HiGIL: Hierarchical Graph Inference Learning for Fact Checking

Qianren Mao[†,¶], Yiming Wang[†], Chenghong Yang[†], Linfeng Du[†], Hao Peng[†], Jia Wu[•]
Jianxin Li[†,*], Zheng Wang[‡]
[†]*Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing, China*
[¶]*Zhongguancun Laboratory, Beijing, China*
[•]*Department of Computing, Macquarie University, Sydney, Australia*
[‡]*The School of Computing, University of Leeds, Leeds, U.K.*
Email: {maoqr,wangym,yangch,dulf,penghao,lijx}@act.buaa.edu.cn
jia.wu@mq.edu.au, z.wang5@leeds.ac.uk

***Abstract*—Fact-checking is vital for countering fake news. This process requires verifying the truthfulness of a claim by reasoning about multiple pieces of evidence. The current dominant approach depends upon capturing the claim-evidence relations from a claim-evidence interaction graph. Existing solutions utilize phrase-level semantics on a single-granularity but ignore other hierarchical features, such as fact- and sentence-level textual semantics and their logical topology. Since the hierarchical features often provide hints to infer collaborative high-order clues that can be essential for fact-checking, they should not be overlooked. This paper proposes a better method to model the claim-evidence graph in a multi-granularity manner. Doing so allows one to exploit more textual semantics and logical topology between a claim and its evidence. To achieve the target, we firstly employ a graph inference learning framework to infer graph nodes on different granular semantic units within their hierarchical topology. Then, an inference learning procedure is designed to optimize the global textual similarity and local topological reachability from the claim-evidence graph. We evaluate our approach by applying it to fact-checking on an open dataset and experimental results show that our technique outperforms existing graph-based techniques by a large margin.**

***Index Terms*—Fact-checking, Graph Inference Learning, Graph Reasoning, Graph Coarsening, Graph Pooling.**

I. Introduction

The prevalence of misinformation is a major concern on social media. Such inaccurate information can influence public opinions, stock prices, and even presidential elections [1]–[4]. By assessing the truthfulness of a claim, fact-checking is an important counter-measure against fake news and the spread of misinformation. The definition is given in [5], where a claim is a factual statement under investigation.

Due to the large volume of information generated, techniques for automating claim assessment are highly desired. While vital, automated fact-checking remains an open challenge. This process involves multiple steps to retrieve the documents composed of evidence sentences (document retrieval), select relevant evidence (evidence selection), and predict the truth of the claim (fact-checking). Our work focuses on the last step of claim assessment (fact-checking). Here, the goal is to label a given claim as 'SUPPORTED', 'REFUTED', or 'NOT ENOUGH INFO', indicating that the evidence can support, refute, or is insufficient for the claim respectively.

*Jianxin Li is the corresponding author.

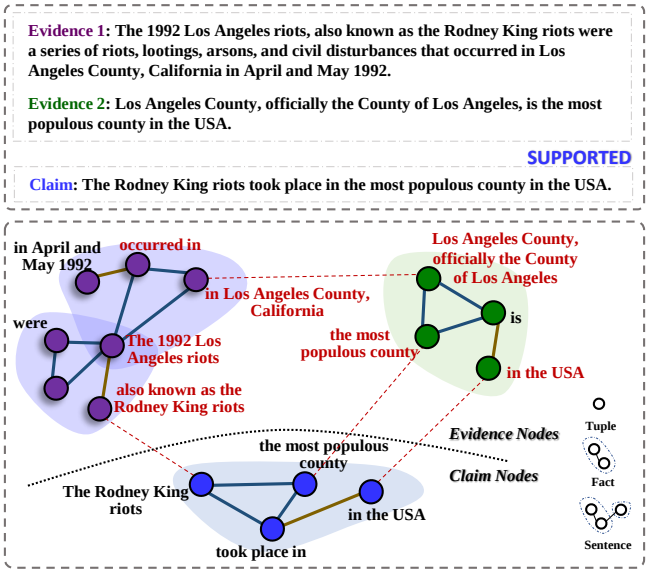


Fig. 1. Reasoning examples of FEVER data over several pieces of evidence for verification. The words (tuples) in red are the key information to verify the claim. The claim requires to reason and aggregate multiple evidence sentences in a hierarchical manner for fact verification.

