Evidence-aware Fake News Detection with Graph Neural Networks

Weizhi Xu[1,2,*], Junfei Wu[3,*,‡], Qiang Liu[1,2], Shu Wu[1,2,†], Liang Wang[1,2*†‡]
[1]Center for Research on Intelligent Perception and Computing (CRIPAC)
National Laboratory of Pattern Recognition (NLPR)
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]School of Computer Science and Technology, Beijing Institute of Technology
weizhi.xu@cripac.ia.ac.cn, junfei.wu@bit.edu.cn, {qiang.liu, shu.wu, wangliang}@nlpr.ia.ac.cn

WWW'22

220504 Chia-Chun Ho

Outline

Introduction

Methodology

Experiments

Conclusion

Comments

Introduction

Fake news

- Which is always fabricated by making some minor changes to the correct statement, is highly deceptive and indistinguishable.

- The widespread of fake news in diverse domains, such as politics and public health, has posed a huge threat to web security and human society.

- Therefore, the research on automatic fake news detection is challenging but in demand.

Introduction
Pattern/evidence -based approaches

- Pattern-based approaches
  - Regard the fake news detection as a feature recognition task, where language models are employed to verify the veracity according to the text pattern.
  - Usually suffer from the poor generalization and interpretability.
- Evidence-based approaches
  - Model the task as a reasoning process, where external evidences are provided to probe the veracity of a claim.
  - Required to discover and integrate useful information in given evidences for claims verification.

Introduction

Evidence-based pipeline

- In this paper, concentrate on the evidence-bases pipeline.
- Existing methods usually follow a two-step paradigm:
  - They first capture the semantics of claims and evidences separately.
  - Next, they model the claim-evidence interaction to explore the semantic coherence or conflict for more accurate and interpretable verdict.

Introduction

Existing evidence-based methods

- To name a few representative models:
  - DeClarE(EMNLP'18): utilize Bi-LSTMs to model textual features, followed by a word-level attention mechanism to capture the claim-evidence interaction.
  - HAN(ACL'19): further considers the sentence-level interaction to explore more general semantic coherence.
- To obtain multi-level semantic interaction, some recent works employ hierarchical attention networks.
  - MAC(ECAL'21), CICD(arXiv'21)

Introduction
Weakness

- Existing work focuses on the specific design of different interaction models (2nd step) while neglecting exploring fine-grained semantics of claims and evidences (1st step).

- There are two main weaknesses in previous methods:

    - Long-distance semantic dependency is less explored.

    - Existing methods neglect the redundant information involved in semantics.

Introduction

Long-distance example

- Look at evidence, two highlighted snippets are separated by plenty of words, which induces a long distance between them.
- Therefore, fusing the information is indispensable and beneficial for claim veracity prediction.
- However, long-distance semantic dependency between such information is hard to be captured due to inherent drawbacks of sequential models.

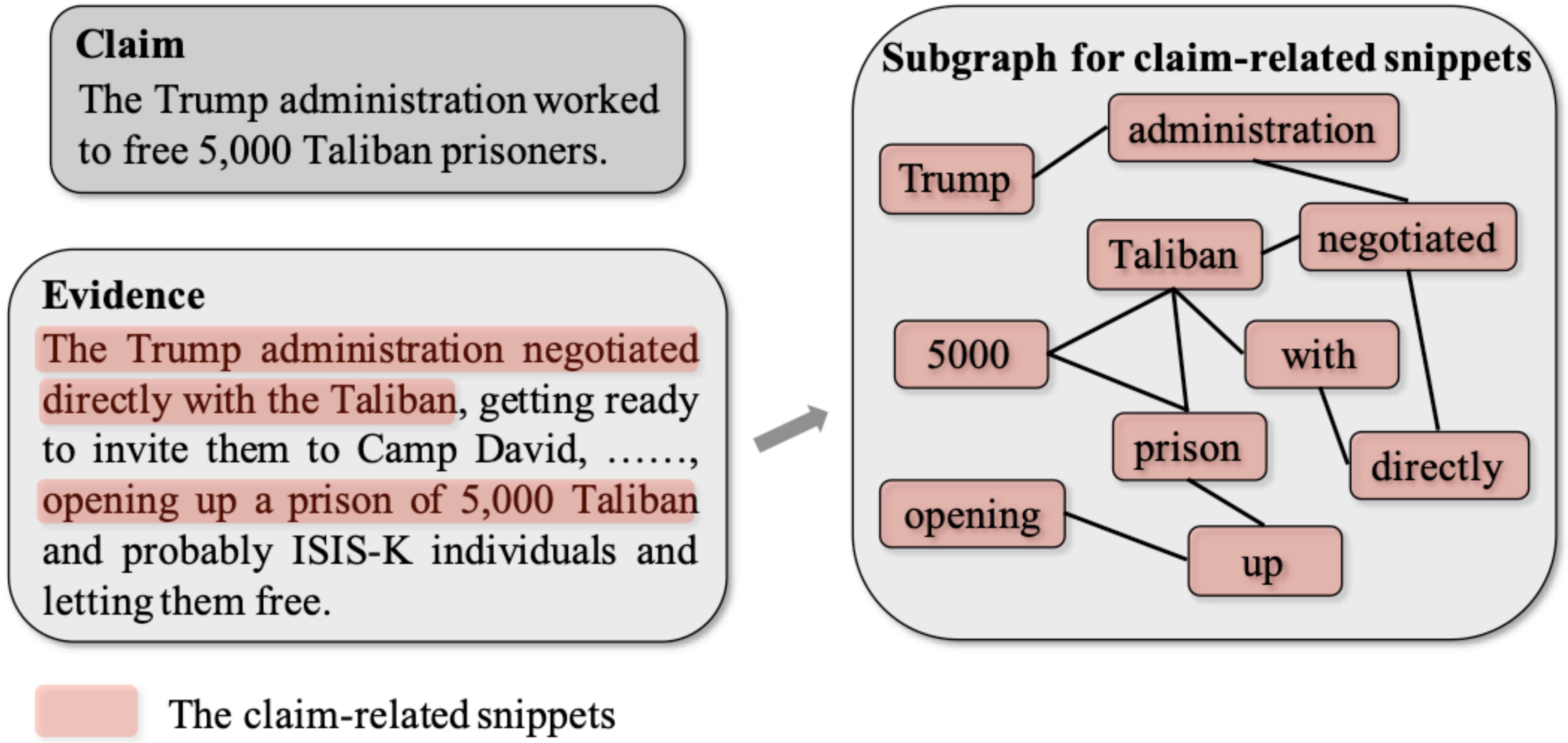


Figure 1: A toy example where a claim and its relevant evidence are given. Two significant snippets for verifying the claim are highlighted ("....." represents that we omit several sentences for conciseness). The right graph is constructed according to the highlighted snippets. Such two snippets have a long distance in the plain text while they are pulled close on the constructed semantic graph via the shared keyword "Taliban". Besides, there is much redundant information (texts except the highlighted parts), which is useless for claim verification.

Introduction

Redundant information example

- Such redundancy is useless or even harmful for fake news detection.
- Though previous models employ attention mechanisms to reduce the effect of unrelated words, these irrelevant texts are still preserved.
- May introduce noises to the downstream claim-evidence interaction, deteriorating the final performance of veracity checking.

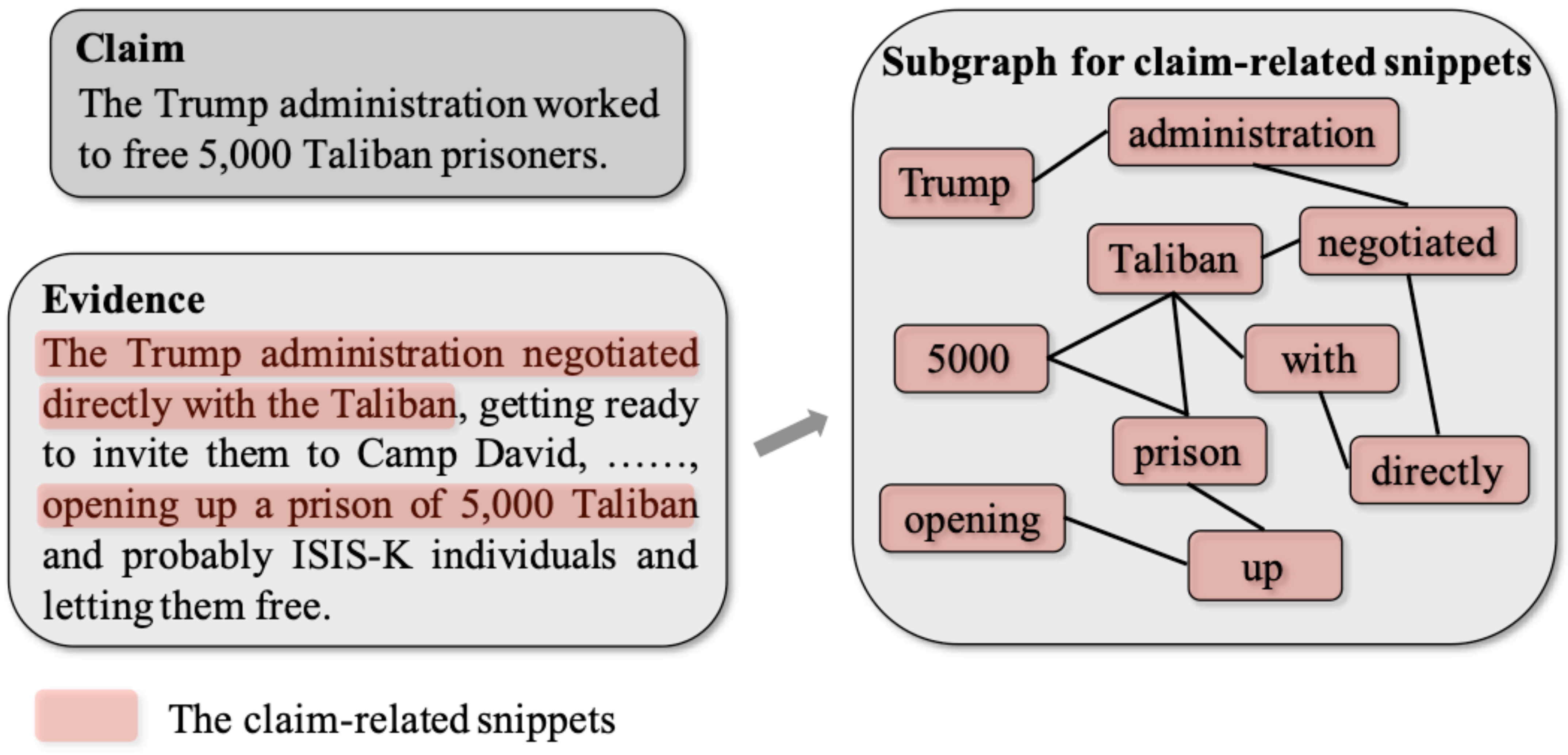


Figure 1: A toy example where a claim and its relevant evidence are given. Two significant snippets for verifying the claim are highlighted ("....." represents that we omit several sentences for conciseness). The right graph is constructed according to the highlighted snippets. Such two snippets have a long distance in the plain text while they are pulled close on the constructed semantic graph via the shared keyword "Taliban". Besides, there is much redundant information (texts except the highlighted parts), which is useless for claim verification.

Introduction

Redundant information solution

- An intuitive solution is to discard words with low attentive scores based on previous methods.
    - However, they compute the score for each word independently, ignoring the complex semantic structure among words.
- Authors argue that it is significant to modeling the redundancy with rich semantic structural information, as the redundancy is not only related to the self-information, but also induced by its contexts.
    - e.g., if a claim can be verified by a snippet in an evidence, the snippet's context will be redundant.

Introduction
GET

- Proposed a unified Graph-based sEmantic sTructure mining framework for exploring fine-grained semantics.
- Specifically, modeling sequential data as graphs can capturing long-distance structural dependency.
- To this end, utilize graph structure to model both claims and evidences, where
  - Nodes indicate words and
  - Edges represent the co-occurrence between two words.

Introduction

GET example

- Thereafter, the dispersed claim-related snippets are pulled close on graphs.
- Thus the useful information could be better fused via neighborhood propagation.

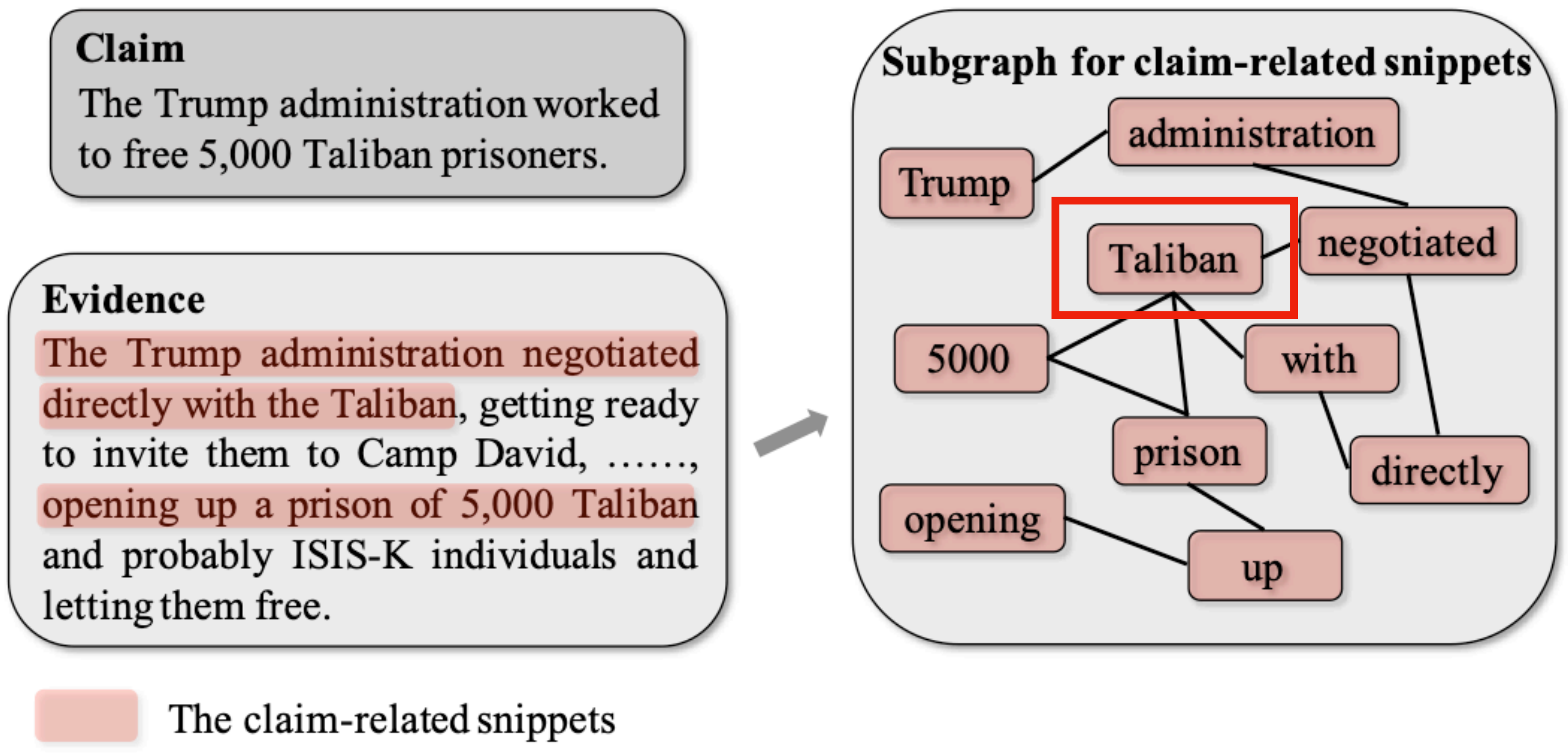


Figure 1: A toy example where a claim and its relevant evidence are given. Two significant snippets for verifying the claim are highlighted ("....." represents that we omit several sentences for conciseness). The right graph is constructed according to the highlighted snippets. Such two snippets have a long distance in the plain text while they are pulled close on the constructed semantic graph via the shared keyword "Taliban". Besides, there is much redundant information (texts except the highlighted parts), which is useless for claim verification.

Introduction
GET

- Moreover, to alleviate the negative impact of redundant information.
  - Treat the redundancy mitigation as a graph structure learning process.
  - Unimportant nodes are discarded according to complex semantic structures.
  - i.e., both self-features (node attributes) and their contexts (graph topology).

