and B. Zhou, "SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 3075–3081.

[20] Y. Zhang, E. Chen, and W. Xiao, "Extractive-abstractive summarization with pointer and coverage mechanism," in *Proc. Int. Conf. Big Data Technol.*, 2018, pp. 69–74.

[21] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 1073–1083.

[22] J. Tan, X. Wan, and J. Xiao, "Abstractive document summarization with a graph-based attentional neural model," in *Proc. 55th Annu. Meeting. Assoc. Comput. Linguistics*, 2017, pp. 1171–1181.

[23] S. Ma, X. Sun, J. Lin, and H. Wang, "Autoencoder as assistant supervisor: Improving text representation for Chinese social media text summarization," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 725–731.

[24] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," in *Proc. 6th Int. Conf. Learn. Represent.*, 2018, pp. 1–13.
[25] M. Yang, C. Li, Y. Shen, Q. Wu, Z. Zhao, and X. Chen, "Hierarchical human-like deep neural networks for abstractive text summarization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2744–2757, Jun. 2021.
[26] M. Gui, J. Tian, R. Wang, and Z. Yang, "Attention optimization for abstractive document summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 1222–1228.
[27] J. Li, T. Luong, and D. Jurafsky, "A hierarchical neural autoencoder for paragraphs and documents," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, 2015, pp. 1106–1115.
[28] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[29] M. Lewis et al., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 7871–7880.
[30] L. Dong et al., "Unified language model pre-training for natural language understanding and generation," in *Proc. 33rd Int. Conf. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13063–13075.
[31] M. Zaheer et al., "Big bird: Transformers for longer sequences," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2020, pp. 17283–17297.
[32] G. Wang, R. Ying, J. Huang, and J. Leskovec, "Multi-hop attention graph neural networks," in *Proc. 30th Int. Joint Conf. Artif. Intell.*, 2021, pp. 3089–3096.
[33] P. Fernandes, M. Allamanis, and M. Brockschmidt, "Structured neural summarization," in *Proc. 7th Int. Conf. Learn. Representation*, 2019, pp. 1–18.
[34] Z. Liang, J. Du, Y. Shao, and H. Ji, "Gated graph neural attention networks for abstractive summarization," *Neurocomputing*, vol. 431, pp. 128–136, 2021.
[35] L. Huang, L. Wu, and L. Wang, "Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 5094–5107.
[36] P. Wei, J. Zhao, and W. Mao, "A graph-to-sequence learning framework for summarizing opinionated texts," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 1650–1660, 2021.
[37] X. Cai, K. Shi, Y. Jiang, L. Yang, and S. Liu, "Hits-based attentional neural model for abstractive summarization," *Knowl. Based Syst.*, vol. 222, 2021, Art. no. 106996.
[38] C. Zhu et al., "Enhancing factual consistency of abstractive summarization," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2021, pp. 718–733.
[39] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. 6th Int. Conf. Learn. Representation*, 2018, pp. 1–12.
[40] Q. Liu, L. Chen, Y. Yuan, and H. Wu, "History reuse and bag-of-words loss for long summary generation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 2551–2560, 2021.
[41] Q. Wang and J. Ren, "Summary-aware attention for social media short text abstractive summarization," *Neurocomputing*, vol. 425, pp. 290–299, 2021.
[42] R. Mihalcea and P. Tarau, "TextRank: Bringing order into text," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2004, pp. 404–411.
[43] M. Isonuma, T. Fujino, J. Mori, Y. Matsuo, and I. Sakata, "Extractive summarization using multi-task learning with document classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2101–2110.
[44] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 4171–4186.
[45] K. Yao, L. Zhang, D. Du, T. Luo, L. Tao, and Y. Wu, "Dual encoding for abstractive text summarization," *IEEE Trans. Cybern.*, vol. 50, no. 3, pp. 985–996, Mar. 2020.
[46] A. Gidiotis and G. Tsoumakas, "A divide-and-conquer approach to the summarization of long documents," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 3029–3040, 2020.
[47] M.-H. Su, C.-H. Wu, and H.-T. Cheng, "A two-stage transformer-based approach for variable-length abstractive summarization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2061–2072, 2020.
[48] E. Egonmwan and Y. Chali, "Transformer-based model for single documents neural summarization," in *Proc. 3rd Workshop Neural Gener. Transl.*, 2019, pp. 70–79.
[49] H. Yan, B. Deng, X. Li, and X. Qiu, "TENER: Adapting transformer encoder for named entity recognition," 2019, *arXiv:1911.04474*.
[50] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, "Modeling coverage for neural machine translation," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 76–85.
[51] B. Hu, Q. Chen, and F. Zhu, "LCSTS: A large scale chinese short text summarization dataset," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1967–1972.
[52] S. Ma, X. Sun, J. Xu, H. Wang, W. Li, and Q. Su, "Improving semantic relevance for sequence-to-sequence learning of Chinese social media text summarization," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 635–640.
[53] W. Kryściński, R. Paulus, C. Xiong, and R. Socher, "Improving abstraction in text summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 1808–1817.
[54] S. Rothe, S. Narayan, and A. Severyn, "Leveraging pre-trained checkpoints for sequence generation tasks," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 264–280, 2020.
[55] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, no. 61, pp. 2121–2159, 2011.
[56] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," in *Proc. Hum. Lang. Technol. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2003, pp. 150–157.
[57] W. Yuan, G. Neubig, and P. Liu, "BARTScore: Evaluating generated text as text generation," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2021, pp. 27263–27277.
[58] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?," in *Proc. 10th Int. Conf. Learn. Representation*, 2022, pp. 1–26.
[59] M. El-Haj et al., "The financial narrative summarisation shared task (FNS 2022)," in *Proc. 4th Financial Narrative Process. Workshop*, 2022, pp. 43–52.
[60] M. Orzhenovskii, "T5-long-extract at FNS-2021 shared task," in *Proc. 3rd Financial Narrative Process. Workshop*, 2021, pp. 67–69.
[61] U. Khanna et al., "Transformer-based models for long document summarisation in financial domain," in *Proc. 4th Financial Narrative Process. Workshop*, 2022, pp. 73–78.
[62] N. Shukla, A. Vaid, R. Katikeri, S. Keeriyadath, and M. Raja, "DiMSum: Distributed and multilingual summarization of financial narratives," in *Proc. 4th Financial Narrative Process. Workshop*, 2022, pp. 65–72.
[63] N. Foroutan, A. Romanou, S. Massonnet, R. Lebret, and K. Aberer, "Multilingual text summarization on financial documents," in *Proc. 4th Financial Narrative Process. Workshop*, 2022, pp. 53–58.
[64] S. Krimberg, N. Vanetik, and M. Litvak, "Summarization of financial documents with TF-IDF weighting of multi-word terms," in *Proc. 3rd Financial Narrative Process. Workshop*, 2021, pp. 75–80.
[65] T. Gokhan, P. Smith, and M. Lee, "Extractive financial narrative summarisation using sentencebert based clustering," in *Proc. 3rd Financial Narrative Process. Workshop*, 2021, pp. 94–98.