Existing fact-checking methods are dominated by a particular case of recognizing textual entailment (RTE) [6] or natural language inference (NLI) [7]. They work by modeling the connections among evidence sentences by simply concatenating them [8]–[16]. Unfortunately, their strategies prevent the inference system from grasping sufficient relational and logical information among the evidences [12]. Some recent works [17], [18] use neural graph networks to aggregate evidences. Although producing promising results, they only model simple granularity graphs (e.g., either the word-level [18] or sentence-level [17], [19] graphs). Their strategies are unlikely to capture the joint relations across evidences from multi-granular information, i.e., word tuples, facts, and sentences.

This paper argues that effective fact-checking requires integrating and reasoning evidence graphs with multiple-

granularity semantic units. For example in Fig.1, a claim sentence is presented with two evidence sentences. The three sentences can be organized as three sub-graphs in which entities from the same sentence have the same node color, and the graph nodes in the graphs are the smallest unit like a word- or phrase-level tuple. These small units can form larger-grained facts. For instance, 'Los Angeles County', 'is', 'the most populous country' from Fig.1. With this breakdown, facts form the largest-grained unit of sentence. Some examples of key information are highlighted in red in Fig.1, which can provide direct clues to the claim and should not be ignored. These clues are multiple nested facts formed from word- or phrase-level tuples provided by the sentences. In other words, multiple-granularity subgraph components can contribute to fact-checking. Therefore, verifying a claim requires sorting out factual clues to understand the reasoning process over facts across multi-granularity evidence.

In light of the aforementioned observation, we aim to develop a learning framework to leverage multi-granulated facts for fact-checking. We achieve this by developing a hierarchical graph inference learning (HiGIL) model. Our model employs graph representation and graph inference learning techniques to guide a downstream model to verify facts with the minimum (e.g., word and phrase level), medium (fact level) and maximum (sentence level) granularity. It then learns to aggregate information in factual graphs to support or reject a claim based on the extracted factual information.

Our implementation utilizes StuffIE, the best-performing open information extraction tool to construct the multi-granular factual graph. This tool provides nested relations among facts and is naturally suitable for graph construction without hand-crafted rules. Our graph representation component is based on a graph convolutional network which is similar to GCN [20] or GCNII [21]. It is designed to work with the graph pooling procedure to generate hierarchical representations of graphs. The graph representation learning procedure integrates a semantic cluster from fine-grained nodes, forming the coarse-grained input for the subsequent graph convolutional layer. To support the graph model for semantics inference learning, our approach optimizes the global textual similarity and the local topological reachability to bridge the semantic relation between the claim and evidence.

We have developed a working prototype of HiGIL and applied it to the FEVER [8] dataset for fact extraction and verification. We compare our approach to a range of graph-based fact-checking methods. Experimental results show that HiGIL can effectively leverage multi-granular semantic units to improve the accuracy in verifying the truthfulness of claims. This paper makes the following contributions:

- It is the first to construct a multi-granular hierarchical factual graph with the extracted results of the best-performing information extraction tool StuffIE, achieving the target of supporting graph reasoning for claim verification.
- It shows how the techniques for graph representation learning and graph inference learning can be integrated to verify facts with minimum (e.g., word and phrase level), medium (fact level) and maximum (sentence level) granularities.
- It showcases how global textual similarity and local topological reachability of a hierarchical factual graph can be used to support graph inference learning for fact-checking.

II. Related Work

In general, fact-checking involves veracity assessment of human-generated claims by extracting evidence from Wikipedia. Existing fact-checking methods mainly formulate the task as a natural language inference (NLI) [7]. However, these NLI models [8]–[16], kind of textual inference models, mainly utilize simple evidence combination methods, such as concatenating the evidence or just dealing with standalone evidence. These methods are unable to grasp sufficient relational information in multiple dependent pieces of evidence.

The line of researches [17]–[19] proposes a graph-based reasoning approach to grasp the hierarchical logical reasoning process for fact-checking. GEAR [17] and KGAT [19] construct graphs with evidence as sentence-level nodes and use deep graph attention networks to propagate clues among neighbor nodes. DREAM [18] infers facts on a graph constructed by outputs of the semantic role labeler (SRL[1]) with phrase-level (or words tuple-level) semantic units. TARSA [12] constructs a fully-connected evidence graph and reaches a topic-aware evidence reasoning for fact verification. FACE-KEG [22] constructs a relevant knowledge graph for fact-checking from a large-scale structured knowledge base. Both graph nodes of TARSA and FACE-KEG are trained from scratch (TFS), making the training process convenient. However, these graph-based models learn only phrase-level or sentence-level representations. Thus, simply reasoning on single-granularity semantic units makes these models insufficient to capture complex or high-order clues since collaborative clues could be propagated hierarchically [23]–[25]. Moreover, both structural and semantic information plays an important role in knowledge graph reasoning [26].