Introduction

Contributions

- Model claims and evidences as graph-structured data and design a graph-based framework to explore the complex semantic structure.
  - To the best of authors' knowledge, this is the first work to propose a unified graph-based method for evidence-based fake news detection.
- Introduce a simple and effective graph structure learning approach for redundancy mitigation.
- By capturing long-distance semantic dependency and reducing redundancy, obtain the fine-grained semantics, which can boost the performance of downstream interaction models.

Methodology
Task Formulation

- Evidence-based fake news detection is a classification task.
    - The model is required to output the prediction of news veracity.

predicted probability of veracity

claim

several related evidences $\{e_1, e_2, \ldots, e_n\}$

$$\hat{y} = f(c, \mathcal{E}, \mathbf{s}, \mathbf{p}, \Theta)$$

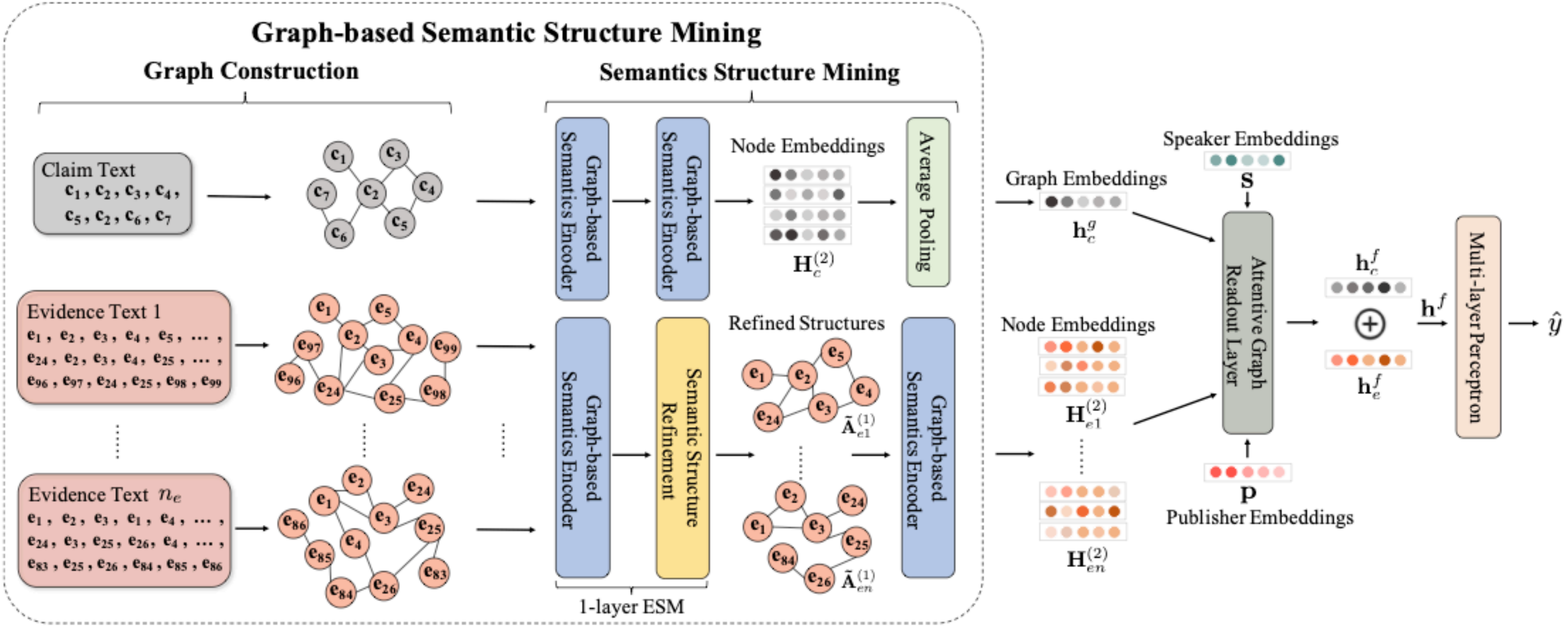
trainable parameters

verification model

their corresponding speakers $\mathbf{s} \in \mathbb{R}^{1\times b}$ or publishers $\mathbf{p} \in \mathbb{R}^{n\times b}$

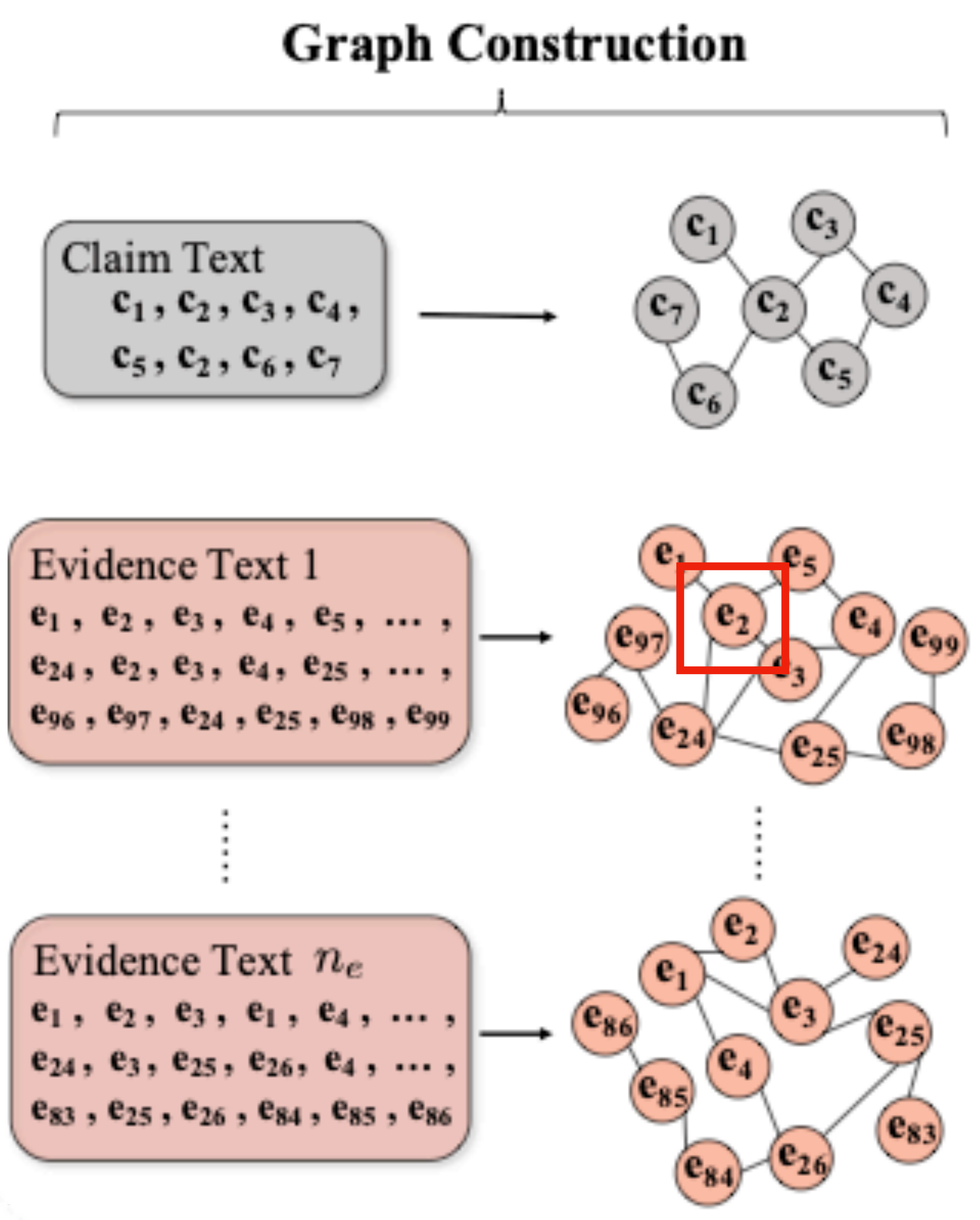
Methodology

Graph-based sEmantic sTructure mining framework (GET)



Methodology

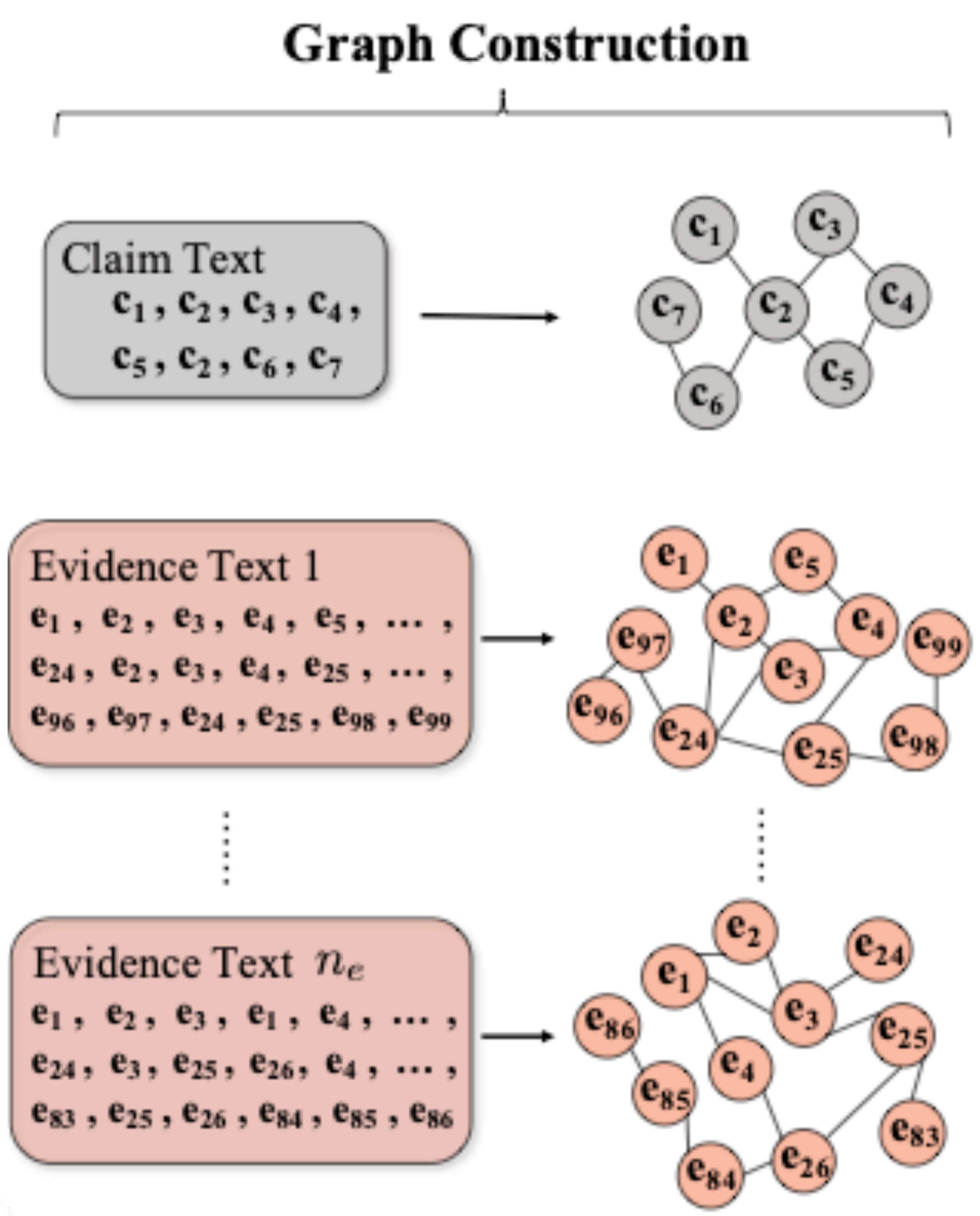
Graph Construction



- First convert original claims and evidences to graphs.
- Use fix-sized sliding window to screen out the connectivity for each word on graphs.
- To model long-distance dependency, merge all same words into one node on graph.
    - Several relevant snippets that scatter far apart is close on graphs, which can be explored via high-order message propagation.
- Initial node representations are corresponding word embeddings.

Methodology

Graph Construction



- To ensure numerical stability, perform Laplacian normalization on adjacency matrices.
- $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\mathbf{D}^{-\frac{1}{2}}, \mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$
- Finally, denote the initial normalized adjacency matrices and node feature matrices of claim and evidence as
- $\tilde{\mathbf{A}}_c^{(0)} \in \mathbb{R}^{N_c \times N_c}, \tilde{\mathbf{A}}_e^{(0)} \in \mathbb{R}^{N_e \times N_e}$ and $\mathbf{H}_c^{(0)} \in \mathbb{R}^{N_c \times d}, \mathrm{H}_e^{(0)} \in \mathbb{R}^{N_e \times d}$

