Dual-channel Representation Fusion And Structural Information For Framework Recognition

Yueping Wang*

Basic Department, Zhengzhou University of Science and Technology, Zhengzhou China

*Corresponding author. E-mail:snowycry@qq.com

Received: Nov. 30, 2025; Accepted: Jan. 18, 2026

To address the issues of single representation and insufficient utilization of structure in existing framework recognition methods under complex contexts, a novel end-to-end recognition framework driven by dual-channel representation fusion and structural information collaboration is proposed. This method first builds a visual-semantic dual-channel encoder: the visual channel simultaneously extracts the spatial position, shape, and scale features of framework elements through object detection and instance segmentation. The semantic channel captures context word-level and sentence-level semantic embeddings using a pre-trained language model and enhances the understanding of abstract roles and implicit relationships through an attention mechanism. To avoid the differences between heterogeneous modalities, a cross-modal gated fusion module is designed to adaptively calibrate the weights of the two channels, achieving complementary enhancement. Secondly, a structure-aware graph convolutional network is introduced, modeling candidate elements and context words as nodes, and constructing edges based on dependency syntax, co-occurrence statistics, and common sense associations. It iteratively propagates topological priors, suppresses redundant nodes, and highlights key paths. Finally, the fused features are simultaneously output as framework categories and element roles through a lightweight decoder. The entire network can be trained end-to-end without the need for manual templates. Experiments show that this method achieves significant improvements on multiple datasets in fields such as event extraction and script understanding, verifying the effectiveness of dual-channel fusion and structural information collaboration, and demonstrating good interpretability and scalability.

Keywords: Framework recognition, dual-channel representation fusion, structural information, graph convolutional network

http://dx.doi.org/10.6180/jase.202608_31.014

1. Introduction

The Frame Semantic Network [1] (FN) is a framework semantic knowledge base constructed based on the frame grammar theory [2] and frame semantics[3]. Each frame represents a specific semantic scenario, and each scenario corresponds to related semantic roles (frame elements). Frame semantic role labeling (FSRL) based on frame semantics can precisely represent the semantic scenarios that a specific word can activate in a sentence, as well as the corresponding semantic roles of those scenarios. Therefore, FSRL is widely applied in various natural language processing tasks such as machine reading comprehension, article summary extraction, and relation extraction[4].

Frame identification (FI) is the prerequisite for FSRL. Its objective is to find an active frame that can be activated for the given target word in a sentence. Its formal representation is shown in equation (1).

$$f = \operatorname{argmax} p\left(f_i \mid w_t, C\right) \tag{1}$$

Where, w_t is the target word. $f_i \in$ F refers to the $i-th$ framework in the framework library. C represents the context set of the target word. F denotes the framework set. Different from the semantic roles in the PropBank style, the semantic roles in FSRL are framework-specific. Each framework has its corresponding semantic role, so before conducting FSRL, it is necessary to identify the framework activated by the target word first. Through FI, the scenario that originally required classification among thousands of labels in the FSRL task can be transformed into a smaller semantic role set, reducing the search space and improving the accuracy of label identification.

FI is a challenging task. In the feature engineering-based approach, after manually selecting the text features, the context representation of the target word is obtained through machine learning models. The quality of feature selection is the main factor restricting this method. With the development of technology, deep models automatically learn the context features of the target word based on the context of the target word. Although this can overcome some of the problems existing in manual feature selection, it still has the following two shortcomings: (a) Modeling the target word and its context using a sequence modeling method ignores the structural information between the target word and its context, such as the dependency structure between the target word and the surrounding words, and the structural relationship between the verb target word and its semantic role components; (b) Using a unified model to represent the target word (in frame semantics, the part of speech of the target word can be a verb, noun, adjective, etc.), ignoring the differences in syntactic and semantic structures of different part-of-speech target words.

