GscFormer: A Graph-Structured and Cross-Masked Multimodal Transformer for Multimodal Sentiment Analysis

Anonymous COLING 2025 submission

Abstract

1 Introduction

As social networks become more prevalent, people are using a diverse range of media, such as text, videos and audios, to convey their feelings and viewpoints. This trend has led to the growth of multimodal sentiment analysis (MSA) as a prominent field of study. Most of the previous works on MSA have concentrated on all kinds of advanced sophisticated fusion strategies, spanning from tensor-based to attention-based and graph-based, or representation learning-based models with approaches include naive self-supervision learning, contrastive learning and knowledge distillation to learn robust representation of the commonalities among modalities and specificity of each modality.

These existing models have shown promising performance, however, there still remains an unsolved problem, it is not possible for them to structure the modal fusion process and meanwhile ensure the regularization of weights, so that they can not effectively achieve stable performance on more fine-grained tasks.

2 Related Work

2.1 Multimodal Sentiment Analysis

2.2 Graph Neural Networks

3 Methodology

In this section, we introduce the task setup of multimodal sentiment analysis(Section 3.1) and the overall architecture of GscFormer(Section 3.2). For the proposed model, the feature encoding procedure of raw modality is first described(Section 3.3). Then, we introduce in detail a module called Graph-Structured Cross-Modal Transformer designed to achieve modality fusion operations(Section 3.4). Finally, we describe briefly the self-supervision learning framework our model based on(Section 3.5).

3.1 Task Setup

The objective of Multimodal Sentiment Analysis (MSA) is to evaluate the sentiment strength or emotional classification by leveraging multimodal data. Existing multimodal datasets typically contain multimodal data include text(S_t), vision(S_v) and audio(S_a), where t, v, a refers to text, vision, audio modality seperately, specifically m refers to multi-modality. The input of the MSA task is utterances $S_u \in \mathbb{R}^{T_u^s \times d_u^s}$ derived from raw video fragments, where $u \in \{t, v, a\}$, T_u^s denotes the raw sequence length and d_u^s denotes the raw representation dimension of modality u. The model outputs unimodal outputs $\hat{y}_u \in R$ and multimodal fusion output $\hat{y}_m \in R$ as the final predictive result to fit the task whose ground truth sentiment label is denoted as $y_m \in R$.

3.2 Overall Architecture

The overview of our model is shown in Figure 3 which consists of three major parts: (1) *Modality Encoding* utilizes tokenizer (for text modality), feature extractors and temporal enhancers (firmware for non-verbal modality vision and audio) to convert raw multimodal signals into numerical feature sequences(word, vision and audio embedding). (2) *Graph-Structured Multimodal Fusion* takes the processed word, vision and audio embedding as input. The module Graph-Structured and Cross-Masked Multimodal Transformer utilizes cross mask to construct graph-structured modality representation which optimizes the bi-directional interactive fusion and intra-modal fusion enhancement operations. (3) *Self-Supervision Learning Framework* generates final representations and defines positive and negative centers by projecting original text embedding, enhanced vision and audio embedding, and fused output to hidden states, whereas uni-

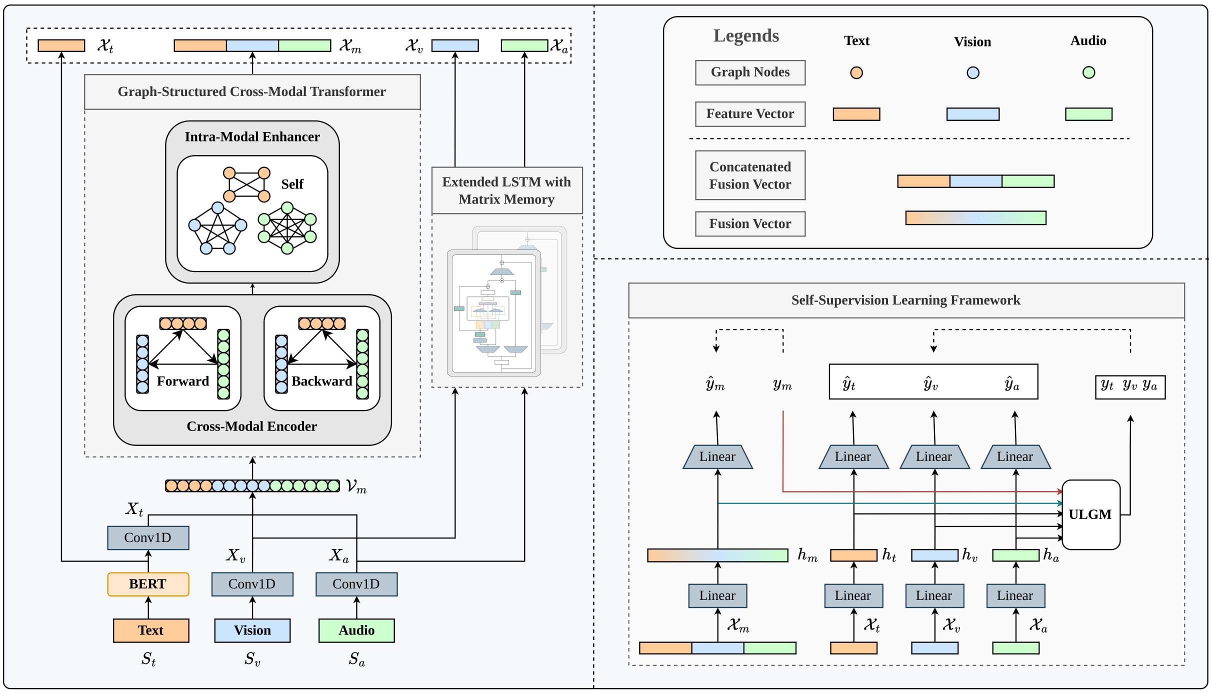

Figure 1: GscFormer Architecture.

modal labels are seperately generated using text, vision, and audio representations.