Adjacency matrix of claim

Adjacency matrix of evidences

Node features of claim

Node features of evidences

Claim

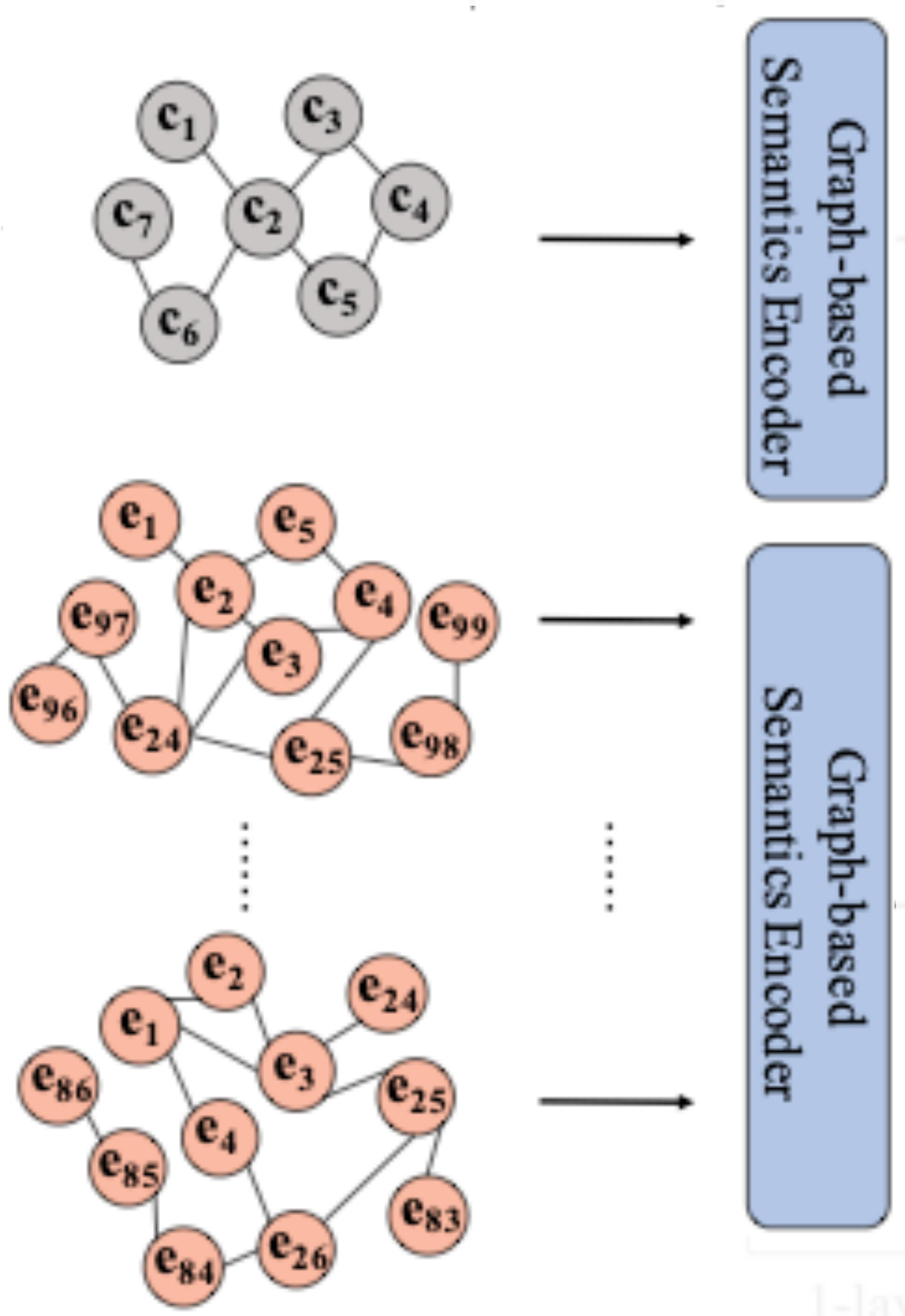
Evidences

# of nodes in initial claim graph

# of nodes in initial evidence graph

Methodology

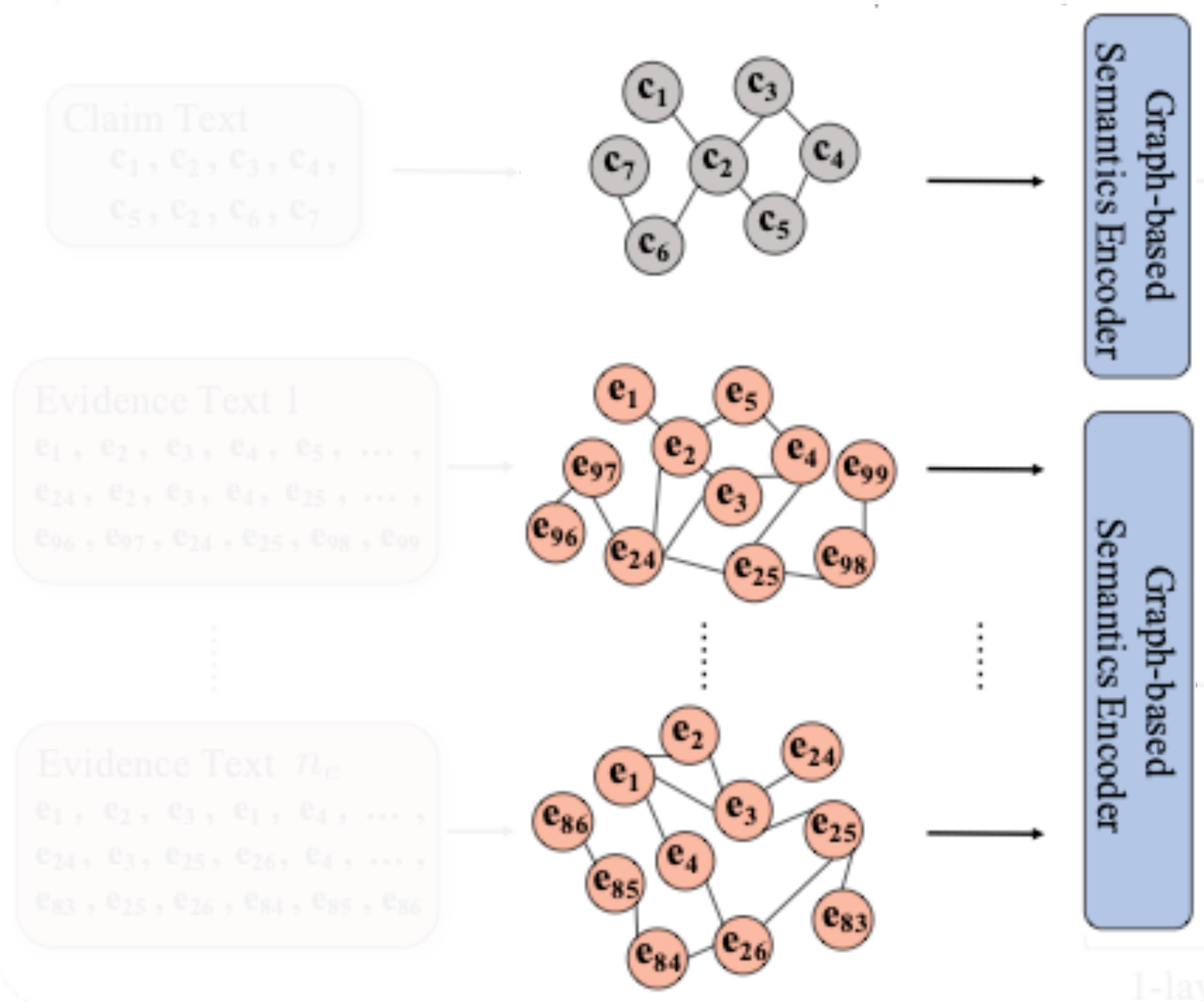
Graph-based Semantics Encoder



- To mine the long-distance semantic dependency, propose to utilize GNNs as the semantics encoder.
- Expect to adaptively keep a balance between self-features and info of neighboring nodes.
- Employ graph gated neural networks (GGNN) to perform neighborhood propagation on both claim and evidence graphs.
  - Enabling nodes to capture their contextual information, which is significant for learning high-level semantics.

Methodology

Graph-based Semantics Encoder

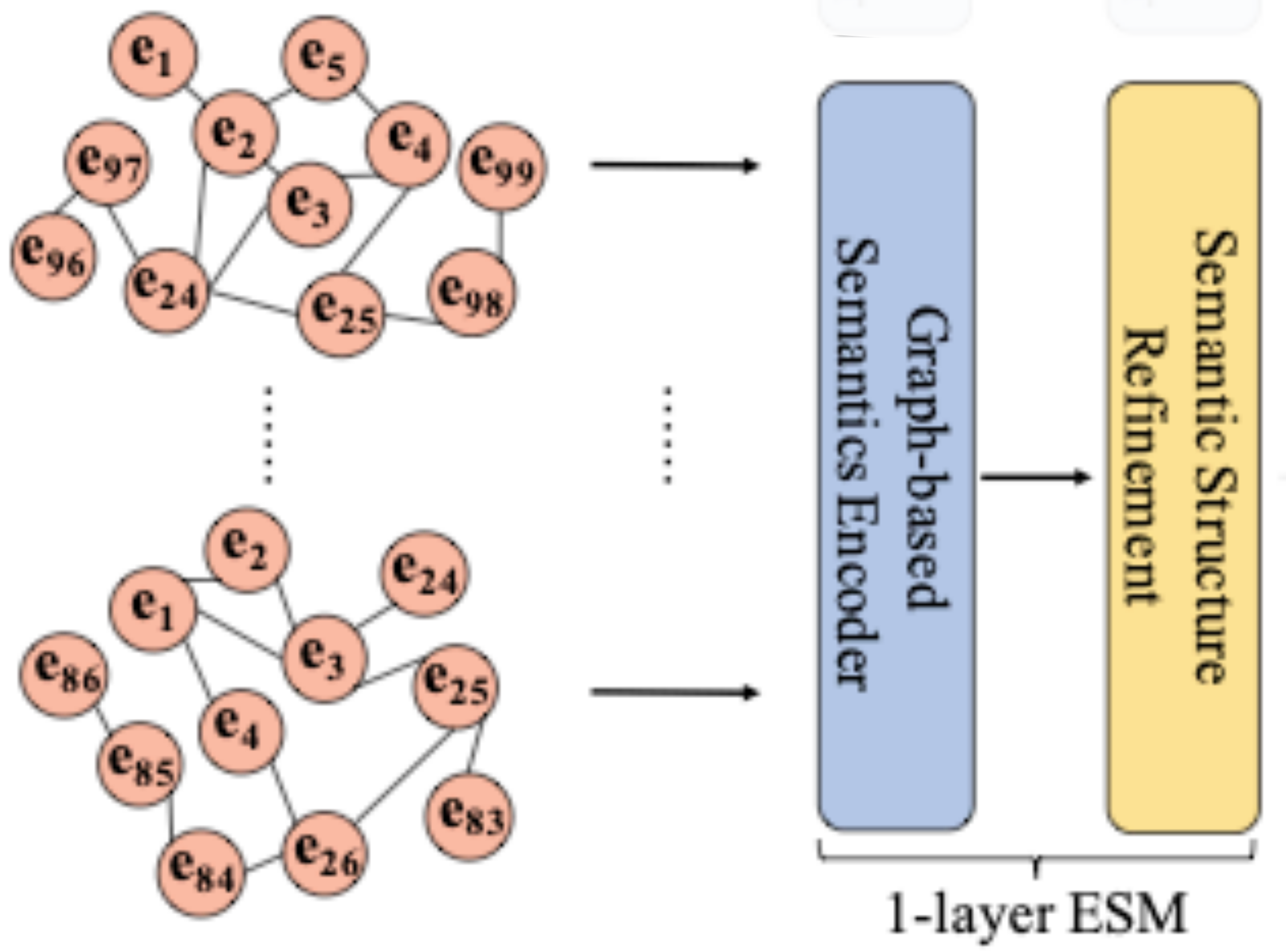


- $\mathbf{a}_i = \sum_{(w_i, w_j) \in C} \tilde{\mathbf{A}}_{ij} \mathbf{W}_a \mathbf{H}_j$ Information of neighboring nodes
- $\mathbf{z}_i = \sigma(\mathbf{W}_z \mathbf{a}_i + \mathbf{U}_z \mathbf{H}_i + \mathbf{b}_z)$ Update Gate
- $\mathbf{r}_i = \sigma(\mathbf{W}_r \mathbf{a}_i + \mathbf{U}_r \mathbf{H}_i + \mathbf{b}_r)$ Reset Gate
- $\tilde{\mathbf{H}}_i = \tanh(\mathbf{W}_h \mathbf{a}_i + \mathbf{U}_h \left(\mathbf{r}_i \odot \mathbf{H}_i\right) + \mathbf{b}_h)$ New information
- $\hat{\mathbf{H}}_i = \tilde{\mathbf{H}}_i \odot \mathbf{z}_i + \mathbf{H}_i \odot (1 - \mathbf{z}_i)$
- For brevity, denote above equations as $\mathbf{GGNN}(\tilde{\mathbf{A}}, \mathbf{H})$.

Methodology

Semantic Structure Refinement

- As evidences always contain redundant information that may mislead model to focus on unimportant features.
- It is beneficial to discover and filter out the redundancy, thus obtaining refined semantic structures.



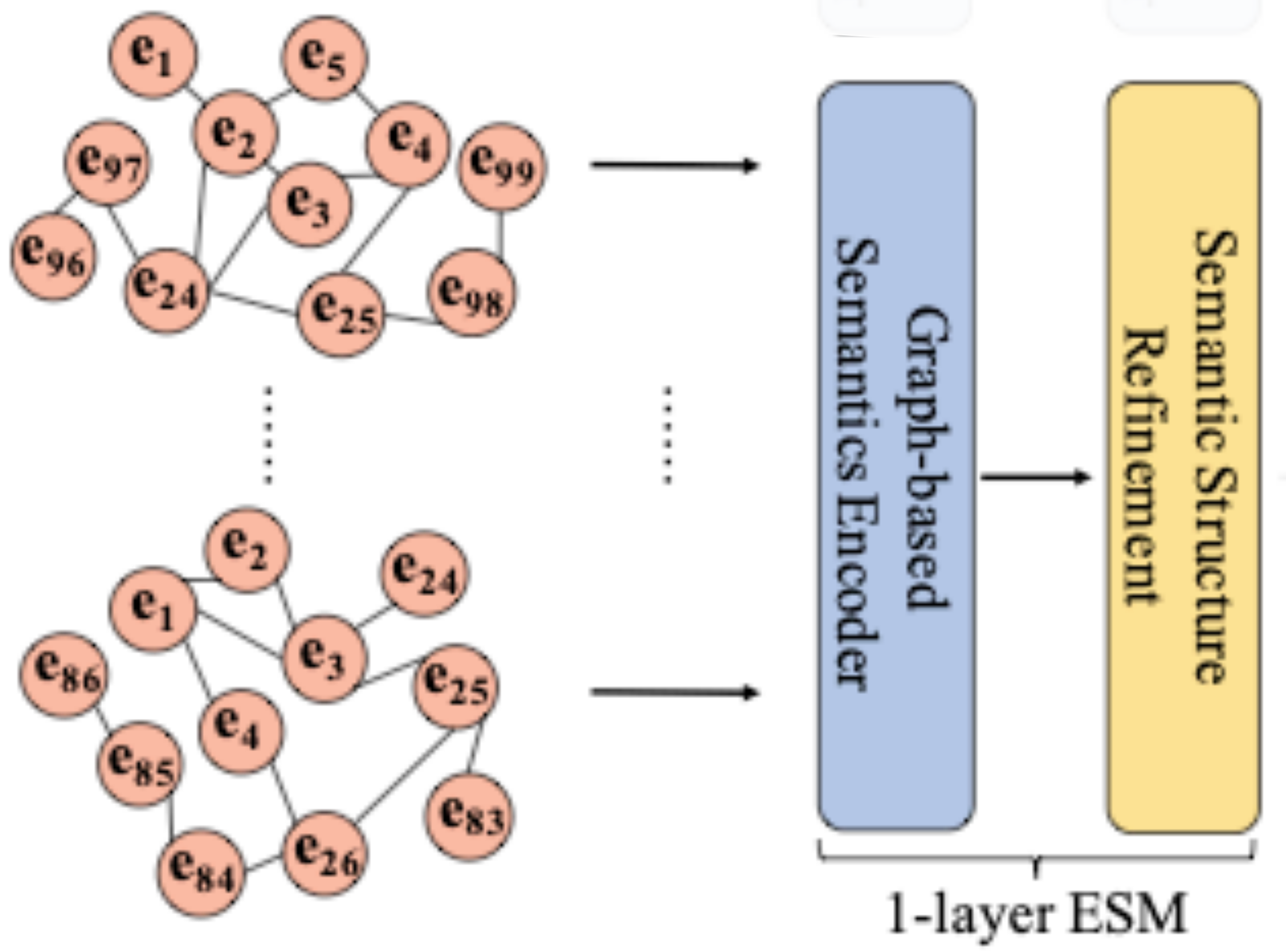
- Treat redundancy mitigation as a graph structure learning process.
- To learn optimized graph topology along with better node representations.

Methodology

Semantic Structure Refinement

- Since redundancy information is mainly involved in words denoted as nodes in evidence graphs, attempt to refine evidence graph structures via discarding redundant nodes.

- Propose to compute a redundancy score for each node, based on which obtain a ranking list and nodes with the top-k redundancy scores will be discarded.



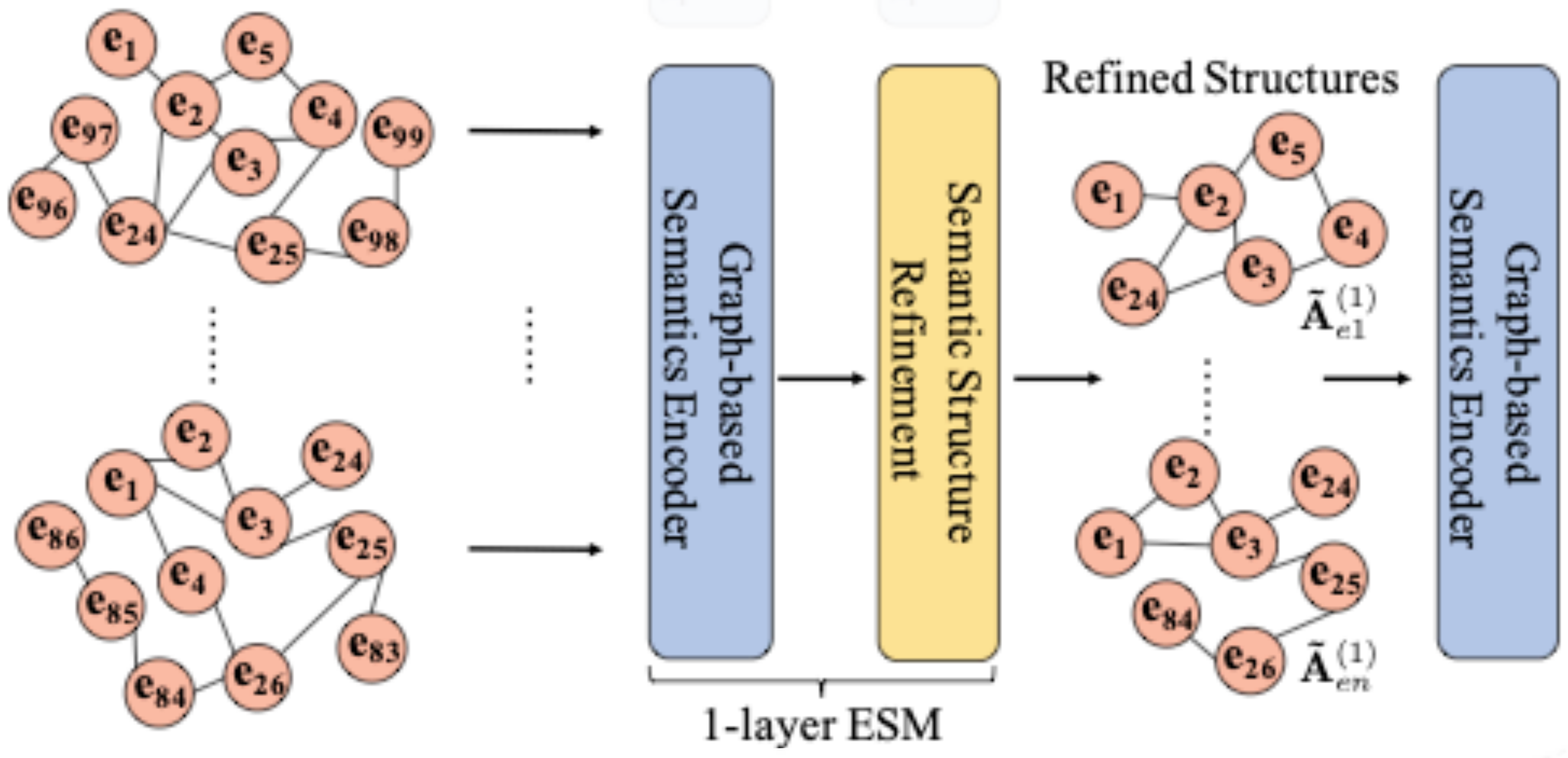
- Utilize a 1-layer GGNN to compute the redundancy scores.
- $\mathbf{s}_r = \mathbf{GGNN}(\tilde{\mathbf{A}}, \hat{\mathbf{H}}_e \mathbf{W}_s)$ trainable weights
- $idx = \mathrm{topk\_index}(\mathbf{s}_r)$ Indices of node with top-k redundancy scores
- $\tilde{\mathbf{A}}_{idx,:} = \tilde{\mathbf{A}}_{:,idx} = 0$ which are discarded by masking their degrees as 0

Methodology

Semantic Structure Refinement

- Only perform semantic structure refinement on evidences since claims are usually short (less than 10 words) so that the semantic structures are simple and unnecessary to be refined.