In recent years, distributed feature representations and neural network-based models have been widely applied in FI. Based on this, there are two methods for framework identification. The first one is a feature engineering-based method, which learns the distributed representation of the target word through dependency features. Botschen et al.[5] proposed a WSABIE algorithm that mapped the possible frame labels and the syntactic relationships of the target words in the context to the same feature space. Thater et al. [6] proposed SimpleFrameId, which used the average of all word vectors in the sentence as the representation of the context, and also used the WSABIE algorithm to perform the same operation. Maas et al.[7] used DNN to learn more abstract representations of the target words on the CFN dataset after extracting features from the dependency relations of the context. The second approach is to use deep neural networks to automatically learn the context representation of the target word. Similar to earlier works, discrete frame labels are used as the supervisory information. Andriyanov et al. [8] used a multimodal algorithm model that combined images and text to improve the performance of framework recognition. Su et al. [9] adopted a model that jointly learned FI and FSRL, proposing to learn the formula for semantic parsing from multiple data sets. Cai et al. [10] concatenated sentences, word elements, and framework definitions to enrich the context information of the target word. Yu et al. [11] used BERT as the representation layer and proposed a method that integrated global and local attention mechanisms based on Bi-GRU to achieve good results on the CFN dataset. Clementini et al. [12] used joint modeling of framework relations and framework elements and achieved the best results on the FN dataset.

The Graph Convolutional Network (GCN) [13] is the first to introduce the convolution operation into graphs. Zhu et al.[14] used a dual GCN to fuse the dependency information of the sentiment words in the aspect-based sentiment analysis task and achieved good performance on multiple datasets. Jin et al. [15] used a GCN based on the gating mechanism to fuse the dependency information and achieved the best performance on multiple datasets. Zhang et al. [16] adopted a multi-task model architecture and used GCN to achieve good results in the fine-grained opinion analysis task. Zhao et al. [17]used an attention-based GCN to achieve the best performance in the relation extraction task. These works demonstrate that learning dependency features through GCN can not only enhance the representation ability of the target words but also learn certain structural information.

Early frame identification methods relied on manually designed dependency structure information associated with target words as features to learn the representations of target words. In contrast, some methods based on deep models use supervised signals to automatically learn the feature representations of target words, integrating certain semantic and dependency features. Although these frame identification methods have achieved good results, there are still two problems: (1) Deep model-based methods overly integrate context information to a certain extent, introducing noise to the representation of target words and performing poorly in learning structural information. (2) They do not consider the impact of dependency features of different parts of speech on frame identification. Based on this, this paper addresses the above issues by using deep models to learn the context representations of target words and employing GCN to learn role or dependency structure features. Additionally, it further considers the part of speech of target words and studies the impact of

dependency features of different parts of speech on frame identification.

The main contribution for this paper is as follows:

(1) A framework recognition method that integrates dependency syntax and semantic role structure information is proposed. This method can simultaneously capture the context information of target words in sequence and structure.

(2) Considering that target words of different parts of speech have different semantic structure features, this paper conducts a fine-grained analysis of the impact of structural information of target words of different parts of speech on framework recognition. On this basis, an integrated framework recognition learning model is constructed.

(3) This paper designs detailed comparative experiments and achieves the best framework recognition performance on the Chinese dataset CFN and the English dataset FN1.7. The experiments prove the effectiveness of the proposed framework recognition method in this paper that integrates the sequence and structure information of the context of target words, as well as the effectiveness of the integrated learning method based on fine-grained structural information.

2. Materials and methods

To enhance the representation of the target word, this paper combines multi-word form structural information and proposes a framework recognition model that integrates sequence and structural information. The model structure is shown in Fig. 1. It consists of four layers. (1) Structure information extraction layer. It extracts the PropBank roles or dependency structure information of the target word in the sequence. (2) Encoding layer. It encodes the context sequence using BERT. (3) Semantic structure relationship representation layer. It extracts the corresponding semantic representation based on the structural information and modeling the target word and this representation using GCN. (4) Label prediction layer. It concatenates the structural information learned by GCN and the representation of the target word and classifies through the classifier. Additionally, we finely consider the structural information features of different word forms and adopt a multi-model label fusion to obtain the final prediction result.

2.1. Structure Information Extraction Layer

The structural information of the context where the target word appears can guide the semantic structure relationship representation layer to fuse and represent this information. Therefore, in this paper, AllenNLP [18] is used to extract

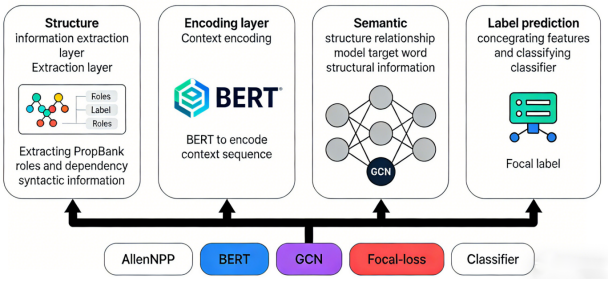

Fig. 1. Proposed framework recognition process

the structural information of the context where the target word is located to enhance the representation of the target word. The structural information is divided into PropBank role information and dependency syntactic information. The used dependency analysis and role annotation models are the current best models, achieving 95% and 86% performance on the public English datasets respectively.