III. Factual Graph Construction

Our work utilizes StuffIE[2], a fine-grained information extraction tool to construct the hierarchical factual graph. As one might observe from the StuffIE output in Fig.2B, extracted facts following numbers are formed as a triple of $\langle$*subject*; *predicate*; *object*$\rangle$ and follow the augment with a form $\langle$*connector*; *content*$\rangle$ alongside their types which represent the semantic role, such as 'DETAILS', ' CONJUCTION, (CONJCT)', 'PURPOSE, (PURP)', 'TIME', and 'POST'. All tuples in Fig.2B are formed by $\langle$A;B$\rangle$. The fact 1.14 and 1.28 are in the evidence sentence 1. The fact 2.12 and 3.5 are in the evidence sentence 2 and the claim sentence, respectively.

The pseudocode in Fig.2C shows the steps for constructing a factual-knowledge graph.

- For the fact itself, nodes are subject, object, and predicate phrases. We use the label '*sub*' or '*obj*' to link three nodes in

[1][Online].Available: https://demo.allennlp.org/semantic-role-labeling
[2][Online].Available: https://gitlab.inf.unibz.it/rprasojo/stuffie

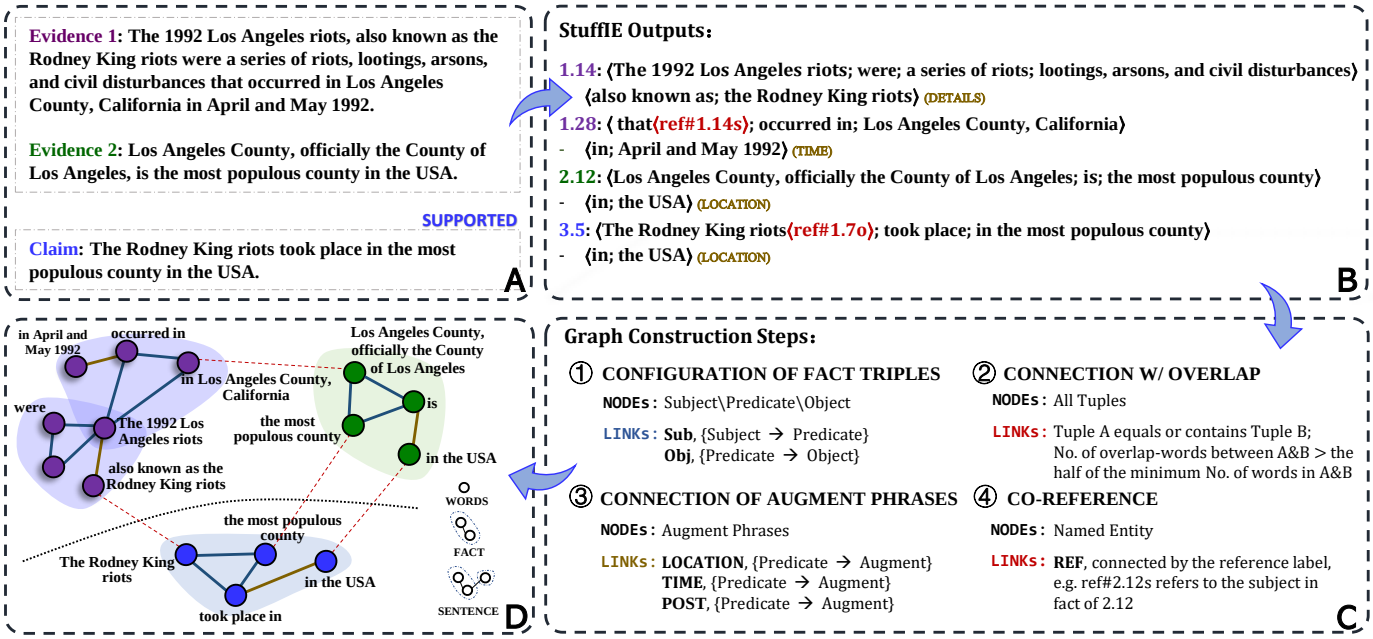


Fig. 2. The claim-evidence example and the constructed multi-granular factual graph. The StuffIE outputs are shown in subfigure B. As one might observe from the StuffIE output in subfigure B, extracted facts following numbers are formed as a triple of ⟨*subject*; *predicate*; *object*⟩ and follow the augment with a form ⟨*connector*; *content*⟩ alongside their types which represent the semantic role, such as 'DETAILS', 'LOCATION', 'TIME', and ' POST'. The pseudocode in subfigure C shows the steps for constructing a multi-granular factual-knowledge graph. Links of 'Sub' and 'Obj' are used to connect internal triples of the fact, and other Links are used to connect facts and their augments. The multi-granular graph is given in subfigure D, which has three types of nodes: tuples, facts and sentences. Nodes visualized in the same color refer to they are from the same sentences.

a fact. For the graph example given in Fig.2C, a '*sub*' label moves from the subject node of '*the embattled Tillerson*' to the predicate node of '*would last in*'. Similarly, from the predicate node of '*would last in*', the '*obj*' label moves to the object node of '*the job*'.