- Stack semantic structure refinement layer over one semantics encoder to form a unified module, namely evidence semantics miner (ESM in short).

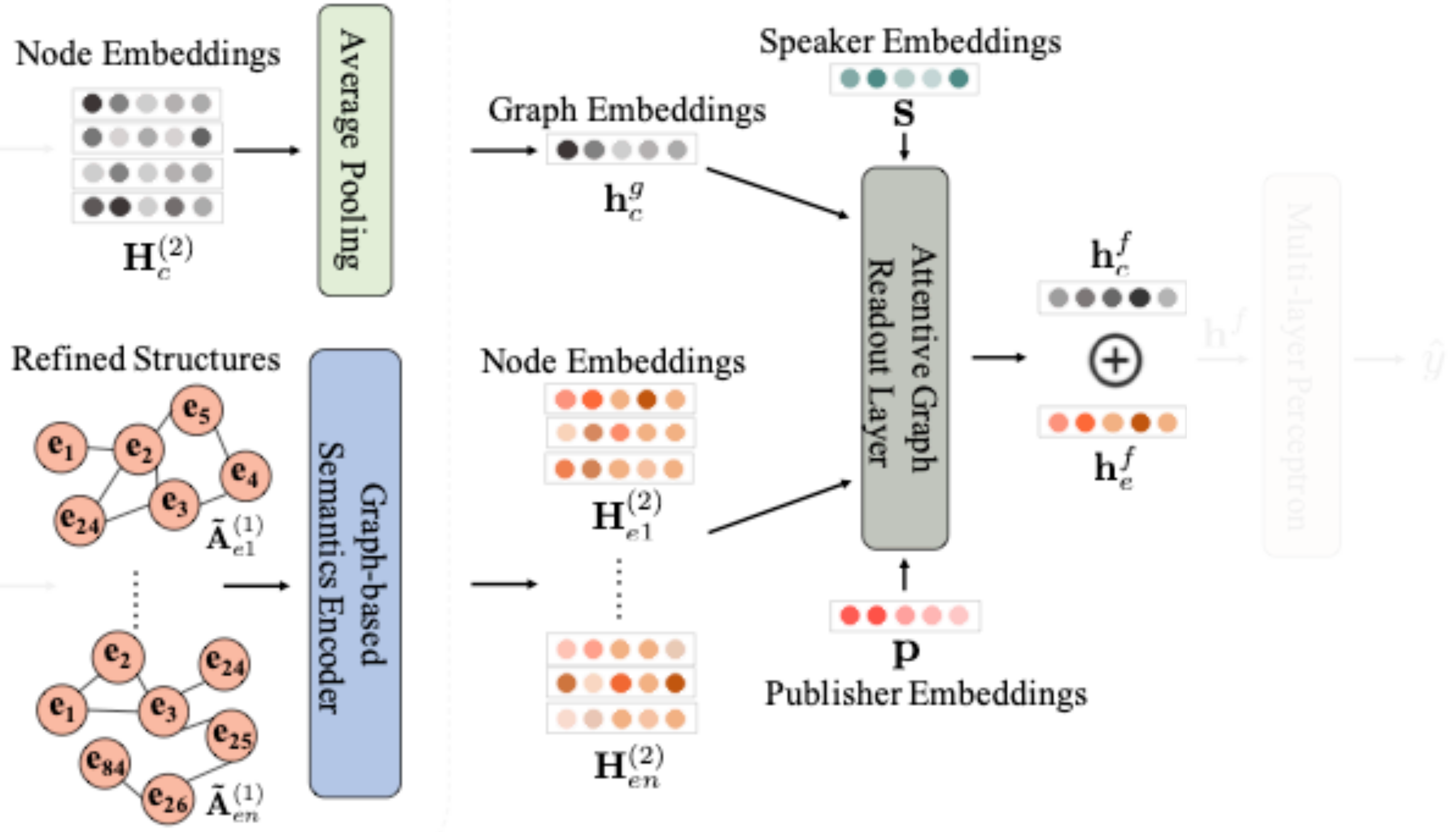


- Eventually followed by a semantics encoder to perform neighborhood propagation on refined semantic graphs, obtaining the fine-grained representations.

Methodology

Attentive Graph Readout Layer

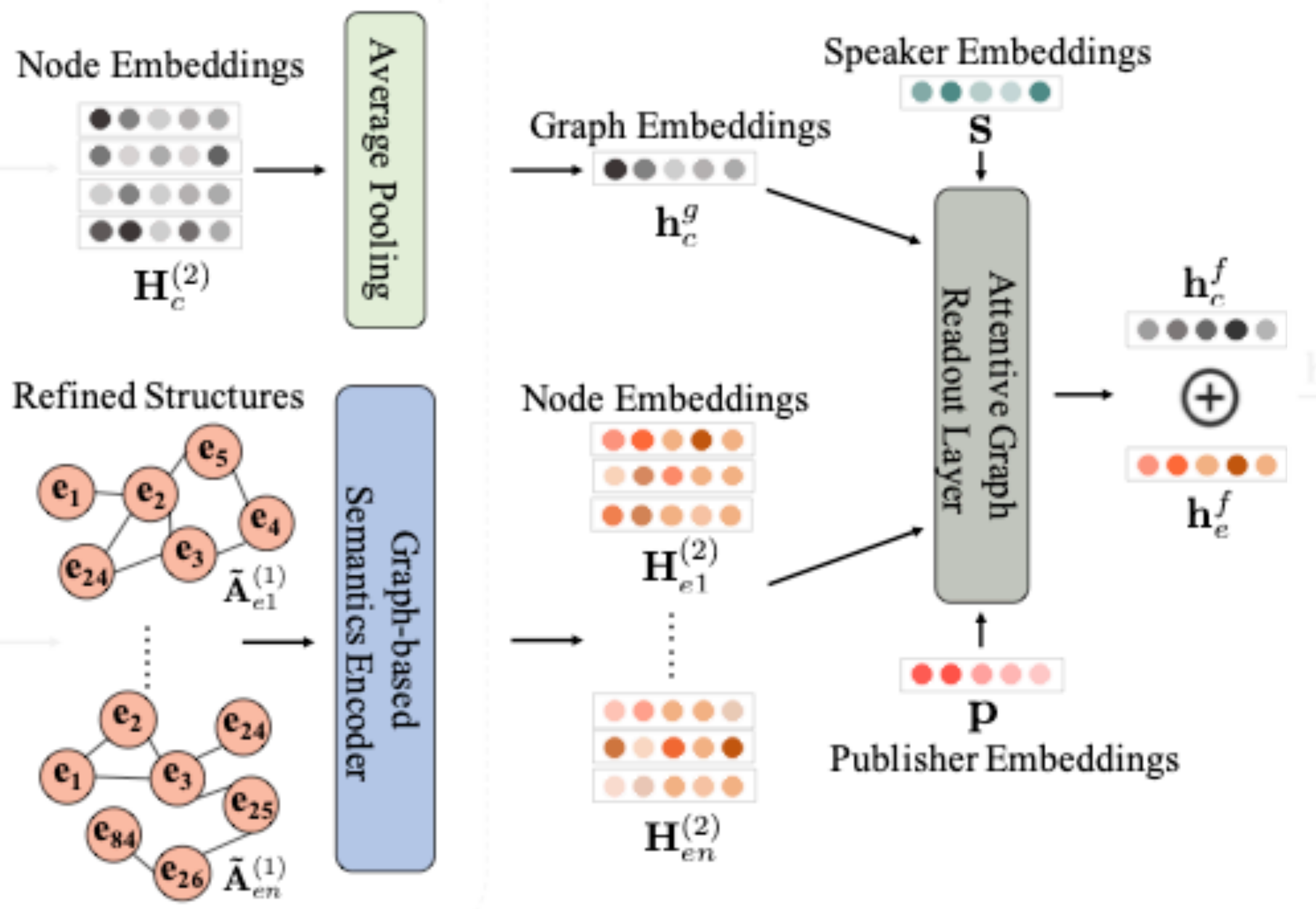
- So far, have obtained
  - refined structures for each evidence
  - fine-grained node embeddings for claims and evidences.
- Next, to perform the claim-evidence interaction, first need integrate all node embeddings into general graph embeddings.



Methodology

Attentive Graph Readout Layer

- Propose to obtain claim-aware evidence representations via attention mechanism.
- In detail, compute attention score of *j*-th word in refined evidence graph with claim representation.
- Thereafter, the evidence representation obtained via weighted summation.



Methodology

Attentive Graph Readout Layer

Length of claim

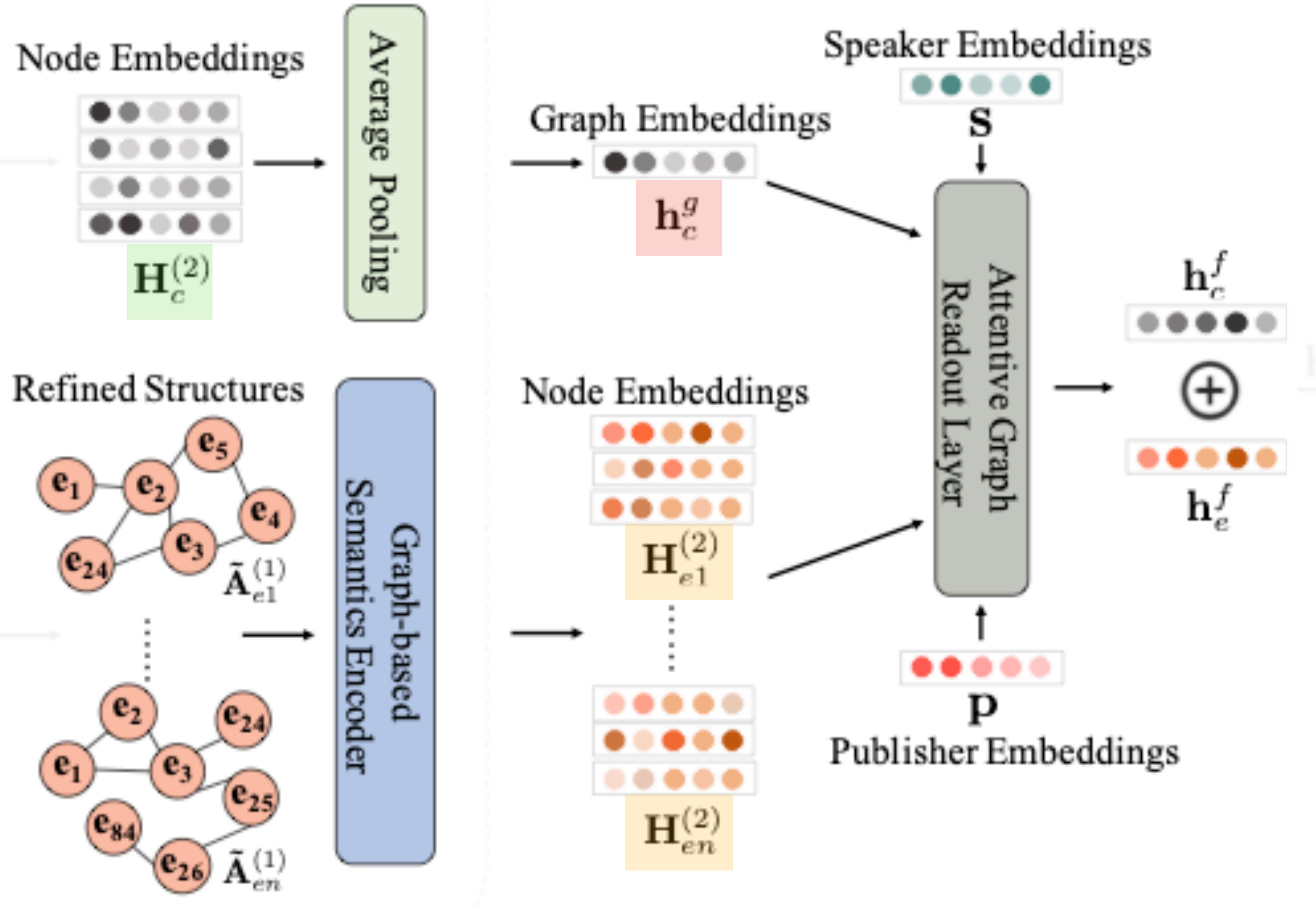
- $\mathbf{h}_c^g = \frac{1}{l_c}\sum_{i=1}^{l_c}\mathbf{H}_{ci}$

Concatenation

- $\mathbf{p}_j = \tanh([\mathbf{H}_{ej}; \mathbf{h}_c^g]\mathbf{W}_c)$
- $\alpha_j = \frac{\exp(\mathbf{p}_j\mathbf{W}_p)}{\sum_{i=1}^{l_e}\exp(\mathbf{p}_i\mathbf{W}_p)}$

Length of evidence

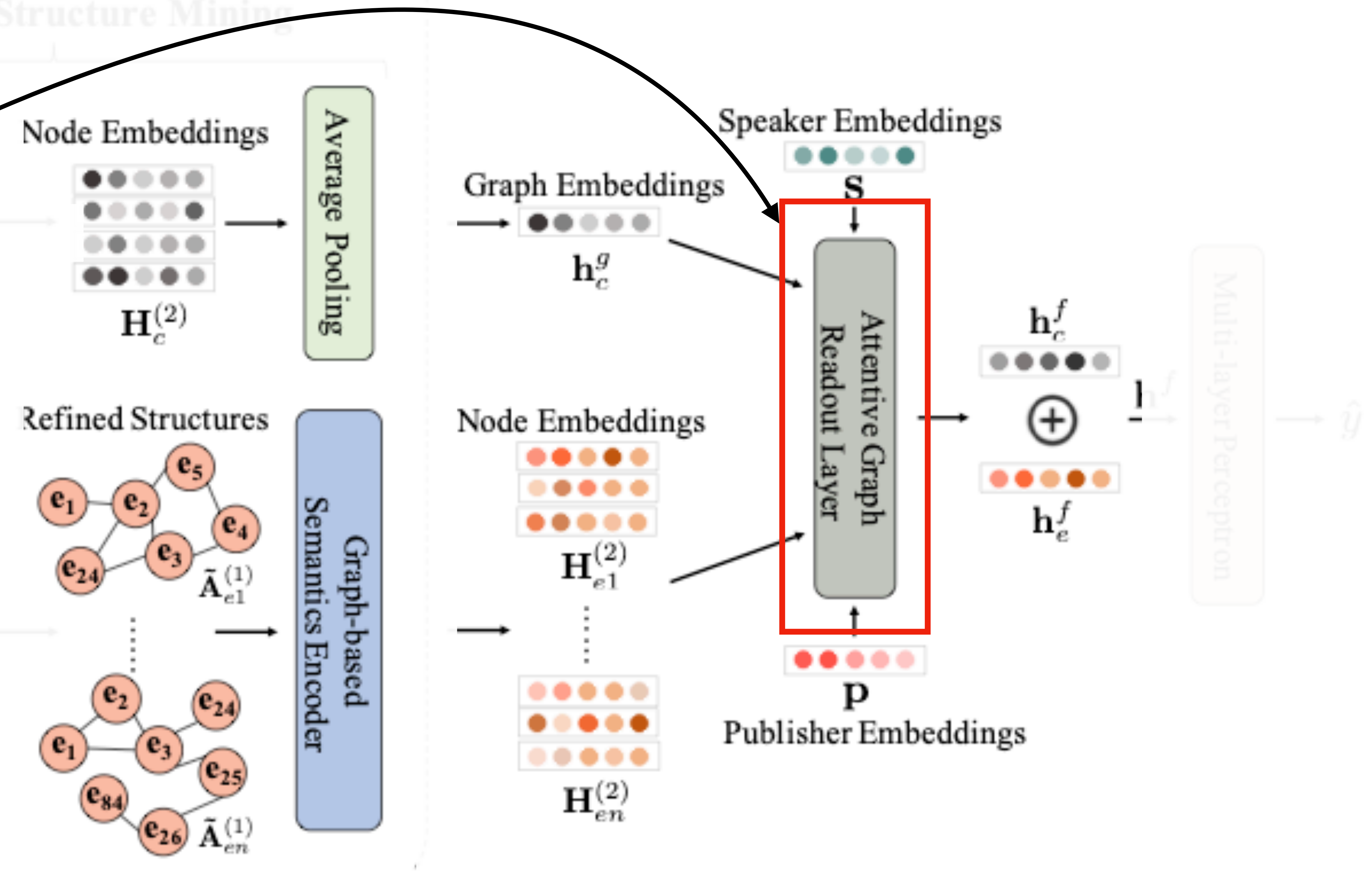
- $\mathbf{h}_e^g = \sum_{j=1}^{l_e}\alpha_j\mathbf{H}_{ej}$
- Denote as $\mathbf{ATTN}(\mathbf{H_e}, \mathbf{h}_c^g)$