3.3 Modality Encoding

For text modality, we use the pretrained transfomer BERT as the text encoder. Input text token sequence is constructed by the raw sentence $S_t = \{w_1, w_2, \ldots, w_n\}$ concatenated with two special tokens ([CLS] at the head and [SEP] at the end) which forms $S_t^{'} = \{[CLS], w_1, w_2, \ldots, w_n, [SEP]\}$. Then, $S_t^{'}$ is inputted into the embedding layer of BERT which outputs the embedding sequence $\mathcal{X}_t = \{t_0, t_1, \ldots, t_{n+1}\}$. Following previous works, input sequences $X_u \in \mathbb{R}^{T_u \times d_u}$, where $u \in \{t, v, a\}$, T_u denotes the extracted sequence length and d_u denotes the extracted representation dimension of modality u, is extracted by one dimensional convolution layer from $\mathcal{X}_t$ and raw vision, audio sequences $S_{\{v,a\}}$.

$$X_t = \text{Conv1D}(\mathcal{X}_t) \tag{1}$$

$$X_{\{v,a\}} = \text{Conv1D}(S_{\{v,a\}}) \tag{2}$$

After that, we use an extended Long Short Term Memory which is fully parallelizable with a matrix memory and a covariance update rule (mLSTM) as the temporal signal enhancer of vision and audio sequence. The mLSTM forward pass and detailed achitecture is defined in the Appendix D.1

We use mLSTM networks to capture and enhance the temporal features of vision and audio:

$$\mathcal{X}_{\{v,a\}} = \text{mLSTM}(X_{\{v,a\}}) \tag{3}$$

The inputs of multimodal fusion layer are $\{X_t, X_v, X_a\}$, while which of the unimodal generator are $\{\mathcal{X}_t, \mathcal{X}_v, \mathcal{X}_a\}$, where $\mathcal{X}_u \in \mathbb{R}^{T_u \times d_u}$

3.4 Graph-Structured Multimodal Fusion

The multimodal data is originally unaligned, without alignment, effective fusion is unrealistic, it is of vital importance to optimize the aligning methods. Following previous works, we regard the low level temporal feature sequences $\{X_t, X_v, X_a\}$ as graph vertex sequences $\{\mathcal{V}_t, \mathcal{V}_v, \mathcal{V}_a\}$, where each time step is treated as a graph vertex. We concatenate vertices to a single sequence $\mathcal{V}_m = [\mathcal{V}_t; \mathcal{V}_v; \mathcal{V}_a]^\top$.

Graph Structure Construction To start with, we utilize self attention mechanism as the basic theory to construct a naive fully connected graph, in which the attention weight matrix is regarded as the adjacency matrix with dynamic weights, which is constructed as follows:

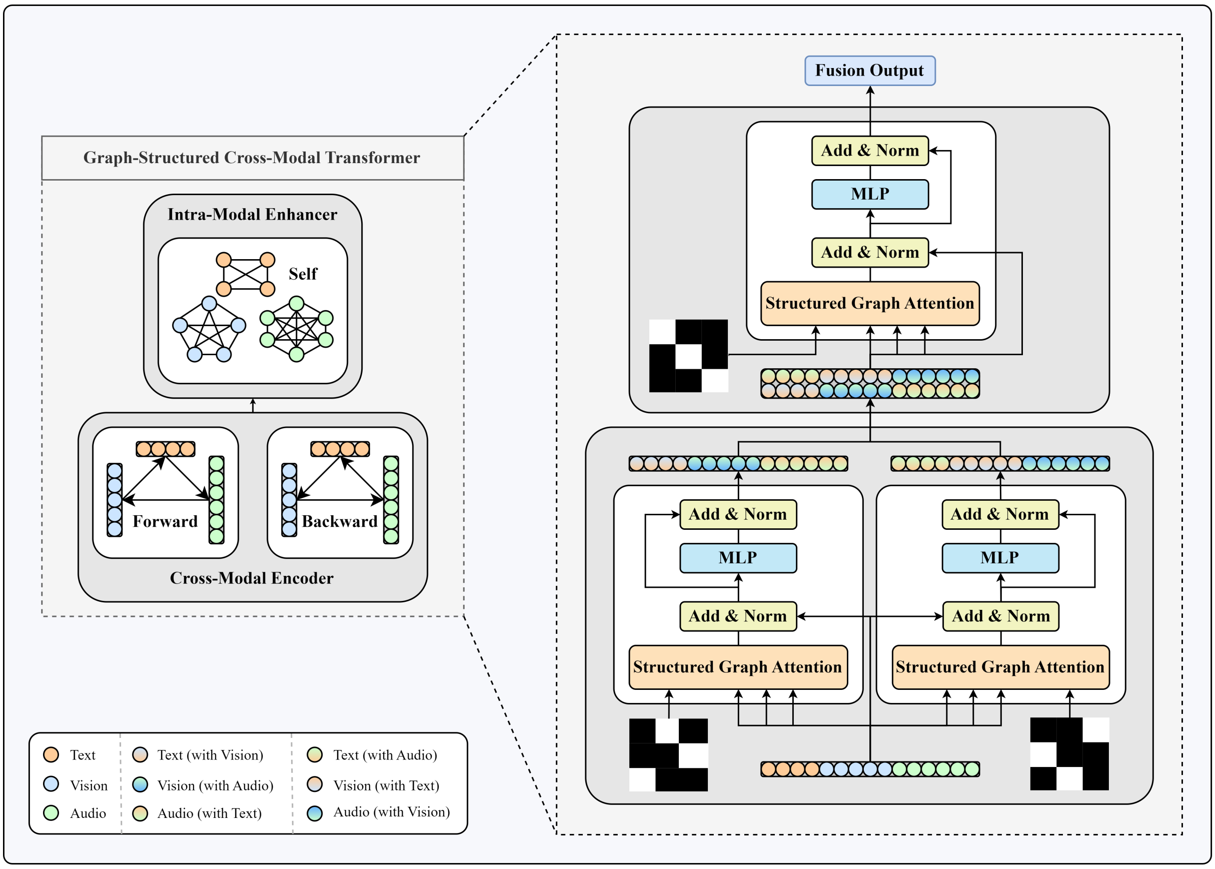

Figure 2: Graph-Structured Cross-Modal Transformer Architecture

$$\mathcal{A} = (\mathcal{W}_q * \mathcal{V}_m) \cdot (\mathcal{W}_k * \mathcal{V}_m)^\top = \begin{pmatrix} \mathcal{E}^{t,t} & \mathcal{E}^{v,t} & \mathcal{E}^{a,t} \\ \mathcal{E}^{t,v} & \mathcal{E}^{v,v} & \mathcal{E}^{a,v} \\ \mathcal{E}^{t,a} & \mathcal{E}^{v,a} & \mathcal{E}^{a,a} \end{pmatrix} \tag{4}$$

where $\mathcal{E}^{i,j} \in \mathbb{R}^{T_i \times T_j}$, $\{i, j\} \in \{t, v, a\}$ is the adjacency matrix of the subgraph constructed by $\mathcal{V}_i$ and $\mathcal{V}_j$.

Aggregation and Fusion Taking the modal sequence as the unit for dynamic edge weight aggregation, take $\mathcal{V}_i$ and $\mathcal{V}_j$ where $\{i, j\} \in \{t, v, a\}$ two modal vertex sequences as an example, assume that the subgraph constructed by the two modal vertex sequences is fully connected, so the adjacency matrix weight aggregation process of the corresponding subgraph is as follows.

$$\mathcal{E}^{i,j} = e(\mathcal{V}_j, \mathcal{V}_i) = (\mathcal{W}_q^j * \mathcal{V}_j) \cdot (\mathcal{W}_k^i * \mathcal{V}_i)^\top \tag{5}$$

Then, we apply $SoftMax$ function to the aggregated edge weight matrix in order to project its scalar elements into a row-wise probability space from 0 to 1.

$$\mathcal{G}(\mathcal{V}_j, \mathcal{V}_i) = \mathcal{S}(e(\mathcal{V}_j, \mathcal{V}_i)) = \frac{\exp(e(\mathcal{V}_j, \mathcal{V}_i))}{\sum_{i \in \mathcal{N}_j} \exp\left(e(\mathcal{V}_j, \mathcal{V}_{i'})\right)} \tag{6}$$

where $\mathcal{S}$ denotes the softmax function.

Finally, some of the edges in the subgraph are randomly masked which is realized by the dropout operation implemented on the adjacency matrix.

$$\mathcal{G}_{dropout}(\mathcal{V}_j, \mathcal{V}_i) = \mathcal{D}(\mathcal{G}(\mathcal{V}_j, \mathcal{V}_i)) \tag{7}$$

where $\mathcal{D}$ denotes the dropout function.

After the aggregation, fusion process is started, which is regarded as the directional information fusion procedure from $\mathcal{V}_j$ to $\mathcal{V}_i$.

$$\hat{\mathcal{V}}_i = \mathcal{G}_{dropout}(\mathcal{V}_j, \mathcal{V}_i) \cdot \mathcal{V}_j \tag{8}$$

Extend the above operation globally as follows:

$$\mathcal{G} = \mathcal{D} \circ \mathcal{S}(\mathcal{A}) \tag{9}$$

$$\hat{\mathcal{V}}_m = \mathcal{G} \cdot \mathcal{V}_m \tag{10}$$

Constructed graph structure is fully connected not only among modality vertex sequences but also inside the modality subgraphs in forms of edges and rings of vertices, which is actually unstructured at all, it loses sight of the separated modality-wise temporal features of the concatenated sequence which makes the sequence unordered, what is more, it over-fuses the inter-modal information, confuses inter-modal information and the intra-modal information and leaves way too much fine-grained information unconsidered. We propose the cross modal mask mechanism to solve this problem.

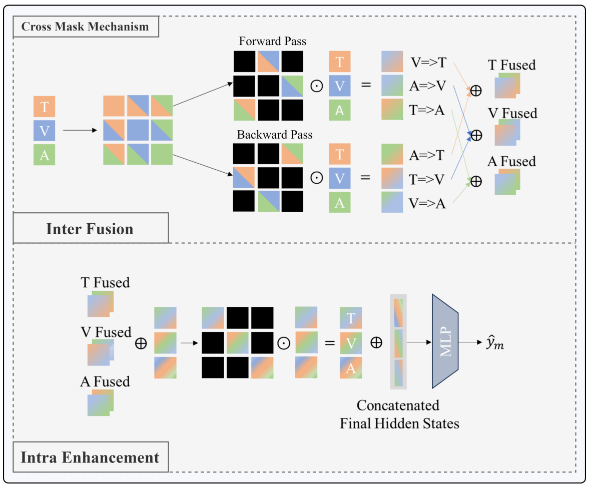

Figure 3: Cross Mask Mechanism

Cross Mask Mechanism In order to avoid the influence of intra-modal subgraph $\mathcal{E}^{i,j}$, we mask the subgraph adjacency matrix constructed by the modal node sequence itself for the corresponding block matrix in $\mathcal{G}$

$$\mathcal{M}_{inter} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \tag{11}$$

where $\mathcal{O}^{i,j} \in \mathbb{R}^{T_i \times T_j}$, $\mathcal{J}^{i,j} \in \mathbb{R}^{T_i \times T_j}$ denotes height of T_i width of T_j all respectively 0, negative infinity matrix

The mask matrix can realize that cross-modal fusion is not affected by subgraphs in each modality. However, the cross-modal fusion at this time also incorporates bidirectional information, and too much key information is still lost. Therefore, we extend the matrix to the following two mask matrices.