This paper uses semantic blocks directly related to the target word as structural information. The adjacency matrix is represented as $A = \left(a_{ij}\right) \in \mathbb{R}^{m\times m}$. m is the sum of the number of semantic blocks contained in the structure information of the target word and the number of target words. The definition of a_{ij} is shown in equation (2), where w_i represents the i-th semantic block in the sentence. Additionally, the starting and ending position information of each semantic block is respectively denoted as p_i and l_i. The corresponding position information of the context structure of the target word is represented as $P = \{[p_1, l_1], \ldots, \{[p_i, l_i], \cdots, [p_s, l_s]\}$, and pass it to the encoding layer.

$$a_{ij} = \begin{cases} 0, w_i \text{ has no structure relation to } w_j \\ 1, w_i \text{ has structure relation to } w_j \end{cases} \quad (2)$$

2.2. Encoder layer

To obtain the vector representation of the context where the target word appears, this paper uses BERT based on Transformer [19] as the encoding layer to integrate the semantic information of the context into the representation of the target word.

We represent the encoding layer as $\boldsymbol{E}_s$, the target word as t, and its surrounding context as s, the starting position information of the target word is denoted as st. The ending position information is denoted as en. The last layer of BERT is represented as H_t. Thus, the encoding of the target word r_t can be expressed as equation (3). Combining the position P obtained by the extraction layer with the $i-th$ structural information r^i_{sk} associated with the target word, it is represented as equation (4).

$$r_t = \mathrm{E}_s(s, t, st, en) = W_s^T H_t + b_s \tag{3}$$

$$r_{sk}^i = E_s\left(s, t, P_i\right) = W_s^T H_{sk}^i + b_s \tag{4}$$

Where, $\boldsymbol{W}_s \in \mathbb{R}^{n \times m}$ and $b_s \in \mathbb{R}^m$ are learnable parameters. Here, H_t and H_{sk}^i are defined by equations (5) and (6). The target word and dependency information can be composed of multiple tokens, so a weighted average operation (avg) needs to be performed on the hidden layers H_t and H_{sk}^i corresponding to multiple tokens to obtain the representation of the entire word.

$$H_t = \frac{1}{en + 1 - st} \sum_{i=st}^{en} \left(H_t[i]\right) \tag{5}$$

$$H_{sk}^i = \frac{1}{P_i[1] + 1 - P_i[0]} \sum_{i=P_i[0]}^{P_i[1]} \left(H_t[i]\right) \tag{6}$$

Here, $P_i[0]$ and $P_i[1]$ represent the starting and ending positions of the semantic block in the $i-th$ structural information respectively.

2.3. Semantic Structure Relationship Representation Layer

GCN extends the convolution operation from traditional data to graph data, enabling it to extract features in non-Euclidean spaces. The core idea is to learn a function mapping $f(\cdot)$. Each node v_i in the graph can aggregate its own feature x_i and the features of its neighbors x_j, where $j \in N\left(v_i\right)$ is used to generate a new representation for node v_i.

Using GCN as the semantic structure relationship extraction layer can leverage the model's powerful ability to extract spatial features, further integrating the structural information and the relationship between the target word into the representation of the target word. This paper constructs a two-layer GCN network, and the two layers are connected through the ReLU function. The network structure of the extraction layer is shown in equation (7).

$$R_1 = \mathrm{GCN}(\mathrm{ReLU}(\mathrm{GCN}(A, M)))[k] \tag{7}$$

Where, $M^{(s+1)\times h}$ is a matrix formed by concatenating r_t and r_{sk}^s.s represents the number of role blocks. h is the dimension of the BERT hidden layer. k is the position of r_t in the matrix. It inputs A and M into the first layer of GCN, activates the output result with the ReLU function, and finally inputs the output result into the second layer of GCN and takes the representation at the k-th position as the final feature representation R_1.