- For multiple tuples, we link two facts with a co-reference. As the example given in Fig.2B, $ref\#1.14s$ refers to '*that*' in the fact 1.28 and represents the subject node (*the 1992 Los Angeles riots*) in the fact 1.14. We also add edge among tuples as Zhou et al., [18] did, if one of the following conditions is satisfied: tuple A equals or contains tuple B; the number of overlapped words between A and B is larger than half of the minimum number of words in A and B.
- For the augments of a fact, we directly use SRL tags to link a fact and its augments. In our working example, the predicate node '*occurred in*' is connected to a augment node '*in April and May 1992*' by the edge of 'TIME', an SRL tag. In the end, nodes inside the claim or evidence and nodes across the claim and evidence are connected by links.

The multi-granular factual graph has naturally formed the hierarchical patterns, in which minimum-grained tuples form the medium-grained facts, and facts form the maximum-grained sentences. It should be noted that there is no guarantee that the extracted information for graph construction is error-free. The StuffIE is the best-performing tool among OpenIE tools for fact extraction. It is also a unique tool that does not require additional manual rules to obtain the nested relationship between facts and their augments [27].

IV. Proposed Method

Given the constructed graph of a claim-evidence pair (a single claim and its multiple evidence), our model is required to predict the claim's truthfulness. The basic idea of our model is to employ hierarchical graph representation learning and hierarchical graph inference learning for fact-checking. The former procedure uses the convolutional graph network (GCN [20]) to update node representations by aggregating the representations from their neighbors. It then implements graph pooling [28] to produce hierarchical representations. The latter explores feature-based and topology-based inference learning. It optimizes global textual similarity (GTSim) and local topological reachability (LTRch) on the graph for the final prediction. The two procedures are implemented on three granular graphs jointly. In doing so, we can simultaneously leverage the complementary strengths of multi-granular graph representation for hierarchical graph inference.

A. Graph Representation Learning

Formally, We denote the constructed claim-evidence graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{X}\}$, where $\mathcal{V}$ is the finite set of $|\mathcal{V}|$ nodes, and $\mathcal{E}$ defines the adjacency relationships among nodes representing the topology of $\mathcal{G}$. Taking the tuple-level graph as an example, we denote $\mathbf{X} \in \mathbb{R}^{n_p \times d}$ as a matrix containing the representation of all tuple-level nodes n_p. We initialize representations $\mathbf{X}_p$ for graph nodes using a language representation model, LRM (e.g., RoBERTa).

These representations are used as inputs to the graph convolutional module, and then will be iteratively updated

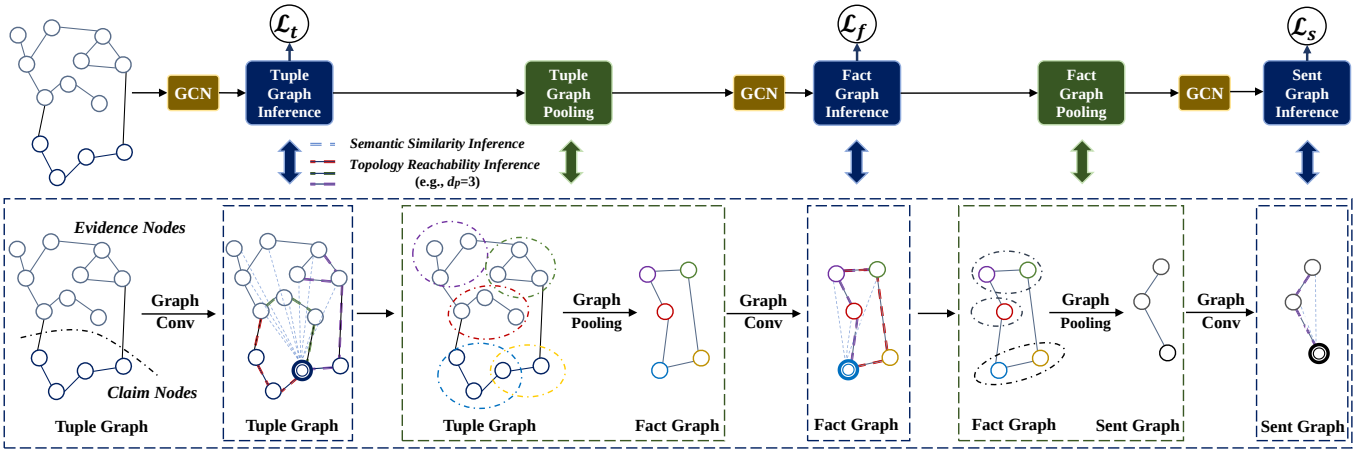


Fig. 3. Illustration of the HiGIL model.