$$\begin{cases} \mathcal{M}_{inter}^{forward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{J}^{a,t} \\ \mathcal{J}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \\ \mathcal{M}_{inter}^{backward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{J}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{J}^{a,v} \\ \mathcal{J}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \end{cases} \tag{12}$$

Based on the two matrices, two uni-directional cyclic graphs can be constructed to achieve a bidirectional combination as follows:

$$\begin{cases} \mathcal{G}_{inter}^{forward} = \mathcal{D} \circ \mathcal{S}(\mathcal{A} + \mathcal{M}_{inter}^{forward}) \\ \mathcal{G}_{inter}^{backward} = \mathcal{D} \circ \mathcal{S}(\mathcal{A} + \mathcal{M}_{inter}^{backward}) \end{cases} \tag{13}$$

Then, the bidirectional combinational cyclic graph is fused with $\mathcal{V}_m$ to realize the bidirectional graph fusion.

$$\begin{cases} \hat{\mathcal{V}}_m^{forward} = \mathcal{G}_{inter}^{forward} \cdot \mathcal{V}_m \\ \hat{\mathcal{V}}_m^{backward} = \mathcal{G}_{inter}^{backward} \cdot \mathcal{V}_m \end{cases} \tag{14}$$

However, after the modal fusion, the subgraph of the modal still needs to be enhanced accordingly. At this time, the intra-modal enhancement mask can be constructed to realize the operation.

$$\mathcal{M}_{intra} = \mathcal{J} - \mathcal{M}_{inter} \tag{15}$$

where $\mathcal{J}$ denotes a negetive infinity matrix at the same size of $\mathcal{M}_{intra}$ y

Concatenate bidirectional features on the feature dimension with $\hat{\mathcal{V}}_m^{\{forward,backward\}}$:

$$\hat{\mathcal{V}}_m^{bidirection} = \| \; \hat{\mathcal{V}}_m^{\{forward,backward\}} \tag{16}$$

where $\|$ denotes the concatenation operation on the feature dimension

Utilizing the concatenated sequence $\hat{\mathcal{V}}_m^{bidirection}$, the intra-modal enhancement graph could be constructed as below.

$$\mathcal{A}_{fusion} = (\mathcal{W}_q^b * \mathcal{V}_m^b)(\mathcal{W}_k^b * \mathcal{V}_m^b)^\top \tag{17}$$

$$\mathcal{G}_{intra} = \mathcal{D} \circ \mathcal{S}(\mathcal{A}_{fusion} + \mathcal{M}_{intra}) \tag{18}$$

where $\mathcal{V}_m^b = \hat{\mathcal{V}}_m^{bidirection}$, $\mathcal{W}_q^b$, $\mathcal{W}_k^b$ denotes the query, key projection weight of $\mathcal{V}_m^b$

Then, we construct the final feature sequence.

$$\hat{\mathcal{V}}_m = \mathcal{G}_{intra} \cdot \hat{\mathcal{V}}_m^{bidirection} \tag{19}$$

Finally, the sequence is decomposed according to the length of the original feature sequence, and the final hidden states of different modes are taken respectively, which are spliced on the feature dimension to predict the final multimodal fusion representation.

$$\mathcal{X}_m = \|_m^{\{t,v,a\}} \hat{\mathcal{V}}_{\{t,v,a\}}^{final}[-1] \tag{20}$$

$$\hat{y}_m = Classifier(\mathcal{X}_m) \tag{21}$$

where $\|$ denotes the concatenation operation on the feature dimension.

The detailed generation algorithm of cross mask for inter-fusion and intra-enhancement is described in Appendix D.2

3.5 Self-Supervision Learning Framework

We integrate the uni-modal label generation module (ULGM) into our method to capture modality-specific information. As shown in Figure 3, $\mathcal{X}_{\{t,v,a\}}$ are utilized to generate the unimodal labels $\hat{y}_{\{t,v,a\}}$, while the final hidden states $h_{\{t,v,a\}}$ generated during the prediction procedure along with the ground truth multimodal label are obtained by ULGM to define the positive and negetive centers with predicted unimodal labels and multimodal fusion representations. Afterwards, we calculate the relative distance from the representation of each modality to the positive and negative centers, and obtain the offset value from the unimodal label to the ground truth multimodal label to generate new unimodal label $y_{\{t,v,a\}}^i$ for i^{th} epoch. In this way, it is more conducive to sentiment analysis to obtain differentiated information of different modalities while retaining the consistency of each modality.

Using the predicted results $\hat{y}_{\{m,t,v,a\}}$ and the ground truth multimodal label y_m along with the generated labels $y_{\{t,v,a\}}$, we implemented a weighted loss to optimize our model.

The weighted loss is defined by Equation 22 whereas the uni-modal loss for each modality is defined as Equation 23

$$\mathcal{L}_w = \sum_{u \in \{m,t,v,a\}} \mathcal{L}_u \tag{22}$$

$$\mathcal{L}_u = \frac{\sum_{i=0}^{\mathcal{B}} w_u^i * |\hat{y}_u^i - y_u^i|}{\mathcal{B}}$$
$$w_u^i = \begin{cases} 1 & u = m \\ \tanh(|\hat{y}_u^i - \hat{y}_m^i|) & u \in \{t,v,a\} \end{cases} \tag{23}$$

where $\mathcal{B}$ denotes the appointed batch size.

4 Experiment

4.1 Datasets

We evaluate our model on three benchmarks, CMU-MOSI, CMU-MOSEI and CH-SIMS. These datasets provide aligned (CMU-MOSI, CMU-MOSEI) and unaligned (all) mutlimodal data (text, vision and audio) for each utterance. Here, we give a brief introduction to the above datasets. Further details on the datasets are described in Appendix B

4.2 Evaluation Criteria

Following prior works, several evaluation metrics are adopted. Binary classification accuracy (Acc-2), F1 Score (F1), three classification accuracy (Acc-3), five classification accuracy (Acc-5), seven classification accuracy (Acc-7), mean absolute error (MAE), and the correlation of the model's prediction with human (Corr). Specifically, Acc-3 and Acc-5 are applied only for CH-SIMS dataset, Acc-2 and F1 are calculated in two ways: negative/non-negative(NN) and negative/positive(NP) on MOSI and MOSEI datasets, respectively.