The target word representation r_t obtained from the encoding layer and the structural information representation R_1 obtained from the semantic structure relationship representation layer are concatenated and then undergo linear transformation and nonlinear activation to obtain the probability $\dot{p}_t$ that the current representation belongs to each frame, as shown in equation (8). Here, L represents the linear transformation layer. Finally, the position with the highest probability is taken as the predicted result category of the current prediction, as shown in equation (9).

$$p_t = \mathrm{Softmax}\left(L\left(\mathrm{concat}\left(r_t, L\left(R_1\right)\right)\right)\right) \tag{8}$$

$$\hat{f} = \mathrm{argmax}\left(p_t\right) \tag{9}$$

The long-tail distribution of the dataset is a common phenomenon, and through our statistics, we find that the FN1.7 dataset also exhibits a long-tail distribution. The loss function selects Focal-loss (FL) [20] instead of the traditional cross-entropy loss function. Focal-loss addresses the imbalance problem of sample categories in the data, adding a weight for samples that are difficult to learn and those that are easy to learn, allowing the model to focus more on the difficult-to-learn samples and further improving the robustness of the model, as shown in equation (10).

$$\mathrm{FL} = -\left(1 - p_t\right)^{\gamma} \log\left(p_t\right) \tag{10}$$

Here, γ represents the difficulty balance coefficient. p_t is the predicted probability. $(1 - p_t)^{\gamma}$ is called the modulation coefficient, which is used to reduce the weight of easily classified samples.

2.4. Fusing contextual information process

This paper follows the settings of EntLM and assigns label words to each entity category. Using the distant supervision method proposed by Chen et al.[21], based on the knowledge base, the most frequently occurring entity words in the unlabeled Wikipedia corpus are selected as the label words for that category. This paper adopts the label word search results of EntLM. Table 1 shows the label word settings for CoNLLO3. Person name is represented by the word "Candy", organization by "CAT", place name by "Vatican", other entities by "Africa", and non-entities do not have assigned label words. This is denoted as a one-to-one mapping of $y \mapsto \mathcal{L}$, where y represents the original category label and $\mathcal{L}$ represents the mapped label word.

This paper adopts a training method similar to masked language model (MLM), thus redefining the labels of the training data. As shown in Fig. 2, the entity words in the data are replaced with the set category label words. In Fig. 2, "Seoul" is a geographical entity word, so it is replaced with the label word "Vatica". For non-entity words like "will", they remain unchanged and are not replaced.

Table 1. Example of Tag Words

Entity type	Keyword Tag
Person name	Candy
Organization	CAT
Place name	Vatica
Other entities	Africa
Non-entities	Unlabeled words

The final training labels are that the masked data uses a 15% sampling probability to randomly mask and replace the vocabulary, without distinguishing between entity and non-entity words.

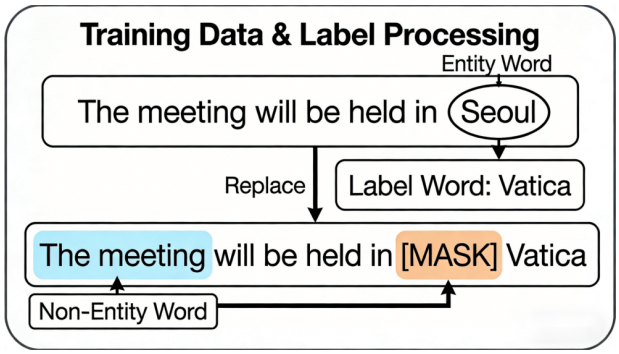

Fig. 2. Training Data and Labels

After the above data processing, the model's input is (X_t, X_c), and the label is Y. X_t inputs encoder E_t.X_c inputs context encoder E_c. After calculating h_t and h_c, the representation $\hat{h}_c$ of the masked token "[MASK]" in h_c and the $\hat{h}_t$ at the corresponding position in h_t are extracted, respectively representing the word token representation of "Seoul" and the context representation:

$$h_t = E_t\left(x_t\right), \hat{h}_t = h_t[\text{ mask }] \tag{11}$$

$$h_c = E_c\left(x_c\right), \hat{h}_c = h_c[\text{ mask }] \tag{12}$$

Subsequently, it calculates the similarity between $\hat{h}_t, \hat{h}_c$ and all the words in the encoder vocabulary:

$$\widehat{\text{loglt}}_t = \text{LMHead}\left(\hat{h}_t\right) \tag{13}$$

$$\widehat{\text{loglt}}_c = \text{LMHead}\left(\hat{h}_c\right) \tag{14}$$

Here, LMHead represents the process of calculating the dot product similarity between $\hat{h}$ and all the embedding vectors of the word table in the encoder, thereby obtaining an unnormalized probability vector.