Methodology

Attentive Graph Readout Layer

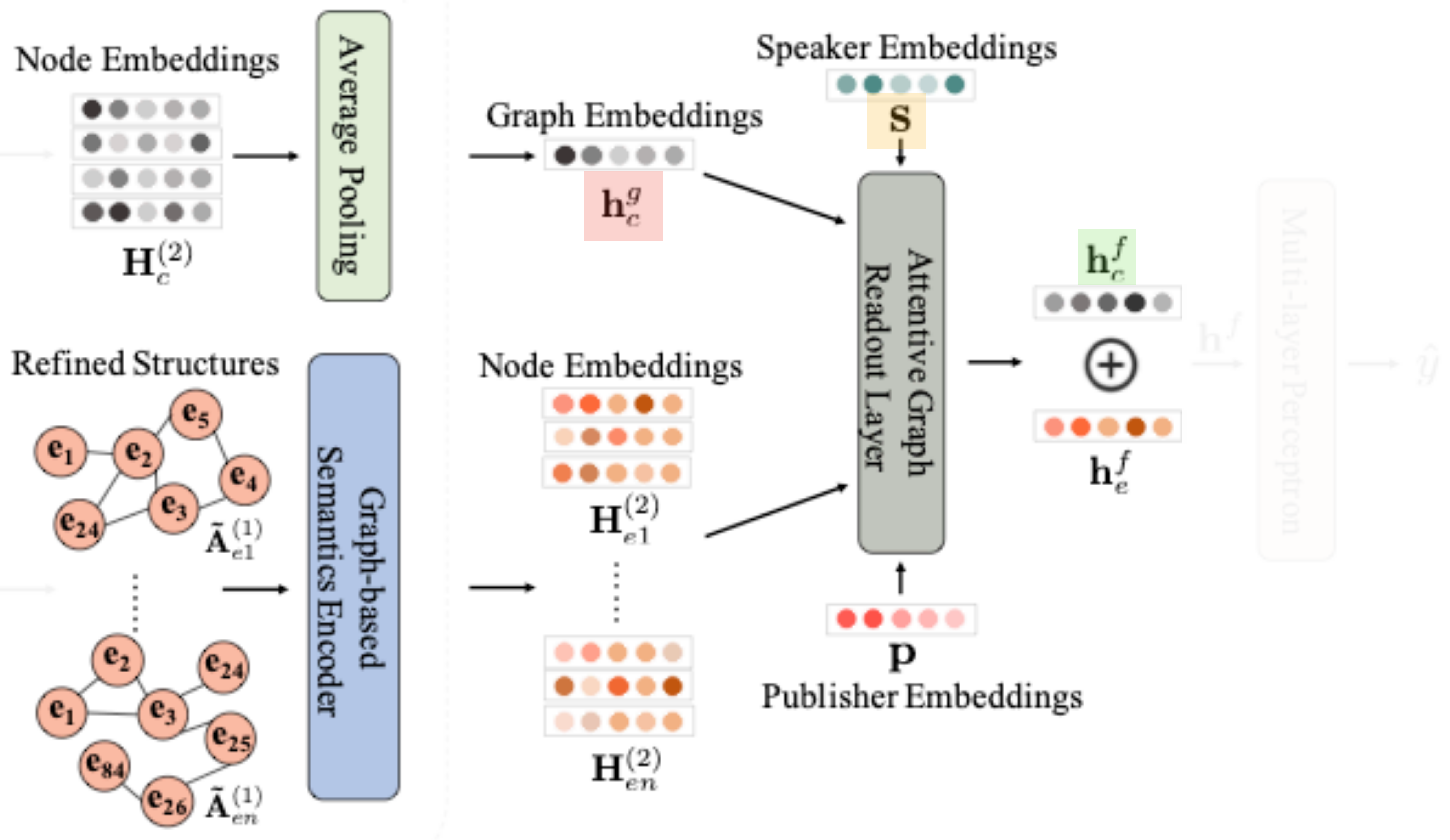
- It's worth noting that based on the fine-grained representations proposed graph-based model outputs.
- The above attention mechanism can be replaced by any interaction method in previous work.



Methodology

Attentive Graph Readout Layer

- Previous work empirically demonstrate that claim speaker and evidence publisher information is important for verification.
- Propose to extend claim and evidence representations by concatenating them with corresponding information vectors.
- $\mathbf{h}_c^f = [\mathbf{h}_c^g; \mathbf{s}], \mathbf{h}_e^r = [\mathbf{h}_e^g; \mathbf{p}]$



Methodology

Attentive Graph Readout Layer

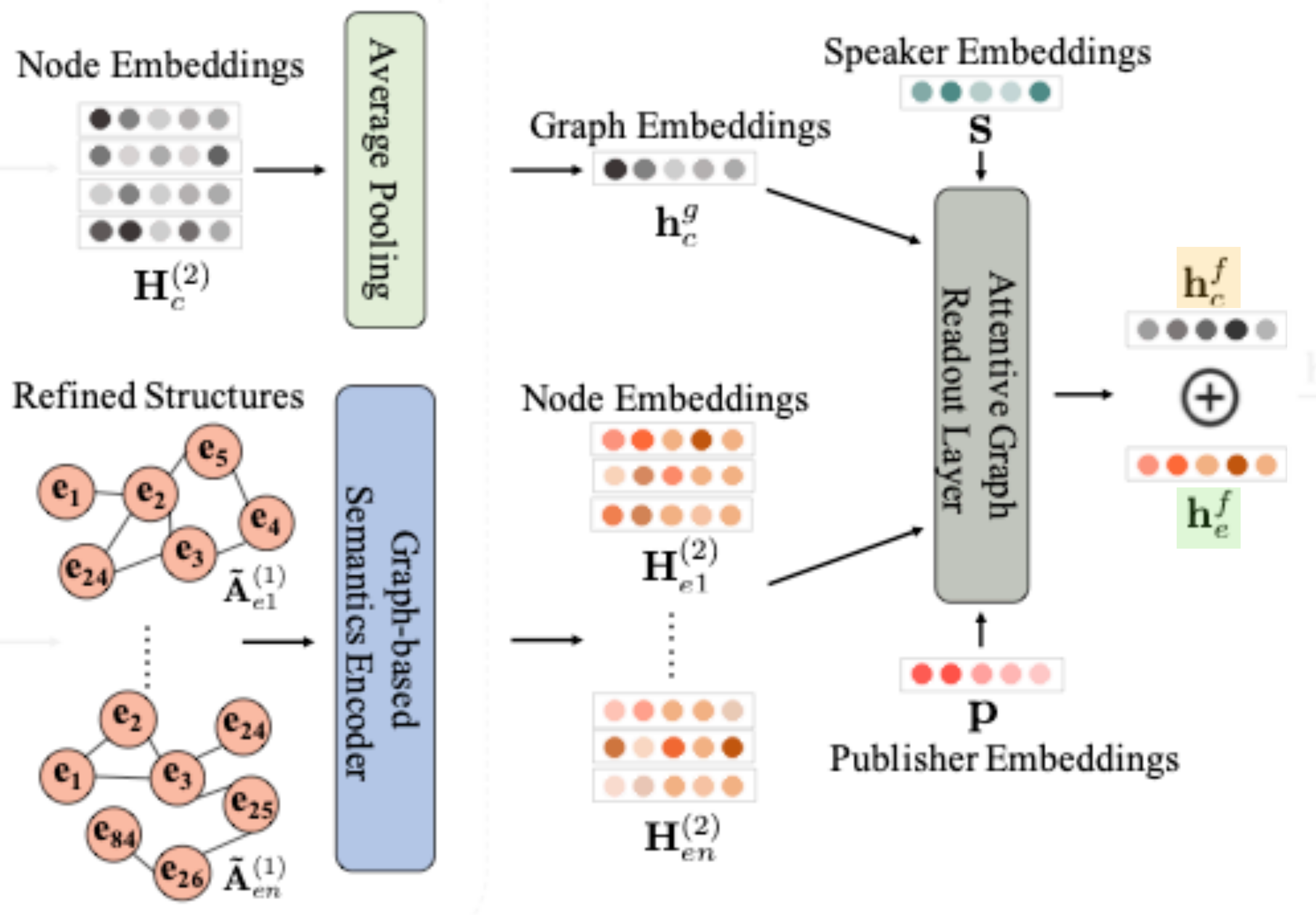
- After obtaining claim & evidence representations, further employ another attentive network.

- To capture document-level interaction between a claim and several evidences.

$$\mathbf{H}_e^r = [\mathbf{h}_{e1}^r; \mathbf{h}_{e2}^r; \ldots; \mathbf{h}_{en}^r]$$

- $$\mathbf{h}_e^f = \mathrm{ATTN}(\mathbf{H}_e^r, \mathbf{h}_c^f)$$

concatenation of embeddings of n evidences



Methodology

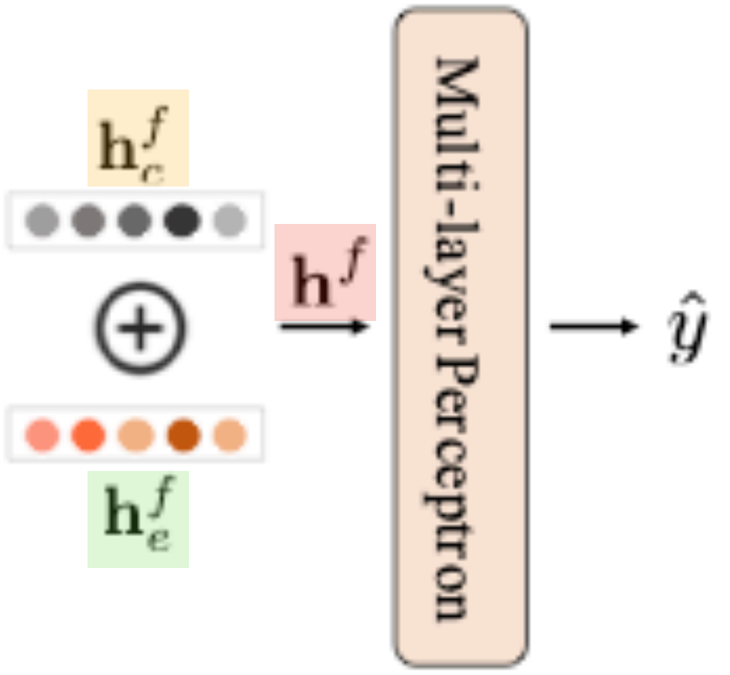
Training Objective

- Eventually, integrate claim and evidence embeddings into one unified representation via concatenation,

- Followed by a multi-layer perceptron to output the veracity prediction.

$$\mathbf{h}^f = [\mathbf{h}_c^f; \mathbf{h}_e^f]$$

- $\hat{y} = \mathrm{Softmax}(\mathbf{W}_f \mathbf{h}^f + \mathbf{b}_f)$



- Utilize the standard cross entropy loss as the objective function.

- $\mathcal{L}_\Theta(y, \hat{y}) = -(y \log \hat{y} + (1 - y)\log(1 - \hat{y}))$

Experiments

Datasets

Dataset	# True	# False	# Evi.	# Spe.	# Pub.
Snopes	1164	3177	29242	N/A	12236
PolitiFact	1867	1701	29556	664	4542

Table 1: The statistics of two datasets. The symbol "#" denotes "the number of". "True" and "False" stand for true claims and false claims, respectively. "Evi.', 'Spe.", and "Pub." denote evidences, speakers and publishers.

- Snopes
  - Claims and their corresponding labels (true or false) are collected from Snopes.
  - Taking each claim as a query, evidences and their publishers are retrieved via the search engine.
- PolitiFact
  - Claim-label pairs are collection from PolitiFact about US politics and evidences are obtained in a similar way to that in Snopes.

Experiments

Pattern-based baselines

- LSTM: utilize LSTM to encode the semantics with the news as input and obtain the final representation of claim via the average pooling.
- TextCNN: apply a 1D-convolutional network to embed the semantics of claim.
- BERT: employ BERT to learn the representation of claim. A linear layer is stacked over the special token [CLS] to output the final prediction.

Experiments

Evidence-based baselines

- DeClarE: employ BiLSTMs and followed by an attention mechanism performing among claim and each word in evidences to generate the final claim-aware representation.
- HAN: use GRUs to embed semantics and design two modules named topic coherence and semantic entailment to model the claim-evidence interaction.
- EHIAN: utilize self-attention mechanism to learn semantics and concentrate on the important part of evidences for interaction.
- MAC: hierarchical attentive framework to model both word- and evidence-level interaction.
- CICD: individual and collective cognition view-based interaction to explore both local and global opinions towards a claim.

Experiments

Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

Model Comparison

Method	Snopes								PolitiFact							
	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F
LSTM	0.621	0.719	0.430	0.484	0.397	0.812	0.791	0.837	0.606	0.609	0.618	0.632	0.613	0.593	0.590	0.604
TextCNN	0.631	0.720	0.450	0.482	0.430	0.812	0.799	0.826	0.604	0.607	0.615	0.630	0.610	0.592	0.591	0.604
BERT	0.621	0.716	0.431	0.477	0.407	0.810	0.793	0.830	0.597	0.598	0.608	0.619	0.599	0.586	0.577	0.597
DeClarE	0.725	0.786	0.594	0.610	0.579	0.857	0.852	0.863	0.653	0.652	0.675	0.667	0.683	0.631	0.637	0.625
HAN	0.752	0.802	0.636	0.625	0.647	0.868	0.876	0.861	0.661	0.660	0.679	0.676	0.682	0.643	0.650	**0.637**
EHIAN	0.784	0.828	0.684	0.617	0.768	0.885	0.882	0.890	0.676	0.679	0.689	0.686	0.693	0.655	0.675	0.636
MAC	0.786	0.833	0.687	0.700	0.686	0.886	0.886	0.887	0.672	0.673	0.718	0.675	0.735	0.643	0.676	0.617
CICD	0.789	0.837	0.691	0.632	**0.775**	0.893	0.890	0.895	0.682	0.685	0.702	**0.689**	0.714	0.657	0.691	0.629
GET	**0.800**‡	**0.846**‡	**0.705**‡	**0.721**‡	0.694	**0.895**‡	**0.890**	**0.902**‡	**0.691**‡	**0.694**‡	**0.723**‡	0.687	**0.764**‡	**0.660**‡	**0.708**‡	0.629

Table 2: The model comparison on two datasets Snopes and PolitiFact. "F1-Ma" and "Fi-Mi" denote the metrics F1-Macro and F1-Micro, respectively. "-T" represents "True News as Positive" and "-F" denotes "Fake news as Positive" in computing the precision and recall values. The best performance is highlighted in boldface. ‡ indicates that the performance improvement is significant with p-value $\leq$ 0.05.

- Proposed model GET outperforms all existing methods on most of metrics on both two datasets by a significant margin, demonstrating the effectiveness of GET.