4.3 Baselines

For CMU-MOSI and CMU-MOSEI, we choose MAG-BERT, MulT, MTAG, MISA, Self-MM, CENet, TETFN, ConFEDE and MTMD as baselines. As for CH-SIMS, on account of the data state of it is all unaligned, the baselines are different from those of the former two datasets, TFN, MFN, MISA, MulT, Self-MM and TETFN are chosen. For a more detailed introduction of the baseline models, please refer Appendix C

4.4 Results

The performance comparison of all methods on MOSI, MOSEI and CH-SIMS are summarized in Table 1 and Table 2.

In Table 1, for a fair comparison in CMU-MOSI and CMU-MOSEI, we split models into two categories for data state: Unaligned and Aligned, †

Table 1: Comparison on CMU-MOSI and CMU-MOSEI.

Model	CMU-MOSI					CMU-MOSEI					Data State
	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑	
MAG-BERT †	82.2 / 84.3	82.1 / 84.2	46.4	0.722	0.785	77.7 / 84.0	78.6 / 84.1	53.9	0.536	0.755	Aligned
MulT *	83.0 / -	82.8 / -	40.0	0.871	0.698	81.6 / -	81.6 / -	50.7	0.591	0.694	Unaligned
MTAG *	82.3 / -	82.1 / -	38.9	0.866	0.722	- / -	- / -	-	-	-	Unaligned
MISA *	81.8 / 83.4	81.7 / 83.6	42.3	0.783	0.761	83.6 / 85.5	83.8 / 85.3	52.2	0.555	0.756	Unaligned
Self-MM †	82.2 / 83.5	82.3 / 83.6	43.9	0.758	0.792	80.8 / 85.0	81.3 / 84.9	53.3	0.539	0.761	Unaligned
CENet †	82.8 / 84.5	82.7 / 84.5	45.2	0.736	0.793	81.7 / 82.3	81.6 / 81.9	52.0	0.576	0.711	Aligned
TETFN †	82.4 / 84.0	82.4 / 84.1	46.1	0.749	0.784	81.9 / 84.3	82.1 / 84.1	52.7	0.576	0.728	Unaligned
ConFEDE *	84.2 / 85.5	84.1 / 85.5	42.3	0.742	0.784	81.7 / 85.8	82.2 / 85.8	**54.9**	**0.522**	**0.780**	Unaligned
MTMD *	84.0 / **86.0**	83.9 / **86.0**	47.5	**0.705**	0.799	84.8 / 86.1	84.9 / 85.9	53.7	0.531	0.767	Unaligned
GscFormer	**85.0 / 86.0**	**85.0 / 86.0**	**48.3**	0.707	**0.801**	**85.0 / 86.3**	**85.1 / 86.2**	53.4	0.538	0.767	Unaligned

Table 2: Comparison results on CH-SIMS.

Model	CH-SIMS					
	Acc-2↑	Acc-3↑	Acc-5↑	F1↑	MAE↓	Corr↑
TFN	77.7	66.3	42.7	77.7	0.436	0.582
MFN	77.8	65.4	38.8	77.6	0.443	0.566
MulT	77.8	65.3	38.2	77.7	0.443	0.578
MISA	75.3	62.4	35.5	75.4	0.457	0.553
Self-MM	78.1	65.2	41.3	78.2	0.423	0.585
TETFN	78.0	64.4	42.9	78.0	0.425	0.582
GscFormer	**80.5**	**67.2**	**45.5**	**80.7**	**0.397**	**0.619**

Table 3: Ablation study on CMU-MOSI. Note: F denotes finetuning pretrained language models, NF denotes not finetuning

Description	CMU-MOSI				
	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑
Module Ablation					
GscFormer	**85.0 / 86.0**	**85.0 / 86.0**	**48.3**	**0.707**	**0.801**
w/o GscT	83.8 / 85.5	83.2 / 85.7	46.5	0.742	0.790
w/o mLSTM	84.6 / 86.0	84.5 / 86.0	47.2	0.730	0.792
w/o ULGM	83.4 / 84.8	83.4 / 84.8	46.7	0.711	**0.801**
Modality Ablation					
M(T,V,A)					
M(T,V)					
M(T,A)					
M(V,A)					
M(T)					
M(V)					
M(A)					
Pretrained Language Model Ablation					
BERT(F)	**85.0 / 86.0**	**85.0 / 86.0**	**48.3**	**0.707**	**0.801**
BERT(NF)	/	/			

denotes that the model is sourced from the GitHub page[1] and the scores are reproduced, * denotes the result is obtained directly from the original paper. For Acc-2 and F1, the left of the "/" corresponds to "negative/non-negetive" and the right corresponds to "negative/positive". For all metrics, the best results are highlighted in bold, and the weaker but still excellent results are double-underlined.

In Table 2, the best results are highlighted in bold, all of the models are sourced from the GitHub page[1] and the scores are reproduced.

Analysis on CMU-MOSI: As shown in the Table 1, the proposed GscFormer surpasses baselines on almost all the metrics on CMU-MOSI dataset. On Acc-2(NN&NP), F1(NN&NP), Acc-7 and Corr, it outperforms all the baselines, especially on Acc-2(NN), F1(NN) and Acc-7, GscFormer achieves a relative improvement of 0.8%, 0.9%, and 0.8% than the best performance of baselines. As for the MAE and Corr, it performs similar with the best baseline MTMD, with a 0.002 reduction on MAE and a 0.002 improvement on Corr.