based on the tuple-level graph topology with a GCN model. The output representation matrix $\mathbf{H}$ of the k-th GCN layer L^k is computed as:

$$\mathbf{H}_i^{(k)} = \text{Relu}\left(\widetilde{\mathbf{D}}^{-\frac{1}{2}}\widetilde{\mathbf{A}}_p\widetilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}_i^{(k-1)}\mathbf{W}_p^{(k)}\right), \quad (1)$$

where $\widetilde{\mathbf{A}}_p = \mathbf{A}_p + \mathbf{I}_p$ is the adjacency matrix of the tuple-level graph and $\mathbf{W}_p^{(k)} \in \mathbb{R}^{d_{k-1}\times d_k}$ is weight matrix of the layer. $\mathbf{H}^{(0)} = \mathbf{X}_p$ is the matrix of input representation of the model. $\mathbf{H}_i \in \mathbb{R}^d$ is the d-dimension representation of node i.

We simplify $\mathbf{H}^{(k)}$ as $\mathbf{H}$ which indicates the representation of all nodes updated by k-layer GCNs. We also use $\mathbf{H}_p = \text{GCN}(\mathbf{A}_p, \mathbf{X}_p)$, to denote an arbitrary GCN module in which $\mathbf{H}_p$ indicates the representation of all tuple nodes. Given $\mathbf{H}_p$, we apply hierarchical graph pooling (**HGPool**) to coarsen the tuple-level graph:

$$\mathbf{A}_f, \mathbf{X}_f = \text{HGPool}(\mathbf{A}_p, \mathbf{H}_p), \quad (2)$$

with generating a coarsened adjacency matrix $\mathbf{A}_f \in \mathbb{R}^{n_f\times d}$ and a matrix of new embeddings $\mathbf{X}_f \in \mathbb{R}^{n_f\times n_f}$ for the fact-level graph. Specifically, the HGPool takes the tuple embeddings $\mathbf{H}_p$ and aggregates them according to the cluster assignment matrix:

$$\mathbf{HG}_{ij} = \mathbb{1}|n_p^i \in n_f^j, \quad (3)$$

where $\mathbf{HG}_{ij}$ refers to whether the i-th node of the tuple-level graph (n_p^i) belongs to j-th cluster of the fact-level graph (n_f^j). HGPool generates hierarchical representation $\mathbf{X}_f$ for each of the n_f clusters.

Similarly, we can obtain the representation of fact $\mathbf{H}_f$ and the representation of sentence $\mathbf{H}_s$ in fact-level graph and sentence-level graph through the above equations. HGPool is heuristic since $n_p^i \in n_f^j$ is already known from the extraction results. To do so, HGPool maintains the logical topology of the constructed graph. Moreover, we explore three simple graph pooling approaches: GMaxPool, GMeanPool and GWghtPool. Take tuple-level graph coarsening as an example:

- **GMaxPool**: The graph max pool performs the *Element-wise Max* operation among fine-grained nodes $\mathbf{H} \in \mathbb{R}^{n_p\times d}$ into coarse-grained nodes $\mathbf{H} \in \mathbb{R}^{n_f\times d}$.
- **GMeanPool**: The coarse-grained nodes are obtained by the *Element-wise Mean* operation among fine-grained nodes.
- **GWghtPool**: The coarse-grained nodes are obtained by the weighted summation, $\mathbf{H} = \sum_{i=1} a_i\mathbf{h}_i \in \mathbb{R}^{n_f\times d}$. $\mathbf{h}_i$ is the representation of i-th fine-grained node, $a_i = softmax_i(\mathbf{W}_f\mathbf{h}_i)$, and $\mathbf{W}_f \in \mathbb{R}^{1\times n_f\times d}$.

Three pooling approaches are inspired by the evidence aggregation procedure employed by Zhou et al., [17] and Tymoshenko et al., [16]. However, they aggregate sentence-level evidence based on LRMs rather than graphs, which is different from our methods.