Haozhou Li is currently working toward the Ph.D. degree in control science and engineering with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, China. His research interests include data mining, text summarization, and sentiment analysis.

Qinke Peng received the B.Sc. degree in mathematics and the M.Sc. and Ph.D. degrees in systems engineering from Xi'an Jiaotong University, Xi'an, China, in 1983, 1986, and 1990, respectively. He is currently a Professor and the Head of the Department of Automation, Xi'an Jiaotong University. His research interests include mining, modeling, and analysis of Big Data, financial informations analysis, and bioinformatics.

Xu Mou received the bachelor's degree in computer science in 2017 from Xi'an Jiaotong University, Xi'an, China, where he is currently working toward the Ph.D. degree in control science and engineering. His main research interests include sentiment analysis, emotion generation, and Big Data analysis.

Ying Wang received the B.S. degree in electrical and computer engineering by the accelerated program and the M.S. degree in system engineering from Xi'an Jiaotong University, Xi'an, China, in 2010 and 2013, respectively. She is currently a Faculty Member with Xi'an Jiaotong University. Her research interests include data mining and algorithm methods.

Zeyuan Zeng is currently working toward the M.S. degree with Xi'an Jiaotong University, Xi'an, China. His research interests include pectral graph theory and data mining.

Muhammad Fiaz Bashir is currently working toward the Ph.D. degree in control science and engineering with Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, China. His research interests include data mining and sentiment analysis.