Analysis on CMU-MOSEI As shown in the Table 1, GscFormer achieves the optimal performance on Acc-2(NN&NP) and F1(NN&NP) where performs admirably on Acc-2(NN) and F1(NN), which surpasses not only all the baselines an average of 3.3% on ACC-2(NN) and 3.0% on F1(NN) but also the best baseline MTMD 0.2% on both Acc-2(NN) and F1(NN). The results of Acc-7, MAE and Corr manage to reach a excellent level among all the baselines, although they were slightly weaker than the best baseline ConFEDE.

Analysis on CH-SIMS As shown in the Table 2, GscFormer achieves optimal results over all baselines, with at least 2.4% in Acc-2, 2.0% in Acc-3, 1.6% in Acc-5, 2.5% in F1, 0.026 in MAE, 0.034 in Corr, all of which are tremendous improvement.

4.5 Ablation Study

In this session, we will discuss our ablation study and its results in detail, which are divided into four parts in Table 6: Module Ablation, Modality Ablation and Pretrained Language Model Ablation.

Module Ablation There are three main modules in our model, including Graph-Structured Cross-Modal Transformer (GscT) for multimodal fusion, extended LSTM with matrix memory (mLSTM) for vision, audio temporal enhancement, Unimodal

[1] https://github.com/thuiar/MMSA

Table 4: Comparison of GscT and MulT on CMU-MOSI and CMU-MOSEI.

Model	CMU-MOSI					CMU-MOSEI					Parameters(M)	FLOPS(G)
	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑		
MulT	79.6 / 81.4	79.1 / 81.0	36.2	0.923	0.686	/	/				4.362	105.174
GscT	83.4 / 84.9	83.4 / 85.0	45.5	0.716	0.803	84.1 / 86.3	84.4 / 86.3	53.5	0.539	0.774	0.891	25.983

Table 5: Graph Structure Case Study on CMU-MOSI

Description	CMU-MOSI				
	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑
Orginal	**85.0 / 86.0**	**85.0 / 86.0**	**48.3**	**0.707**	**0.801**
Structure-1					
Structure-2					
Structure-3					
Self-Only	81.6 / 83.2	81.7 / 83.3	43.3	0.750	0.791

Table 6: The Computational Overhead of Different Vision/Audio Modality Enhancement Models

Model	V mLSTM	A mLSTM	ViT	Wav2Vec	Whisper
Parameters(M)			127.272	94.395	17.120
FLOPS(G)			35.469	68.543	315.128

Label Generation Module (ULGM) for self supervision. In Table 6 part module Ablation, w/o denotes the absence of corresponding module in the model.

The results in Table 6 indicates module GscT and ULGM are necessary for achieving state-of-the-art performance. GscT module constructs the graph structure of three modalities , without module GscT, the performance of the whole model has a substantial decrease in all metrics, especially 1.2% on Acc-2(NN), 1.8% on F1(NN), 1.8% on Acc-7, and 0.035 on MAE. Without module mLSTM, the performance weakens mainly on fine-grained metrics, 1.1% on Acc-7 and 0.023 on MAE. Without module ULGM, the performance weakens on all the metrics, especially on binary and seven classification task, 1.6%/1.2% on Acc-2 and 1.6%/1.2% on F1, 1.6% on Acc-7.

Modality Ablation In GscFormer, the multimodal representation (M) is used for the final classification task task. In the original case, M is composed of unimodal text (T), vision (V), and audio (A). In order to fully investigate the influence of the combined form of multimodal representation on the representation ability of the whole model, we designed the Modality Ablation study, which contains the three-modal case: M(T,V,A); the two-modal case: M(T,V), M(T,A), M(V,A); and the single-modal case: M(T), M(V), M(A). Note that the structure of the model in the single-modal case is already missing, and the graph structured attention degenerates to naive multi-head self-attention.

Pretrained Language Model Ablation

4.6 Case Study

Graph Structure Ablation The structure of the graph has a significant impact on the performance of the model, so we conduct an ablation study on its graph structure. The graph structure of the three modalities can only be constructed in the following four structures: Original strucuture (Org): $\{t$ -> v -> $a\}$ & $\{a$ -> v -> $t\}$; Structure-1: $\{a$ -> v -> a, a -> $t\}$ & $\{v$ -> t -> v, t -> $a\}$; Structure-2: $\{v$ -> t -> v, v -> $a\}$ & $\{a$ -> t -> a, a -> $v\}$; Structure-3: $\{a$ -> v -> a, v -> $t\}$ & $\{a$ -> t -> a, t -> $v\}$. As a contrast, we constructed a graph with only intra-mask which is diordered in multimodal temporal information: Self-Only: only mask the intra-modal subgraphs. The cross masks of all the structures is described in detail in Appendix A.2

4.7 Further Analysis

The complete GscFormer model has achieved excellent performance by achieving efficient alignment during multimodal fusion. In this section, we visualize its alignment and compare it fairly with MulT, a model that implements a similar function.

Alignment of Sequences

Comparison with MulT MulT mainly uses cross-attention to realize efficient modal sequence alignment and fusion. Like the core module of GscFormer, which is GscT, MulT realizes directional fusion and post-fusion enhancement between modes. However, GscT abstracts the whole process into a complete graph structure, uses interleaved mask to realize the graph structure construction, and merges forward and reverse processes respectively. The fusion process of this process realizes weight sharing with Transformer, avoids high-level weight isolation, avoids overfitting, and achieves better weight regularization.

Vision/Audio Encoder Efficiency

As shown in Table 4, .

As can be seen in Table Param, the parameter size of GscT is similar to that of MulT. By constructing large matrix, the serial operation of small matrix is integrated into parallel operation of large matrix, and high operation efficiency is achieved.

5 Conclusion

Limitations

References

A Experiment

A.1 Self-Supervision Modality Selection

Table 7: Self-Supervision Modality Selection on CMU-MOSI.