Experiments

Model Comparison

Method	Snopes								PolitiFact							
	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F
LSTM	0.621	0.719	0.430	0.484	0.397	0.812	0.791	0.837	0.606	0.609	0.618	0.632	0.613	0.593	0.590	0.604
TextCNN	0.631	0.720	0.450	0.482	0.430	0.812	0.799	0.826	0.604	0.607	0.615	0.630	0.610	0.592	0.591	0.604
BERT	0.621	0.716	0.431	0.477	0.407	0.810	0.793	0.830	0.597	0.598	0.608	0.619	0.599	0.586	0.577	0.597
DeClarE	0.725	0.786	0.594	0.610	0.579	0.857	0.852	0.863	0.653	0.652	0.675	0.667	0.683	0.631	0.637	0.625
HAN	0.752	0.802	0.636	0.625	0.647	0.868	0.876	0.861	0.661	0.660	0.679	0.676	0.682	0.643	0.650	**0.637**
EHIAN	0.784	0.828	0.684	0.617	0.768	0.885	0.882	0.890	0.676	0.679	0.689	0.686	0.693	0.655	0.675	0.636
MAC	0.786	0.833	0.687	0.700	0.686	0.886	0.886	0.887	0.672	0.673	0.718	0.675	0.735	0.643	0.676	0.617
CICD	0.789	0.837	0.691	0.632	**0.775**	0.893	0.890	0.895	0.682	0.685	0.702	**0.689**	0.714	0.657	0.691	0.629
GET	**0.800**‡	**0.846**‡	**0.705**‡	**0.721**‡	0.694	**0.895**‡	**0.890**	**0.902**‡	**0.691**‡	**0.694**‡	**0.723**‡	0.687	**0.764**‡	**0.660**‡	**0.708**‡	0.629

Table 2: The model comparison on two datasets Snopes and PolitiFact. "F1-Ma" and "Fi-Mi" denote the metrics F1-Macro and F1-Micro, respectively. "-T" represents "True News as Positive" and "-F" denotes "Fake news as Positive" in computing the precision and recall values. The best performance is highlighted in boldface. ‡ indicates that the performance improvement is significant with p-value ≤ 0.05.

- Compared to the pattern-based methods (i.e., the first three methods in Table 2), evidence-based approaches have a substantial performance improvement.

Experiments

Model Comparison

Method	Snopes								PolitiFact							
	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F	F1-Ma	F1-Mi	F1-T	P-T	R-T	F1-F	P-F	R-F
LSTM	0.621	0.719	0.430	0.484	0.397	0.812	0.791	0.837	0.606	0.609	0.618	0.632	0.613	0.593	0.590	0.604
TextCNN	0.631	0.720	0.450	0.482	0.430	0.812	0.799	0.826	0.604	0.607	0.615	0.630	0.610	0.592	0.591	0.604
BERT	0.621	0.716	0.431	0.477	0.407	0.810	0.793	0.830	0.597	0.598	0.608	0.619	0.599	0.586	0.577	0.597
DeClarE	0.725	0.786	0.594	0.610	0.579	0.857	0.852	0.863	0.653	0.652	0.675	0.667	0.683	0.631	0.637	0.625
HAN	0.752	0.802	0.636	0.625	0.647	0.868	0.876	0.861	0.661	0.660	0.679	0.676	0.682	0.643	0.650	**0.637**
EHIAN	0.784	0.828	0.684	0.617	0.768	0.885	0.882	0.890	0.676	0.679	0.689	0.686	0.693	0.655	0.675	0.636
MAC	0.786	0.833	0.687	0.700	0.686	0.886	0.886	0.887	0.672	0.673	0.718	0.675	0.735	0.643	0.676	0.617
CICD	0.789	0.837	0.691	0.632	**0.775**	0.893	0.890	0.895	0.682	0.685	0.702	**0.689**	0.714	0.657	0.691	0.629
GET	**0.800**‡	**0.846**‡	**0.705**‡	**0.721**‡	0.694	**0.895**‡	**0.890**	**0.902**‡	**0.691**‡	**0.694**‡	**0.723**‡	0.687	**0.764**‡	**0.660**‡	**0.708**‡	0.629

Table 2: The model comparison on two datasets Snopes and PolitiFact. "F1-Ma" and "Fi-Mi" denote the metrics F1-Macro and F1-Micro, respectively. "-T" represents "True News as Positive" and "-F" denotes "Fake news as Positive" in computing the precision and recall values. The best performance is highlighted in boldface. ‡ indicates that the performance improvement is significant with p-value ≤ 0.05.

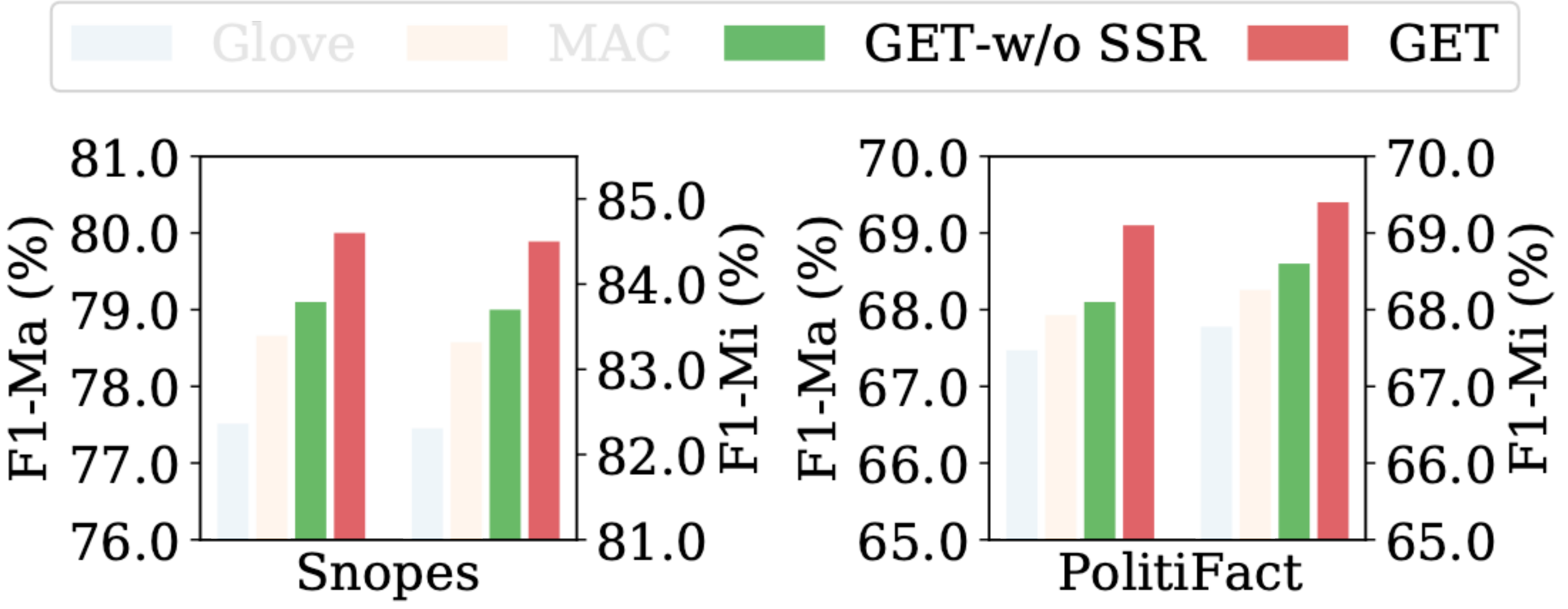
- Among five evidence-based baselines, the performance of DeClarE & HAN is inferior to other three models.

Experiments
Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

Ablation Study



- To verify the positive effect of structure refinement for reducing the useless redundancy in evidences.
  - Conduct the ablation study where the structure learning layer is removed and other parts are kept unchanged.
- Can observe obvious decline on both datasets regarding the F1-Micro and F1-Macro.
- Demonstrates the necessity of performing structure refinement on semantic graphs and confirm the effectiveness of proposed structure learning method.
- Indicates that reducing the effect of unimportant information via attention mechanisms will lead to suboptimal results, since they still maintain the noisy semantic structure unchanged.

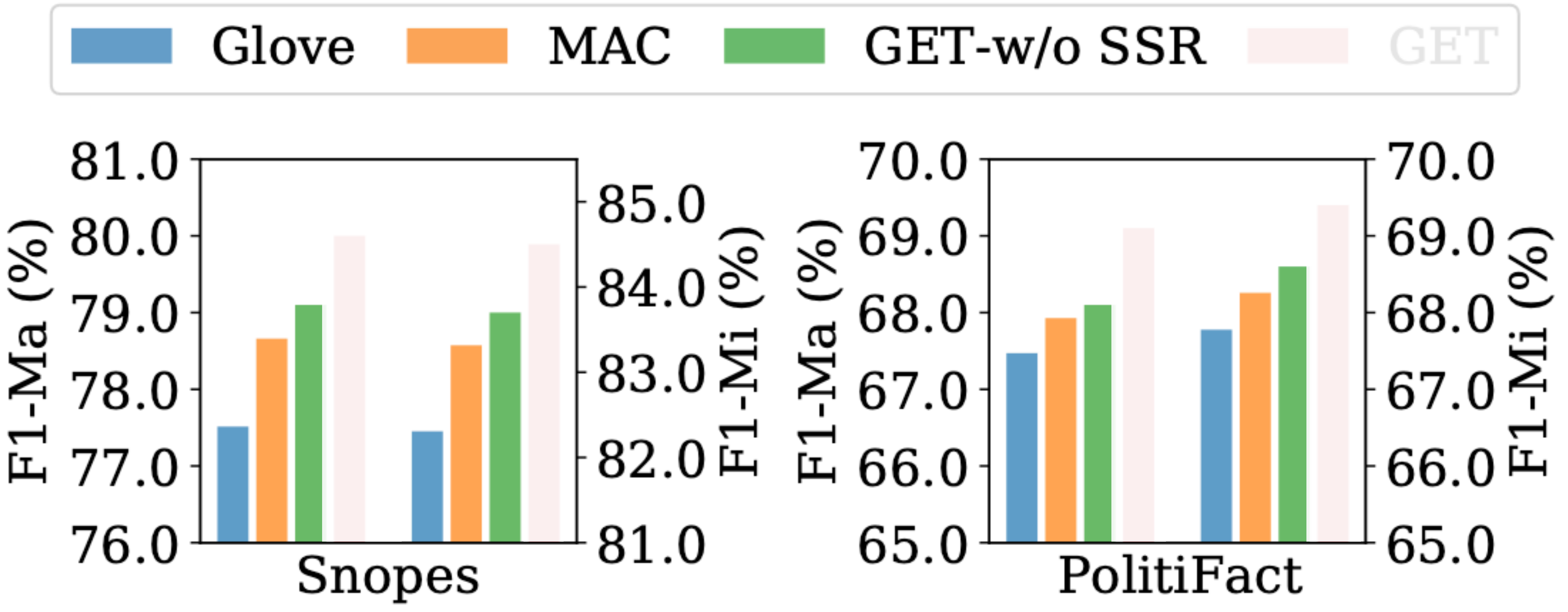
Experiments

Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

Ablation Study



- To demonstrate the superiority of proposed graph-based semantics encoder.
  - Conduct experiments on two model variants.
  - Glove: pretrained word embeddings are directly fed into the attentive readout layer.
  - MAC: semantics encoder is a BiLSTM the same as the baseline.
- Glove has the poorest performance since the contextual information is not captured.
- Moreover, GET-w/o SSR is superior to that of MAC, indicating that the long-distance structural dependency involved in semantic structure, which is less explored in sequential models, is significant for veracity checking.

Experiments
Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

GET with Different Claim-evidence Interaction Modules

Dataset	Metric	DeC	GET-DeC	EHI	wGET-EHI
Snopes	F1-Ma	0.725	**0.761**	0.784	**0.795**
	F1-Mi	0.786	**0.813**	0.828	**0.841**
	F1-T	0.594	**0.649**	0.684	**0.693**
	F1-F	0.857	**0.873**	0.885	**0.897**
PolitiFact	F1-Ma	0.653	**0.681**	0.676	**0.688**
	F1-Mi	0.652	**0.685**	0.679	**0.690**
	F1-T	0.675	**0.714**	0.689	**0.713**
	F1-F	0.631	**0.647**	0.655	**0.663**

- Semantic structure mining framework can be adaptively connected with any interaction module.
- To further verify the positive impact of graph-based structure mining, replace the concatenation attention mechanism in proposed base model with different interaction modules used in existing work.
- Choose two modules in representative work: word-level attention DeClarE, the other is self-attention to obtain global claim-evidence interactions in EHIAN.

Experiments

GET with Different Claim-evidence Interaction Modules

Dataset	Metric	DeC	GET-DeC	EHI	wGET-EHI
Snopes	F1-Ma	0.725	**0.761**	0.784	**0.795**
	F1-Mi	0.786	**0.813**	0.828	**0.841**
	F1-T	0.594	**0.649**	0.684	**0.693**
	F1-F	0.857	**0.873**	0.885	**0.897**
PolitiFact	F1-Ma	0.653	**0.681**	0.676	**0.688**
	F1-Mi	0.652	**0.685**	0.679	**0.690**
	F1-T	0.675	**0.714**	0.689	**0.713**
	F1-F	0.631	**0.647**	0.655	**0.663**

- It's obvious that GET-DeC and GET-EHI both surpass their corresponding competitors.
- Indicates the effectiveness of proposed unified graph-based semantic structure mining framework, with being agnostic to the downstream interaction modules.
- Can employ such graph-based framework in any evidence-based fake news detection model in a plug-in-play manner.

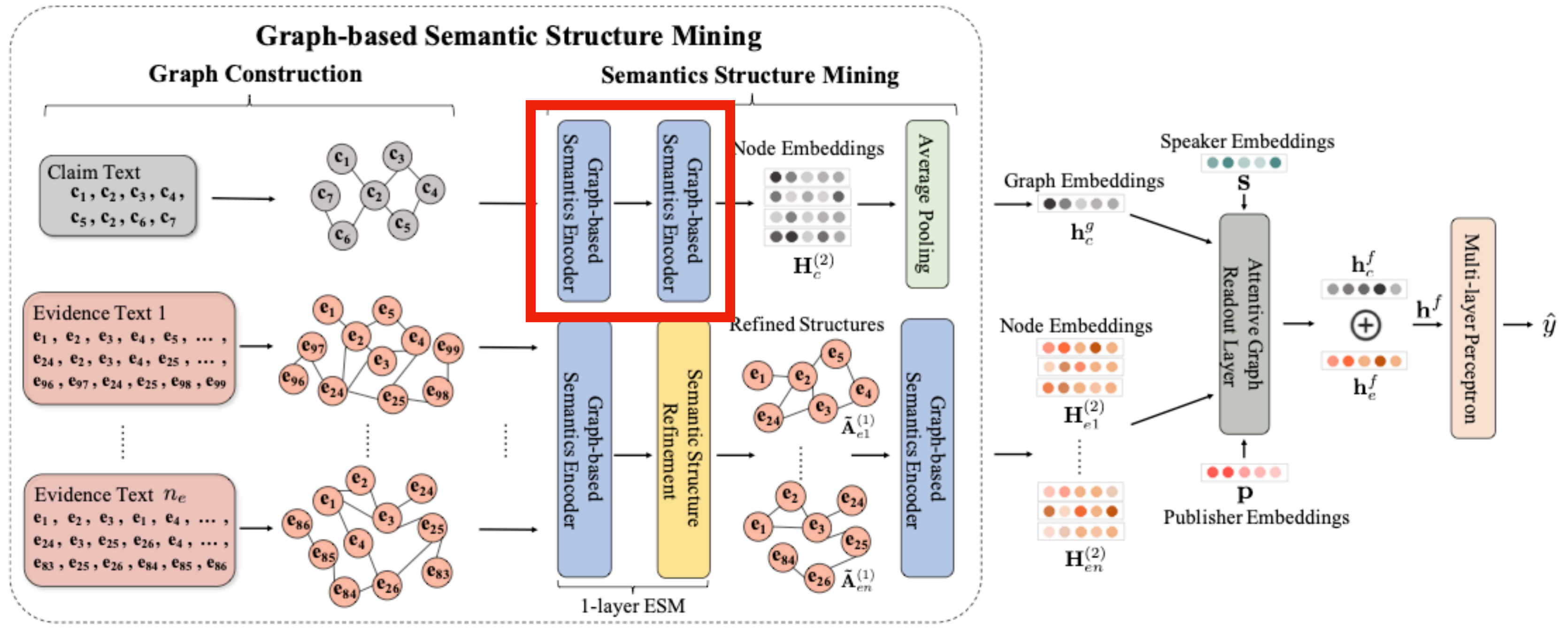
Experiments

Research Questions

- How does GET perform compared to previous fake news detection baselines?
- How does the redundant information involved in evidences affect the fake news detection?
- How is the performance of different semantic encoders?
- How does GET perform with different interaction modules?
- How does GET perform under different hyper-parameter settings?