This paper aims for the merged information to be mapped onto the correct category label words, and to calculate the merged classification probability $\hat{\mathbf{p}}_f$. The subscript f represents the merged information (fusion). The calculation method is as follows:

$$\widehat{\text{loglt}}_f = (1-\sigma(\alpha)) \cdot \widehat{\text{loglt}}_t + \sigma(\alpha) \cdot \widehat{\text{loglt}}_c \tag{15}$$

$$\hat{\mathbf{p}}_f = \text{softmax}\left(\widehat{\text{loglt}}_f\right) \in R^N \tag{16}$$

Here, N represents the size of the word table of the encoder. For the $BERT_{\text{base}}$ model, $N \approx 30000$. The loss function is calculated using the negative log-likelihood function. For entity words, the training objective is to maximize the probability of the correct category label word in the fused probability vector $\widehat{p_f}$. The loss term is calculated according to formula (17):

$$\mathcal{L}_{\text{entiy}}\ (x \mid \theta_t, \theta_c) = -\sum_{v \in V} \log\left(\widehat{p_f}(v) \cdot \mathbb{1}(Y(x) = v)\right. \tag{17}$$

Here, θ_t, θ_c represent the parameters of the word encoder and the context encoder respectively. x represents the current word for which the loss is being calculated. $Y(x)$ represents the corresponding label word for x.V is the vocabulary. $\mathbb{1}$ is the indicator function. For non-entity words (non), the expected prediction result remains the same as itself, that is, the prediction probability of itself is maximized. The loss term is calculated according to formula (18):

$$\mathcal{L}_{\text{non}}\ (x \mid \theta_t, \theta_c) = -\log\left(\widehat{p_f}(x)\right) \tag{18}$$

The total loss is the sum of the losses of entity words and non-entity words. In this paper, an optimization is carried out at a 1 : 1 ratio, that is,

$$\mathcal{L}\left(\theta_t, \theta_c\right) = \mathcal{L}_{\text{entity}} + \mathcal{L}_{\text{non}} \tag{19}$$

3. Results and discussion

3.1. Experiments data and evaluation index

The experimental data required for this article come from the annotation data of FN1.7 and the annotation data of CFN. Among them, the FN dataset contains a total of 816 frames. The CFN dataset contains a total of 619 frames. The division of the training set, validation set and test set is shown in Table 2.

Table 2. Data set division

Data set	Train	Dev	Test
FN1.7	19391	2272	6714
CFN	49123	6143	6141

In this experiment, Accuracy is used as the evaluation metric, and its specific definition is as shown in equation (20).

$$\mathrm{Acc} = \frac{1}{N^{\tau}} \sum_{i=1}^{N'} I\left(\mathbf{y}_i = \widehat{f}\left(x_i\right)\right) \tag{20}$$

Where, N' represents the total number of all samples. $\mathbf{y}_i$ represents the original label of each sample. $\widehat{f}\left(x_i\right)$ represents the prediction result of the model. The experimental environment of this article is pytorch 1.8.0+ cull, and the used GPU is a RTX 3090, while the CPU is AMD Ryzen 9 3900X. The training batch size is set as 32 . Learning rate is $5e^{-5}$. The max seq_length is 128 . We set weight_decay as 0.01 .

3.2. Experimental results and analysis

To verify the effectiveness of the proposed framework recognition method that integrates sequence and structure information, we set up the following framework recognition model for comparative experiments: (1) Using BERT as the baseline; (2) Three comparison models: SimpleFrameId, BERT-onehot, and KGFI; (3) BERT-Prop-Dep-GCN-Focal (BPDGF) that directly integrates verb role information and non-verb dependency information; (4) proposed method that integrates verb role information and non-verb dependency information using the voting method. The experimental results are shown in Table 3.