B. Graph Inference Learning

We employ hierarchical graph inference learning by combining feature-based and topology-based inference learning with the multi-granular graph. Specifically, the feature-based node inference learning to optimize global textual similarity is named as **GTSim** mechanism. The topology-based node inference learning to optimize the local topological reachability is named as **LTRch** mechanism. The two reference mechanisms are combined to generate a claim-specific evidence representation in the multi-granular graph for each node before making the final prediction.

GTSim mechanism: $f(n_p^e, n_p^c, \mathcal{R}) \Rightarrow \mathbf{Z}^{GTS}$: Here we have n_f^e and n_f^c which denote the number of evidence and claim nodes in the tuple-level graph, respectively. $\mathcal{R}$ refers to the relationship of global textual similarity between n_f^e and n_f^c. The GTSim mechanism is to obtain claim-centric evidence-aggregated representation $\mathbf{Z}^{\text{GTS}}$ for further graph inference. Let $\mathbf{H}_p^e \in \mathbb{R}^{n_p^e\times d}$ and $\mathbf{H}_p^c \in \mathbb{R}^{n_p^c\times d}$ denote matrices containing node representations in the evidence-based sub-graph and claim-based sub-graph, respectively. The GTSim optimizes global textual similarity between $\mathbf{H}_p^e$ and $\mathbf{H}_p^c$ for the graph inference learning. Specifically, the GTSim takes each $\mathbf{h}_i^c \in \mathbf{H}_p^c$ as a query, and takes all node representations $\mathbf{h}_j^e \in \mathbf{H}_p^e$ as keys.

Then, it computes normalized attention coefficient which shows the textual similarity score $s_{i\leftarrow j}^{\mathsf{GTS}}$ between the evidence node $j \in n_p^e$ and the claim node $i \in n_p^c$:

$$s_{i\leftarrow j}^{\mathsf{GTS}} = softmax_j(\text{LeakyReLU}(\vec{\mathbf{a}}^T \left[\mathbf{W}_c\mathbf{h}_i^c || \mathbf{W}_e\mathbf{h}_j^e\right])), \quad (4)$$

where $\mathbf{a}^T$ represents transposition, and $\vec{\mathbf{a}}^T \in \mathbb{R}^{2d}$ and '$||$' represent concatenation. After that, we calculate a claim-centric evidence-aggregated representation $\mathbf{Z}^{\mathsf{GTS}} = \left[z_1, ..., z_{n_p^c}\right]$ using the weighted sum over all evidence node representations $\mathbf{H}_p^e$:

$$z_i = \sum_{j \in n_p^e} s_{i\leftarrow j}^{\mathsf{GTS}} \mathbf{h}_j^e. \quad (5)$$

The $\mathbf{Z}^{\mathsf{GTS}}$ is obtained by the global textual similarity. With consideration of local topological reachability, we obtain another kind of claim-centric evidence-aggregated representation.

***LTRch mechanism**: $f(n_p^e, n_p^c, \mathcal{V}) \Rightarrow \mathbf{Z}^{LTR}$:* Here we have the topology relationships (topology edges $\mathcal{V}$) among all graph nodes. We compute the path reachability (proposed by Xu et al., [29]) from n_p^e to n_p^c by employing random walks on the tuple-level graph. Thus, the information in evidence nodes n_p^e in the sub-graph is propagated to claim nodes n_p^c in the claim sub-graph based on the reachability probability matrix.

For the claim-evidence graph $\mathcal{G}$, we define the random-walk transition matrix $\mathbf{P} = \mathbf{D}^{-1}\mathcal{E}$ where $\mathbf{D} \in \mathbb{R}^{n\times n}$ is the diagonal degree matrix. Thus, $\mathbf{P}_{i\leftarrow j}$ is the probability of moving from node $j \in n_p^e$ to node $i \in n_p^c$ in one step. We denote that $\mathbf{P}_{i\leftarrow j}^K$ is the probability of moving from node $j \in n_p^e$ to node $i \in n_p^c$ in K steps, where $\mathbf{P}_{ij}^K = \mathbf{P}_{ij}$ when $K = 1$, and if $K > 1$:

$$\mathbf{P}_{i\leftarrow j}^K = \sum_o \mathbf{P}_{i\leftarrow o}\mathbf{P}_{o\leftarrow j}^{K-1}, \quad (6)$$

where the K-step probability $\mathbf{P}_{i\leftarrow j}^K$ starts from node $j \in n_p^e$ and ends at node $i \in n_p^c$, taking a single step to any node $o \in n_p$, and then taking $K-1$ steps to node $i \in n_p^c$.