Description	CMU-MOSI				
	Acc-2↑	F1↑	Acc-7↑	MAE↓	Corr↑
M+T+V+A	**85.0 / 86.0**	**85.0 / 86.0**	**48.3**	**0.707**	**0.801**
M+T+V	84.4 / 85.7	84.3 / 85.7	44.5	0.742	0.742
M+T+A	83.9 / 85.7	83.7 / 85.6	46.1	0.731	0.796
M+V+A	83.8 / 85.2	83.8 / 85.3	44.6	0.748	0.794
M+T	83.4 / 85.7	83.3 / 85.6	45.0	0.731	0.796
M+V	83.5 / 85.4	83.5 / 85.4	45.8	0.724	**0.801**
M+A	82.5 / 84.6	82.4 / 84.6	46.1	0.709	0.800
M	83.4 / 84.8	83.4 / 84.8	46.7	0.711	**0.801**

In our proposed Self-Supervision Learning Framework, multi-modality (M) is used for classification, and unimodal text (T), vision (V), and audio (A) are used to generate unimodal labels in ULGM to ensure that the model learns a robust representation of the multimodal data. To fully analyze the importance of each modality in the model, we design modality selection experiments for self-supervised modality adoption. There are three modal label generation: M+T+V+A; two modal label generation: M+T+V, M+T+A, M+V+A; single modal label generation: M+T, M+V, M+A; and no label generation: M.

A.2 Graph Structures

The graph structures constructed in ablation study in the part Graph Structure Ablation of Table 6. Cross masks for each different graph structure are as follows:

Original Structure (Bidirectional)

$$\begin{cases} \mathcal{M}_{inter}^{forward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{J}^{a,t} \\ \mathcal{J}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \\ \mathcal{M}_{inter}^{backward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{J}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{J}^{a,v} \\ \mathcal{J}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \end{cases} \tag{24}$$

Structure-1:

$$\begin{cases} \mathcal{M}_{inter}^{forward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{J}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{J}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \\ \mathcal{M}_{inter}^{backward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{J}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{J}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \end{cases} \tag{25}$$

Structure-2:

$$\begin{cases} \mathcal{M}_{inter}^{forward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{J}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{J}^{a,v} \\ \mathcal{J}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \\ \mathcal{M}_{inter}^{backward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{J}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{J}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \end{cases} \tag{26}$$

Structure-3:

$$\begin{cases} \mathcal{M}_{inter}^{forward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{J}^{a,t} \\ \mathcal{J}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{J}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \\ \mathcal{M}_{inter}^{backward} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{J}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{J}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{J}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \end{cases} \tag{27}$$

Self-Only:

$$\mathcal{M}_{inter} = \begin{pmatrix} \mathcal{J}^{t,t} & \mathcal{O}^{v,t} & \mathcal{O}^{a,t} \\ \mathcal{O}^{t,v} & \mathcal{J}^{v,v} & \mathcal{O}^{a,v} \\ \mathcal{O}^{t,a} & \mathcal{O}^{v,a} & \mathcal{J}^{a,a} \end{pmatrix} \tag{28}$$

B Datasets

CMU-MOSI: The CMU-MOSI is a commonly used dataset for human multimodal sentiment analysis. It consists of 2,198 short monologue video clips (each clip lasts for the duration of one sentence) expressing the opinion of the speaker inside the video on a topic such as movies. The utterances are manually annotated with a continuous opinion score between [3, +3], [3: highly negative, 2 negative, 1 weakly negative, 0 neutral, +1 weakly positive, +2 positive, +3 highly positive].

CMU-MOSEI: The CMU-MOSEI is an improved version of CMU-MOSI. It contains 23,453 annotated video clips (about 10 times more than CMU-MOSI) from 5,000 videos, 1,000 different speakers, and 250 different topics. The number of discourses, samples, speakers, and topics is also larger than CMU-MOSI. The range of labels taken for each discourse is consistent with CMU-MOSI.

CH-SIMS: The CH-SIMS includes the same modalities in Mandarin: audio, text, and video, collected from 2281 annotated video segments. It includes data from TV shows and movies,making it culturally distinct and diverse, and provides multiple labels for the same utterance based on different modalities, which adds an extra layer of complexity and richness to the data.

C Baselines

TFN: The Tensor Fusion Network (TFN) uses modality embedding subnetwork and tensor fusion to learn intra- and inter-modality dynamics.

MFN: The Memory Fusion Network (MFN) explicitly accounts for both interactions in a neural architecture and continuously models them through time.

MAG-BERT: The Multimodal Adaptation Gate for Bert (MAG-Bert) incorporates aligned nonverbal information to text representation within Bert.

MulT: The Multimodal Transformer (MulT) uses cross-modal transformer based on cross-modal attention to make modality translation.

MTAG: The Modal-Temporal Attention Graph (MTAG) is a graph neural network model that incorporates modal attention mechanisms and dynamic pruning techniques to effectively capture complex interactions across modes and time, achieving a parametrically efficient and interpretable model.

MISA: The Modality-Invariant and -Specific Representations (MISA) projects representations into modality-sprcific and modality-invariant spaces and learns distributional similarity, orthogonal loss, reconstruction loss and task prediction loss

Self-MM: Learning Modality-Specific Representations with Self-Supervised Multi-Task Learning (Self-MM) [8] designs a multi- and a uni- task to learn inter-modal consistency and intra-modal specificity

CENet: Cross-Modal Enhancement Network (CENet) uses K-Means clustering to cluster the visual and audio modes into multiple tokens to realize the generation of the corresponding embedding, thus improving the representation ability of the two auxiliary modes and realizing a better BERT fine-tuning migration gate

TETFN: Text Enhanced Transformer Fusion Network (TETFN) strengthens the role of text modes in multi-modal information fusion through text-oriented cross-modal mapping and single-modal label generation, and uses Vision-Transformer pre-training model to extract visual features

ConFEDE: Contrastive Feature Decomposition (ConFEDE) constructs a unified learning framework that jointly performs contrastive representation learning and contrastive feature decomposition to enhance representation of multimodal information.