Experiments

of semantics encoder layer for claims T_E



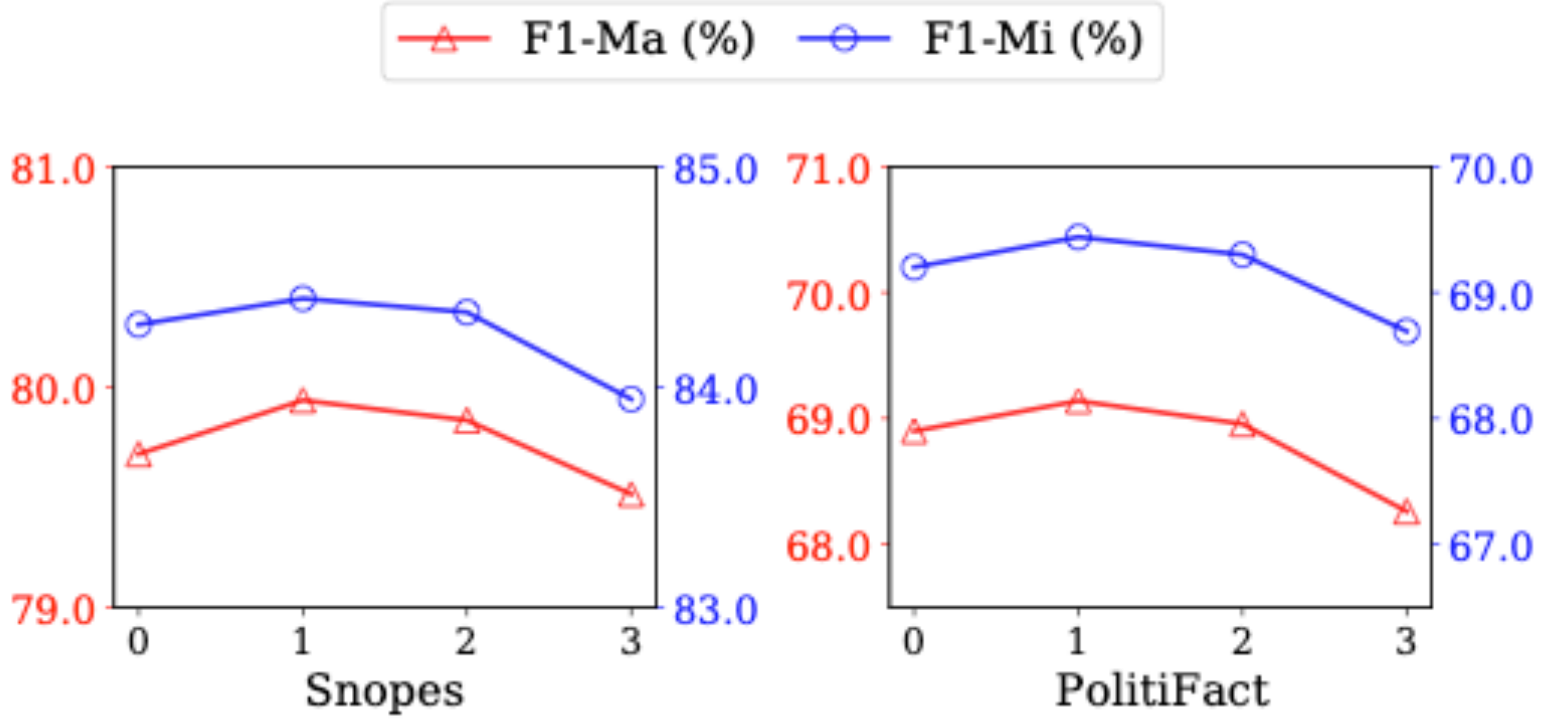


Figure 4: The influence of different semantics encoder layers T_E for claims on model performance.

- This hyper-parameter decides propagation field on graphs, since stacking T_E-layer encoder (GGNN) makes each node aggregate information within T_E-hop neighborhood.
- There is no drastic rise and fall when T_E is changed from 0 to 3.
- Specifically, the model with $T_E = 1$ slightly outperforms its counterparts.
- Only one obvious decline is observed between $T_E = 2$ and $T_E = 3$.

Experiments

Discarding rate r

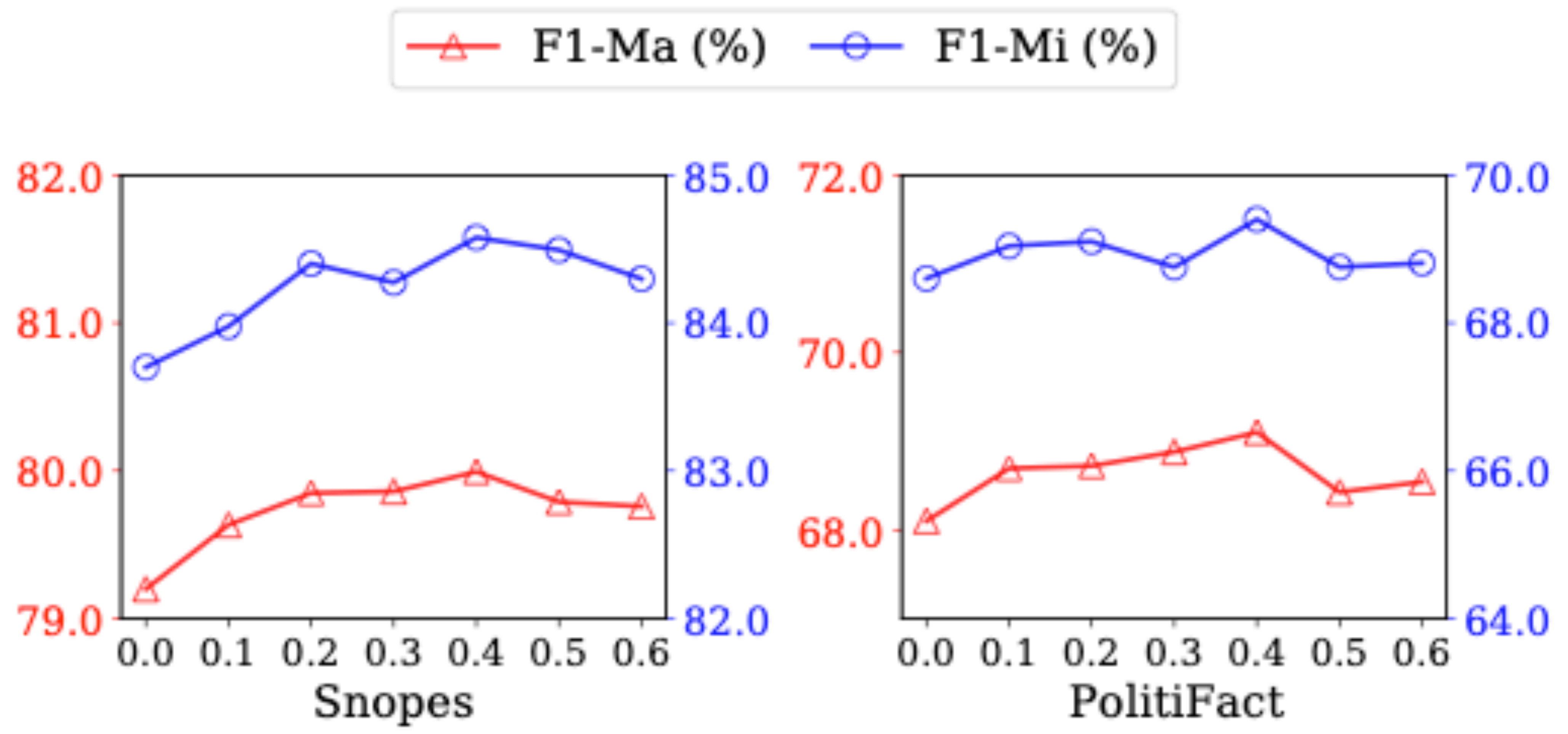
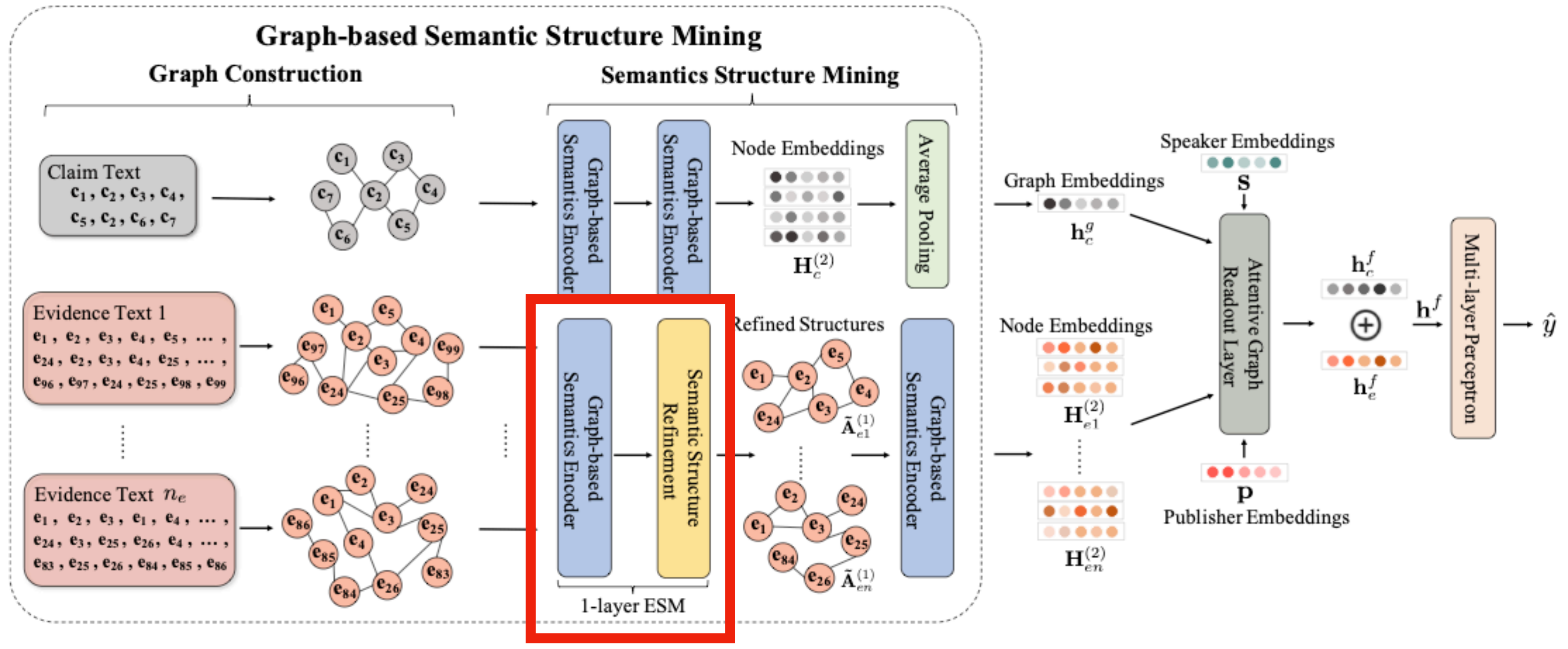


Figure 5: The influence of different discarding rates r on model performance.

- This rate decides the proportion of redundant information in evidences filter out.
- When $r = 0$, is same as GET-w/o SSR in the ablation study, where structure refinement layer is removed and no words are dropped.
- Performance grows with r increasing and peaks at the best when $r = 0.4$.
- When r is larger than 0.6, an obvious performance decline can be seen.

Experiments

of ESM layer T_R



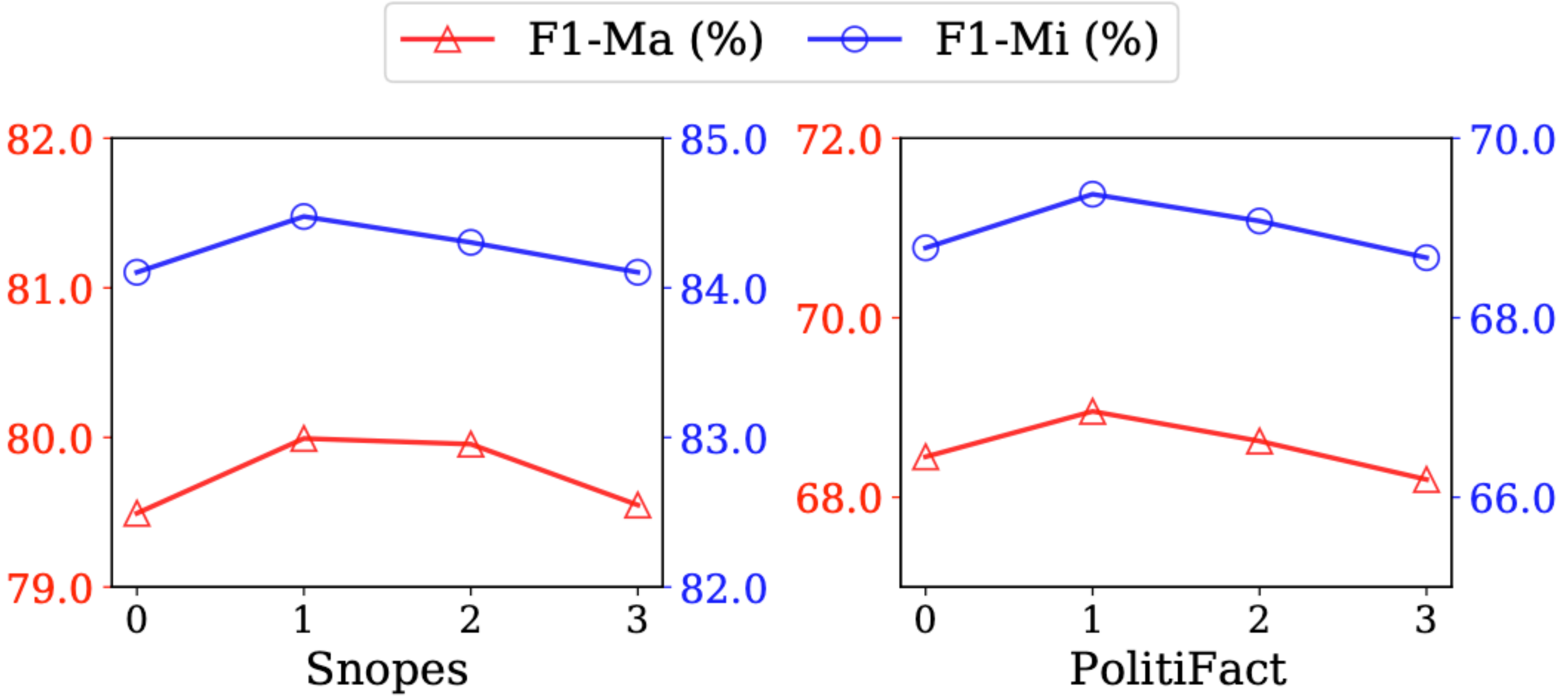
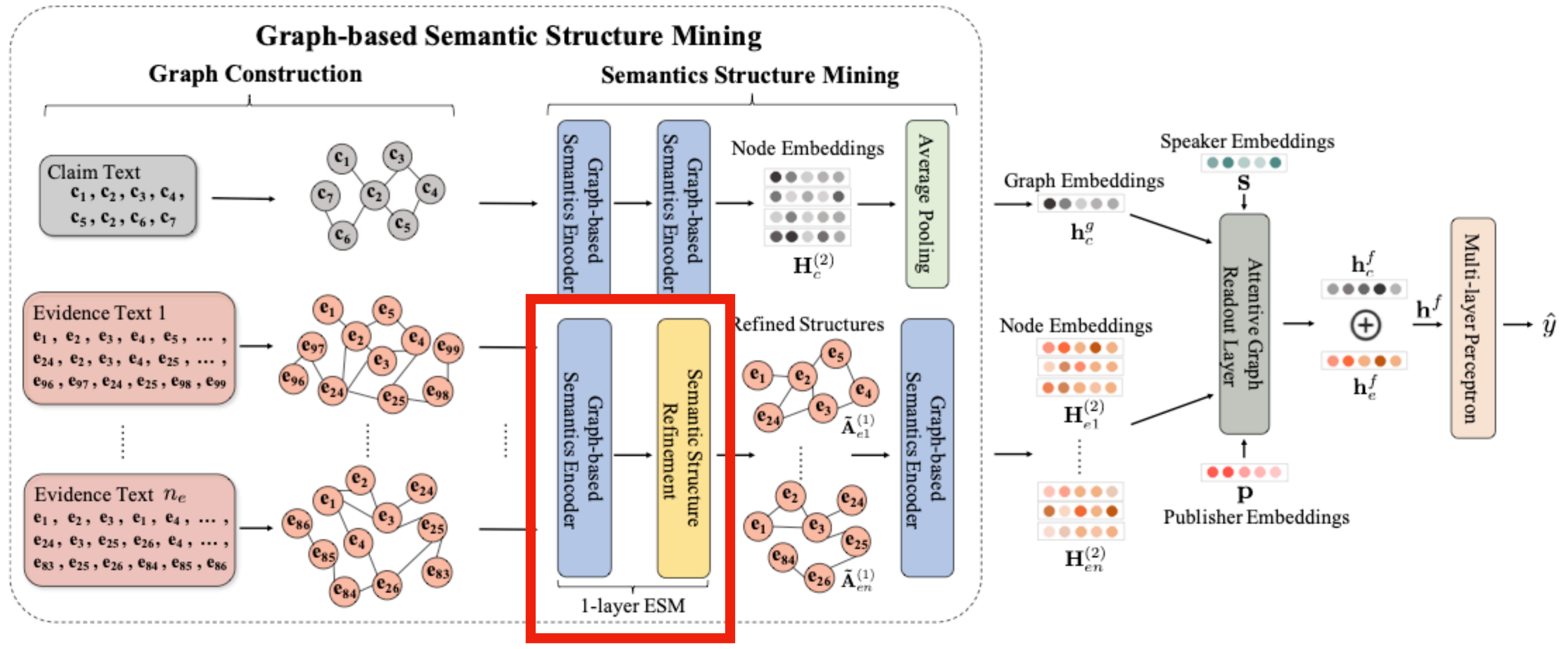


Figure 6: The influence of different evidence semantics miner layers T_R on model performance.