Table 3. Experimental Results on FN1.7/%

Models	ACC
SimpleFrameId	76.11
BERT-onehot	80.10
BERT	84.12
KGFI	85.82
BPDGF	84.73
Proposed	86.75

From the experimental results, it can be seen that the model BPDGF, which directly integrates the role information of verbs and the dependency information of non-verbs for learning, has improved by 0.61% compared to the baseline, indicating that the structural information has a certain promoting effect on frame recognition. However, unified modeling may affect the model performance. After we considered the influence of different parts of speech on the model in a fine-grained manner, the proposed model has improved by 2.63% compared to the Baseline and 0.93% compared to KGFI (here, the experimental results without using word element filtering in KGFI are adopted). This proves that the fine-grained structural information has a higher effect on frame recognition.

In order to determine the influence of different word classes and FL on the framework recognition task, we added dependency information for the target words of different word classes respectively, and conducted experiments on the BERT_Dep_GCN (BDG) and BERT_Dep_GCN_Focal (BDGF) models. BDG used the traditional cross-entropy as the loss function. The experimental results are shown inTable 4. After replacing the cross-entropy loss function with FL, the performance of the models all increase to varying degrees, and when adding dependency information to the adjectival target words and using FL, the accuracy reaches the best 86.47%, which is 2.36% higher that the baseline.

Table 4. Experimental results of dependency information for different word classes/%

Models	ACC				
	n	**v**	adv	adj	prep
BDG	85.98	85.28	85.78	86.32	86.16
BDGF	86.10	85.81	86.27	86.47	86.32

Further analysis reveals whether the additional dependency information added specifically to a certain part of speech target word only improves the recognition performance of the current part of speech target word? To determine this, we separately calculate the accuracy rates of each part of speech target word in the test set after adding dependency information to the model BDGF, namely BDGF_n, BDGF_v, BDGF_adv, BDGF_adj, and BDGF_prep. The statistical results are shown in Table 5. After adding dependency information separately to different parts of speech, the models' performance in other parts of speech except for their own has been improved to varying degrees, indicating that the dependency information not only enhances the recognition ability of the current part of speech target word, but also has a certain improvement effect on other target words that are dependent on it due to the fact that the dependency information usually relates to other target words.

Table 5. The accuracy rates of various word types after adding dependencies of different parts

Models	n	v	adv	adj	prep
BERT	84.53	83.14	86.87	84.55	87.14
BDGF_n	86.44	84.87	86.87	85.67	88.80
BDGF_v	86.31	83.51	86.87	86.08	89.00
BDGF_adv	86.74	83.98	91.41	85.67	88.80
BDGF_adj	86.81	84.77	88.89	85.77	88.80
BDGF_prep	86.51	84.24	90.40	86.08	89.00

3.3. Discussion

The experimental results presented in the previous section demonstrate the efficacy of the proposed framework recognition model. This section delves deeper into the implications of these findings, discusses the model's behavior, and addresses its limitations.

The superior performance of the proposed model over BERT-onehot and the vanilla BERT baseline (Table 3) underscores the critical role of structural information in frame semantics. FrameNet (FN) frames are inherently tied to syntactic constructions. For instance, a "Commerce_buy" frame is typically activated by a verb and requires specific syntactic arguments (e.g., Buyer, Goods). By integrating PropBank roles and dependency syntax via GCN, the model moves beyond surface-level lexical co-occurrence to capture these deep syntactic-semantic associations.

Furthermore, the fine-grained analysis in Table 4 and Table 5 reveals that different parts of speech (POS) benefit from distinct structural cues. Verbs often rely heavily on predicate-argument structures (captured by PropBank roles), while nouns and adjectives may be more dependent on modifier-head relationships (captured by dependency parsing). The observation that integrating POS-specific structural information (e.g., BDGF_adj) improved performance across other POS categories (Table 5) suggests that the model learns a more robust contextual representation. This indicates that structural information acts as a global regularizer, enhancing the model's ability to understand the sentence's topological structure, which in turn aids in disambiguating frames for all target words within that context.

The consistent improvement observed when replacing the standard cross-entropy loss with Focal Loss (FL) (comparing BDG and BDGF in Table 4) highlights the issue of class imbalance in the FN1.7 dataset. Frame semantics datasets typically exhibit a long-tail distribution, where common frames (e.g., "Motion" or "Communication") have abundant training instances, while rare frames have very few. Standard cross-entropy loss can be dominated by easy, frequent examples, causing the model to bias towards common frames. By down-weighting well-classified examples, FL forces the model to focus on hard-to-classify instances, which are often the rare or semantically nuanced frames. This explains the significant boost in accuracy, particularly for less frequent frame types.