We perform the node reachability as a K-dimensional vector $\left[\mathbf{P}_{i\leftarrow j}, \mathbf{P}_{i\leftarrow j}^2, ..., \mathbf{P}_{i\leftarrow j}^K\right]$. Then, we map the reachable vector into a weight value $p_{i\leftarrow j}^{\mathsf{LTR}}$ and implement it with a MLP layer:

$$p_{i\leftarrow j}^{\mathsf{LTR}} = \text{MLP}\left[\mathbf{P}_{i\leftarrow j}, \mathbf{P}_{i\leftarrow j}^2, ..., \mathbf{P}_{i\leftarrow j}^K\right]. \quad (7)$$

After that, we calculate a claim-centric evidence-aggregated representation $\mathbf{Z}^{\mathsf{LTR}} = \left[z_1, ..., z_{n_p^c}\right]$ using the weighted sum over all evidence node representations $\mathbf{H}_p^e$ based on the local topology reachability weights $p_{i\leftarrow j}^{\mathsf{LTR}}$:

$$z_i = \sum_{j \in n_p^e} p_{i\leftarrow j}^{\mathsf{LTR}} \mathbf{h}_j^e. \quad (8)$$

We next calculate two kinds of alignment vectors, a^{GTS} and a^{LTR}, to augment node representations with aggregated features produced by the global textual similarity and the local topology reachability:

$$a_i^{\mathsf{GTS}} = f_{align}(\mathbf{h}_i^c, \mathbf{z}_i^{\mathsf{GTS}}), \quad (9)$$

$$a_i^{\mathsf{LTR}} = f_{align}(\mathbf{h}_i^c, \mathbf{z}_i^{\mathsf{LTR}}), \quad (10)$$

where $f_{align}(\cdot)$ denotes the alignment function [18], [30], in which $f_{align}(\cdot) = \mathbf{W}\left[x, y, x-y, x\cdot y\right]$ and $\mathbf{W} \in \mathbb{R}^{d\times 4*d}$.

Then, we obtain alignment matrix by concatenating two kinds of alignment vectors:

$$\mathbf{A} = \left[a_1^{\mathsf{GTS}} || a_1^{\mathsf{LTR}}, ..., a_{n_p^c}^{\mathsf{GTS}} || a_{n_p^c}^{\mathsf{LTR}}\right]. \quad (11)$$

The mean pooling obtains the final outputs over $\mathbf{A}$, and it will be fed into a MLP network for the final prediction. In the tuple graph, the cross entropy loss is:

$$\mathcal{L}_t = \text{CrossEntropy}(y^*, P(y|\mathbf{A})), \quad (12)$$

where y^* is the ground truth verification label. We can obtain another two training loss $\mathcal{L}_f$ and $\mathcal{L}_s$ on the two graphs.

The whole framework is trained end-to-end by minimizing the final loss:

$$\mathcal{L} = \mathcal{L}_t + \mathcal{L}_f + \mathcal{L}_s. \quad (13)$$

V. Experimental Analysis

A. Dataset and Evaluation Metrics

We conduct our experiments on FEVER 1.0[3], a large-scale benchmark dataset. This dataset contains 185,455 annotated claims together with links to 5,416,537 Wikipedia documents from the June 2017 Wikipedia dump. Each claim of the FEVER dataset has a human-annotated label that classifies the claims as SUPPORTS, REFUTES, or NOT ENOUGH INFO. For the Wikipedia documents, we use the processed document retrieval results [4] given by Liu et al., [19], which contain the predicted Wikipedia article titles (i.e., document IDs). These document retrieval results were extensively used in prior works to evaluate fact-checking models, such as GEAR [17] and re-produced KGAT [16]. We use the same dataset splits as the FEVER Shared Task. The data statistics are shown in TableI.

We consider two evaluation metrics, the label accuracy (LA) and FEVER score[5]. The former measures how accurately a method classifies a claim into one of the three categories: SUPPORTS, REFUTES, or NOT ENOUGH INFO, by comparing the results given by human annotators. The latter measures the percentage of correctly retrieved evidence for the SUPPORTED and REFUTED categories, and the claims labeled as NOT ENOUGH INFO do not require evidence in this case. We also submit our model to the FEVER Challenge site to report the blind test performance.

B. Salient Baseline Models

We apply graph-based approaches to learn the structural-semantic relationship of factual information using the pre-trained RoBERTa-based LRM to initialize the graph components. Specifically, we consider these salient graph-based fact-checking models, GEAR [17], KGAT [19], Trf-XH [31], DREAM [18], and TARSA [12] as our baselines. GEAR utilizes a graph attention network on a fully-connected evidence graph and aggregates all evidence through an attention layer.