- This parameter controls the information propagation field on graphs and the extent of structure refinement.
- The performance is first improved from $T_R = 0$ to $T_R = 1$.

Experiments

of ESM layer T_R



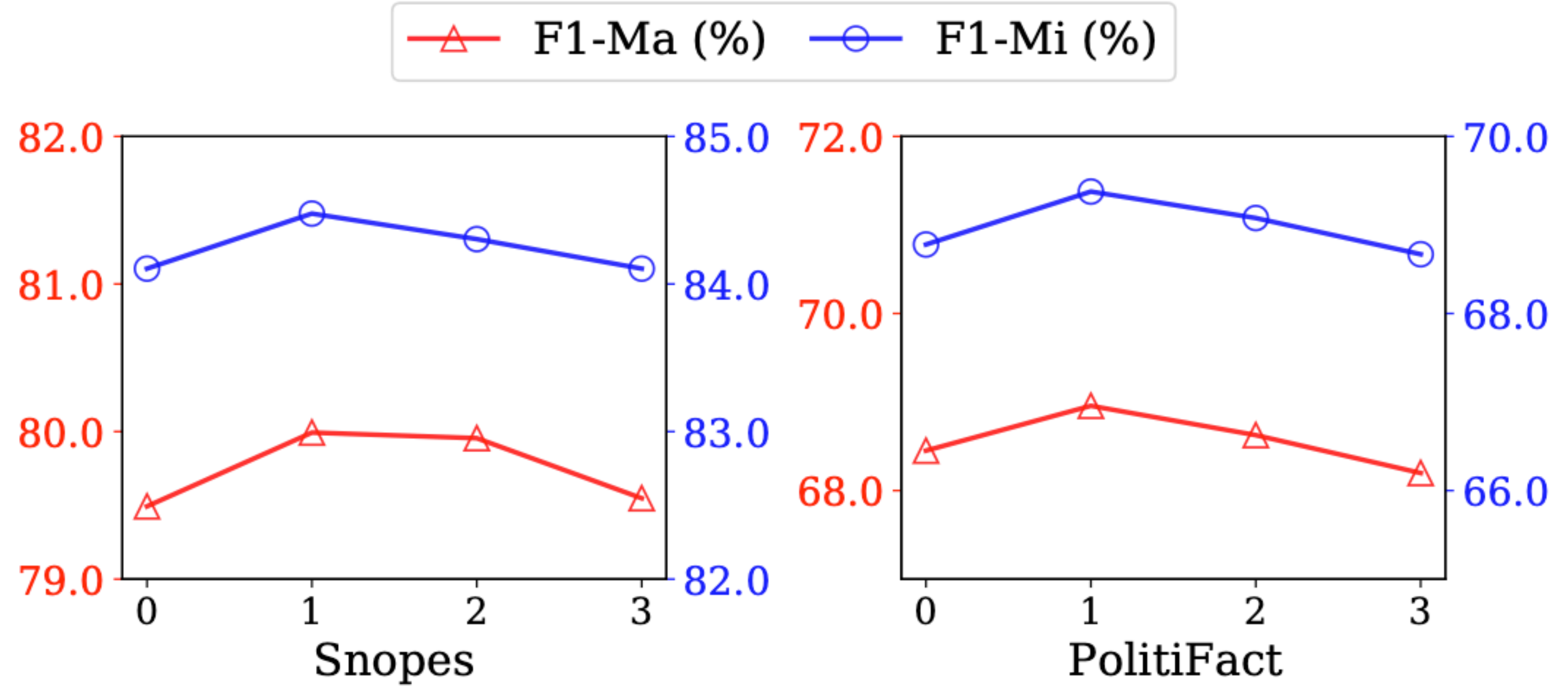
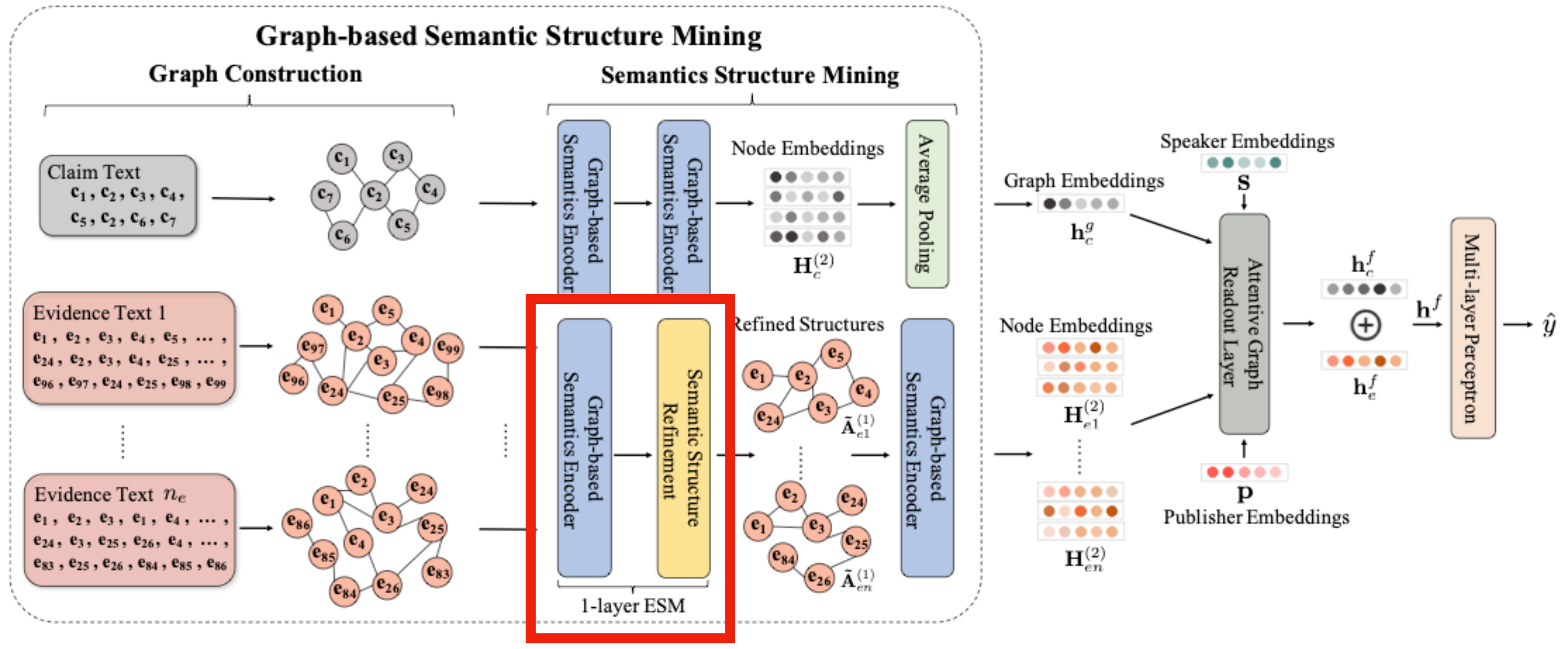


Figure 6: The influence of different evidence semantics miner layers T_R on model performance.

- When $T_R = 0$, model downgrades into the one with only a semantics encoder layer.
- Inferior performance is mainly due to two aspects:
  - It is unable to capture the high-order semantics of long evidences since only features from 1-hop neighborhood are aggregated.
  - Moreover, no redundancy reduction may affect other claim-relevant useful information, since they are fused via neighborhood propagation.

Experiments

of ESM layer T_R



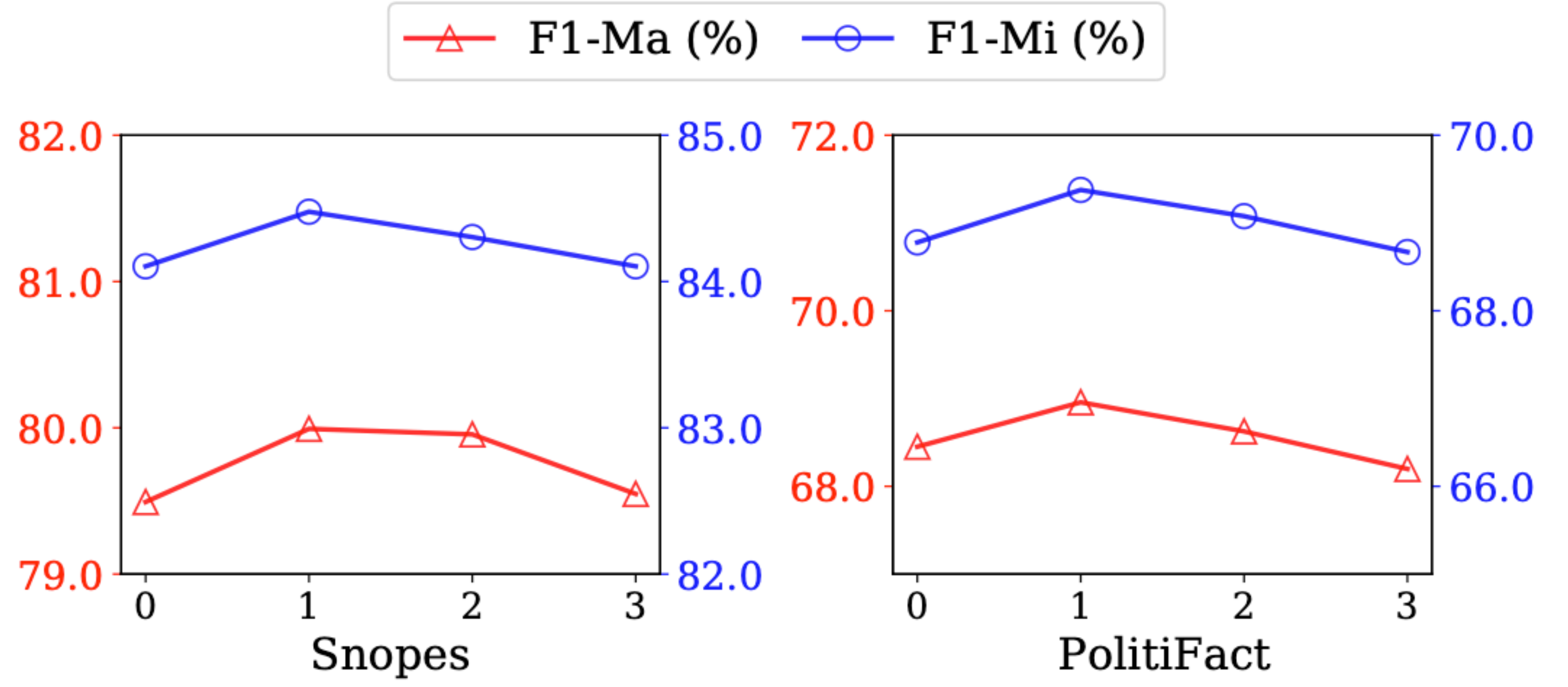


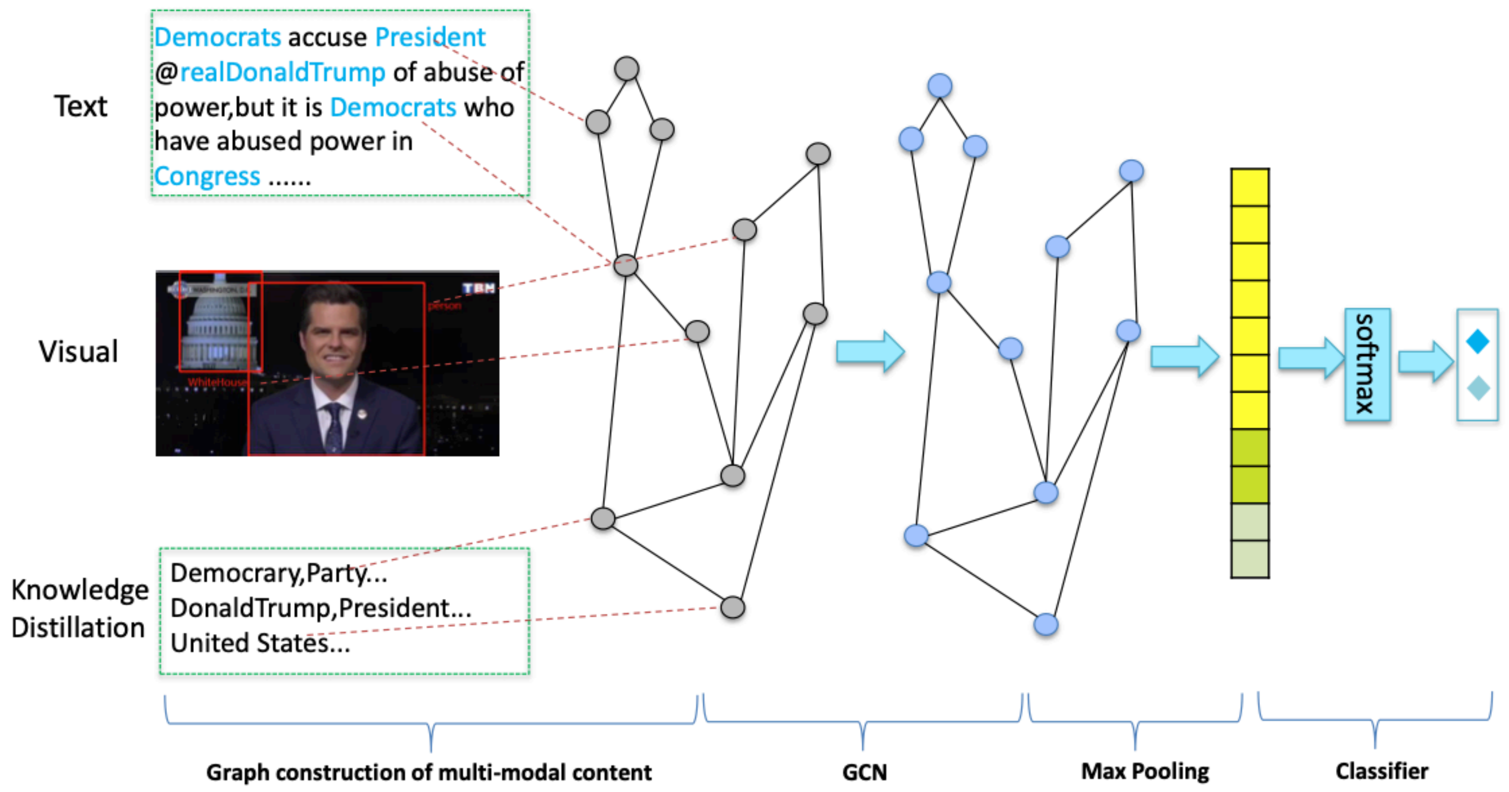
Figure 6: The influence of different evidence semantics miner layers T_R on model performance.

- A moderate fall of performance can be seen when T_R ranges from 1 to 3.
- Probably because the networks suffer from the over-smoothing problem, which is common in GNNs.
- Besides, information is overly discarded so that the evidence semantics is not well modeled, which is also a main reason.

Conclusion
of GET

- Proposed a unified graph-based FND model named GET to explore the complex semantic structure.
- Based on constructed claim and evidence graphs, the long distance semantic dependency is captured via the information propagation.
- Structure learning module is introduced to reduce the redundant information, obtaining fine-grained semantics that are more beneficial for the downstream claim-evidence interaction.

Comments
of GET



KMGCN (ICMR'20)

- Concept of modeling claim & evidences as graph like KMGCN.
- Structure refinement can effectively remove redundant information.