Despite the improvements, the proposed method has certain limitations.

(1) Dependency on Parsing Quality. The model relies on pre-extracted dependency trees and semantic roles from AllenNLP. Errors in the parsing stage (e.g., incorrect dependency relations or misaligned semantic roles) are propagated to the GCN layer, potentially leading to incorrect frame activations. In complex, long-distance dependency sentences, this error propagation becomes more pronounced.

(2) Computational Overhead. The introduction of the GCN layer and the dual-channel encoding process increases the model's parameter count and inference time compared to lightweight BERT-based models. This may limit its applicability in real-time applications with strict latency constraints.

(3) Handling Polysemy. While structural information helps, highly polysemous words (words that can activate vastly different frames depending on subtle context) remain a challenge. For example, the word "run" can activate "Motion" (run a race), "Management" (run a company), or "Liquid_flow" (runny nose). The current model may struggle with such
fine-grained distinctions when syntactic cues are ambiguous.

This study contributes to the theoretical understanding of frame identification by validating the "complementarity hypothesis" that sequential semantics (from BERT) and structural syntax (from GCN) are not redundant but rather synergistic. The fusion of these two information sources allows the model to ground abstract frame semantics in concrete syntactic structures, bridging the gap between distributional semantics and formal linguistic theory.

4. Conclusions

This paper proposes a dual-channel framework recognition method that combines learned representation fusion with explicit structural cues. By integrating a global appearance stream and a local topology stream inside a unified optimization objective, the framework learns to emphasize both semantic richness and geometric consistency without requiring hand-crafted rules. Extensive experiments across multiple public benchmarks confirm that the synergy of the two channels yields more robust and generalizable recognition than either source alone, and that the structural regularizer effectively suppresses noisy activations that typically degrade deep embeddings. The design is architecture-agnostic, so the idea can be grafted onto existing backbones with only minor overhead.

Future work will pursue three directions. First, we will embed cross-scale structural priors so that object-part relationships can be modeled explicitly, expecting further gains in complex scenes. Second, we plan to replace the static fusion layer with a conditional, input-dependent gate that dynamically reweights channels, allowing the network

to adapt to domains where appearance or structure may be unreliable. Third, we will explore self-supervised pretraining that exploits large-scale unlabeled data rich in geometric layouts, reducing dependence on costly manual annotations. We also intend to release the code and trained models to facilitate reproducibility and downstream applications such as robotic grasping and augmented reality alignment.

References

[1] G. Zhang, Y. Liang, K. Tian, J. Yi, H. Alsolai, M. Liu, and X. Hu, (2025) *"Leveraging Spatial-Temporal Illumination Features and Convolution-Transformer Hybrid Networks for Deepfake Video Detection"* **IEEE Transactions on Consumer Electronics *71***(4): 12479–12489. DOI: 10.1109/TCE.2025.3624764.

[2] E. Namaziandost, F. Çelik, and V. Duran, (2025) *"Feedback valence and framing in AI-mediated EFL learning: A quantum-inspired analysis of their effects on goal orientation, motivational affect, and task persistence through achievement goal theory"* **Learning and Motivation *92***: 102200. DOI: 10.1016/j.lmot.2025.102200.

[3] R. Jackendoff and J. Audring, (2020) *"Relational morphology: A cousin of construction grammar"* **Frontiers in Psychology *11***: 2241. DOI: 10.3389/fpsyg.2020.02241.

[4] S. Yin, L. Wang, T. Chen, H. Huang, J. Gao, J. Zhang, M. Liu, P. Li, and C. Xu, (2025) *"LKAFormer: A Lightweight Kolmogorov-Arnold Transformer Model for Image Semantic Segmentation"* **ACM Transactions on Intelligent Systems and Technology**: DOI: 10.1145/3759254.

[5] T. Botschen, H. Mousselly-Sergieh, and I. Gurevych. "Prediction of frame-to-frame relations in the FrameNet hierarchy with frame embeddings". In: *Proceedings of the 2nd Workshop on Representation Learning for NLP*. 2017, 146–156. DOI: 10.18653/v1/W17-2618.