MTMD: Multi-Task Momentum Distillation (MTMD) treats the modal learning process as multiple subtasks and knowledge distillation between teacher network and student network effectively reduces the gap between different modes, and uses momentum models to explore mode-specific knowledge and learn robust multimodal representations through adaptive momentum fusion factors.

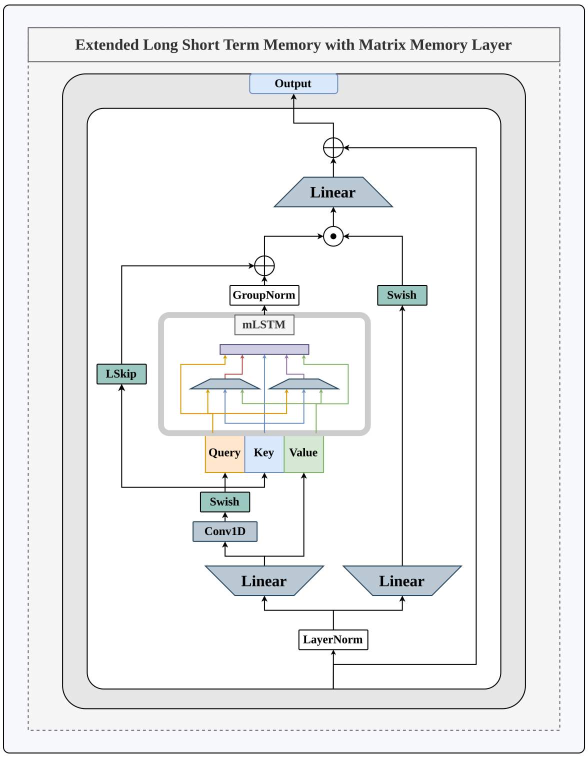

Figure 4: Parallelized Extended LSTM with Matrix Memory.

D Algorithms

D.1 Extended Long Short Term Memory with Matrix Memory

$$C_t = f_t C_{t-1} + i_t v_t k_t^\top \tag{29}$$

$$n_t = f_t n_{t-1} + i_t k_t \tag{30}$$

$$h_t = o_t \odot \tilde{h}_t, \qquad \tilde{h}_t = \frac{C_t q_t}{\max\{|n_t^\top q_t|, 1\}} \tag{31}$$

$$q_t = W_q x_t + b_q \tag{32}$$

$$k_t = \frac{1}{\sqrt{d}} W_k x_t + b_k \tag{33}$$

$$v_t = W_v x_t + b_v \tag{34}$$

$$i_t = \exp(\tilde{i}_t), \qquad \tilde{i}_t = w_i^\top x_t + b_i \tag{35}$$

$$f_t = \sigma(\tilde{f}_t) \text{OR} \exp(\tilde{f}_t), \qquad \tilde{f}_t = w_f^\top x_t + b_f \tag{36}$$

$$o_t = \sigma(\tilde{o}_t), \qquad \tilde{o}_t = W_o x_t + b_o \tag{37}$$

The forward pass of mLSTM can be described as the above equation group, while the detailed architecture is shown in Figure 4

D.2 Cross Mask Generation Algorithm

Algorithm 1 Cross Masking Algorithm

```
Input: Segmentation of the length of three-modal sequence seg = $\{T_t, T_v, T_a\}$, Mode of the mask generation $mode \in \{inter, intra\}$, Direction of fusion procedure $dir \in \{forward, backward\}$;
Output: The generated mask of appointed mode and direction;
Let $\{l_t, l_v, l_a\} = seg$
Define segments $s1 = (0, l_t)$, $s2 = (l_t, l_t + l_v)$, $s3 = (l_t + l_v, l_t + l_v + l_a)$
Let $l_{sum} = l_t + l_v + l_a$
Initialize an empty list $\mathcal{M}_{list}$
for each $i$ in $[0, 1, 2]$ do
    for each element in $seg[i]$ do
        Initialize $m_{row}$ as a tensor of ones with size $l_{sum}$
        if $i == 0$ then
            Set $m_{row}[0 : s1[1]] = 0$
            if $mode == cross$ then
                if $dir == forward$ then
                    Set $m_{row}[s3[0] :] = 0$
                else if $dir == backward$ then
                    Set $m_{row}[s2[0] : s2[1]] = 0$
                end if
            end if
        else if $i == 1$ then
            Set $m_{row}[s2[0] : s2[1]] = 0$
            if $mode == cross$ then
                if $dir == forward$ then
                    Set $m_{row}[s3[0] :] = 0$
                else if $dir == backward$ then
                    Set $m_{row}[0 : s1[1]] = 0$
                end if
            end if
        else if $i == 2$ then
            Set $m_{row}[s3[0] : s3[1]] = 0$
            if $mode == cross$ then
                if $dir == forward$ then
                    Set $m_{row}[0 : s1[1]] = 0$
                else if $dir == backward$ then
                    Set $m_{row}[s2[0] : s2[1]] = 0$
                end if
            end if
        end if
        Append $m_{row}$ to $\mathcal{M}_{list}$
    end for
end for
if $mode == cross$ then
    Let $\mathcal{M} = \text{Stack}(\mathcal{M}_{list})$
    return GenerateMask($\mathcal{M}$)
else if $mode == self$ then
    return GenerateMask($|\text{Stack}(\mathcal{M}_{list}) - 1)|$)
end if=0
```

The detailed generation method of cross mask for not only the forward and backward inter-fusion but also the intra-enhancement is shown on the algorithm table above. It is of vital importance for our model to accurately construct the graph structure of the concatenated sequence list.

Also, the masks could be constructed during the initialization procedure.