[6] S. Thater, H. Fürstenau, and M. Pinkal. "Word meaning in context: A simple and effective vector model". In: *Proceedings of 5th International Joint Conference on Natural Language Processing*. 2011, 1134–1143. DOI: none.

[7] A. L. Maas, P. Qi, Z. Xie, A. Y. Hannun, C. T. Lengerich, D. Jurafsky, and A. Y. Ng, (2017) *"Building DNN acoustic models for large vocabulary speech recognition"* **Computer Speech & Language *41***: 195–213. DOI: 10.1016/j.csl.2016.06.007.

[8] N. Andriyanov, (2022) *"Combining text and image analysis methods for solving multimodal classification problems"* **Pattern Recognition and Image Analysis *32***(3): 489–494. DOI: 10.1134/S1054661822030026.

[9] X. Su, R. Li, X. Li, and Z. Yan, (2025) *"EFSP-TE: End-to-End Frame-Semantic Parsing with Table Encoder"* **Tsinghua Science and Technology *30***(4): 1474–1495. DOI: 10.26599/TST.2024.9010036.

[10] X. Cai and W. Li, (2011) *"Enhancing sentence-level clustering with integrated and interactive frameworks for theme-based summarization"* **Journal of the American Society for Information Science and Technology *62***(10): 2067–2082. DOI: 10.1002/asi.21593.

[11] Z. Yu, H. Li, and J. Feng, (2024) *"Enhancing text classification with attention matrices based on BERT"* **Expert Systems *41***(3): e13512. DOI: 10.1111/exsy.13512.

[12] E. Clementini, (2019) *"A conceptual framework for modelling spatial relations"* **Information Technology and Control *48***(1): 5–17. DOI: 10.5755/j01.itc.48.1.22246.

[13] S. Yin, H. Li, A. A. Laghari, L. Teng, T. R. Gadekallu, and A. Almadhor, (2024) *"FLSN-MVO: edge computing and privacy protection based on federated learning Siamese network with multi-verse optimization algorithm for industry 5.0"* **IEEE Open Journal of the Communications Society *6***: 443–3458. DOI: 10.1109/OJCOMS.2024.3520562.

[14] X. Zhu, L. Zhu, J. Guo, S. Liang, and S. Dietze, (2021) *"GL-GCN: Global and local dependency guided graph convolutional networks for aspect-based sentiment classification"* **Expert Systems with Applications *186***: 115712. DOI: 10.1016/j.eswa.2021.115712.

[15] Z. Jin, M. Tao, X. Zhao, and Y. Hu, (2022) *"Social media sentiment analysis based on dependency graph and co-occurrence graph"* **Cognitive Computation *14***(3): 1039–1054. DOI: 10.1007/s12559-022-10004-8.

[16] F. Zhang, W. Zheng, and Y. Yang, (2024) *"Graph convolutional network with syntactic dependency for aspect-based sentiment analysis"* **International Journal of Computational Intelligence Systems *17***(1): 37. DOI: 10.1007/s44196-024-00419-6.

[17] P. Zhao, L. Hou, and O. Wu, (2020) *"Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification"* **Knowledge-Based Systems *193***: 105443. DOI: 10.1016/j.knosys.2019.105443.

[18] J. Yu, L. Zhao, S. Yin, and M. Ivanović, (2024) *"News recommendation model based on encoder graph neural network and bat optimization in online social multimedia art education"* **Computer Science and Information Systems *21***(3): 989–1012. DOI: 10.2298/CSIS231225025Y.

[19] R. Verma and R. Bhatt, (2025) *"Hybrid DCN-transformer framework with role-based access control (RBAC) policy for threats classification in cloud"* **International Journal of Information Technology**: 1–8. DOI: 10.1007/s41870-025-02743-2.

[20] M. Bayat and S. Kharel, (2025) *"Leveraging Artificial Intelligence for Predictive Maintenance and Condition Rating of Off-System Bridges"* **Applied Sciences *15***(21): DOI: 10.3390/app152111301.

[21] J. Chen and P. Wang. "Efficient Nearest Neighbor Promptbased Learning for Fewshot Ner in Manufacturing". In: *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2024, 1077–1082. DOI: 10.1109/SMC54092.2024.11169726.