[3][Online].Available: https://fever.ai/resources.html
[4][Online].Available: https://github.com/thunlp/KernelGAT/tree/master/data
[5][Online].Available: https://github.com/sheffieldnlp/fever-scorer

TABLE I
STATISTICS OF THE FEVER DATASET.

Split	TRAIN	DEV	TEST
SUPPORTED	80,035	6,666	6,666
REFUTED	29,775	6,666	6,666
NOT ENOUGH INFO	35,659	6,666	6,666
Tuple nodes (ALL)	592,715	82,771	85,440
Fact nodes (ALL)	214,713	30,185	31,250
Sent nodes (ALL)	145,449	19,998	19,998

KGAT regards sentences as the nodes of a graph and uses kernel graph attention network to aggregate information. Trf-XH models the structured text sequences by linking them with a graphical structure and reasons by multi-hop questions answering framework based on the Transfomer-XL [32]. DREAM constructs a fine-grained graph for the prediction in which each evidence sentence is parsed into tuples [6] with the off-the-shelf SRL toolkit. TARSA [12] explores a topic-aware evidence reasoning and stance-aware aggregation for fact verification, and it is trained from scratch (TFS). FACE-KEG [22] constructs a relevant knowledge graph for fact-checking from a large-scale structured knowledge base.

C. Hyperparameter Settings

We use the pre-trained LRMs (RoBERTa-base or RoBERTa-large) model to initialize the graph node components. The hyperparameters of the integrated LRM are the same as those of the corresponding pre-trained RoBERTa, including weight decay, the dimension of hidden state vectors, and the number of heads. These hyperparameters are then fine-tuned for the downstream tasks. We set the dropout rate of HiGIL as 0.1. We use 3, 2, and 1 GCN layers for the tuple, fact, and sentence graphs, respectively. For implementing GCNII, the layers of the three granular graph should be 10, 6, 2, respectively. We apply the Adam optimizer for model training with a cross-entropy loss function. The learning rate of the pre-trained LRMs is 2e-6, and the learning rate of the graph convolutional modules is 2e-3. We set the batch size as 8 for training and 32 for inference. The maximum sequence length for inputs is 256. Integrating with RoBERTa-base and RoBERTa-large, the dimension d of node representation in graph convolutional modules is set to 768 and 1024, respectively. The reachability parameter K in Eq.(6) is set to the 3, 2 and 1 in the tuple, fact and sentence graph, respectively.

D. Implementations

We implement HiGIL on PyTorch and use the pre-trained Transfomer implementation (v4.2.2) from Huggingface[7]. For graph processing, we employ DGL[8] v0.7.2. All models are trained on a single 32GB NVIDIA Tesla V100 GPU.

[6]Sentence could be parsed as multiple tuples, and a tuple is composed of several words.

[7][Online].Available: https://huggingface.co/

[8][Online].Available: https://github.com/dmlc/dgl

TABLE II
LA AND FEVER RESULTS ON THE BLIND TEST SET. WE UNDERLINE THE RESULTS OF THE BEST GRAPH-BASED BASELINES. WE ALSO HAVE MARKED THE FACT-CHECKING MODELS UNDER DIFFERENT DOCUMENT RETRIEVAL RESULTS FOR A FAIR COMPARISON. MODELS MARKED WITH ♣ INDICATE USING DOCUMENT RETRIEVAL RESULTS AND DATASET PARTITION GIVEN BY LIU ET AL., [19], THE ° INDICATES THE MODELS USE THE DOCUMENT RETRIEVAL RESULTS BY THEIR METHOD, BUT THEIR CODE/OUTPUTS ARE NOT AVAILABLE ONLINE YET.

Model	Nodes Initialization	LA	FEVER
GEAR♣ [17]	BERT-base	71.60	67.19
Trf-XH° [31]	BERT-base	72.39	69.07
KGAT♣ [19]	BERT-base	72.81	69.40
KGAT♣ [33]	CorefBERT-base	72.88	69.82
FACE-KEG° [22]	TFS	73.90	71.20
TARSA° [12]	TFS	73.97	70.70
KGAT♣ [19]	RoBERTa-large	74.07	70.38
DREAM° [18]	XLNet	<u>76.85</u>	70.60
KGAT♣ [33]	CorefBERT-large	74.37	70.86
KGAT♣ [33]	CorefRoBERTa-large	75.96	<u>72.30</u>
HiGIL♣	RoBERTa-base	**76.30**	**72.12**
HiGIL♣	RoBERTa-large	**77.05**	**73.61**

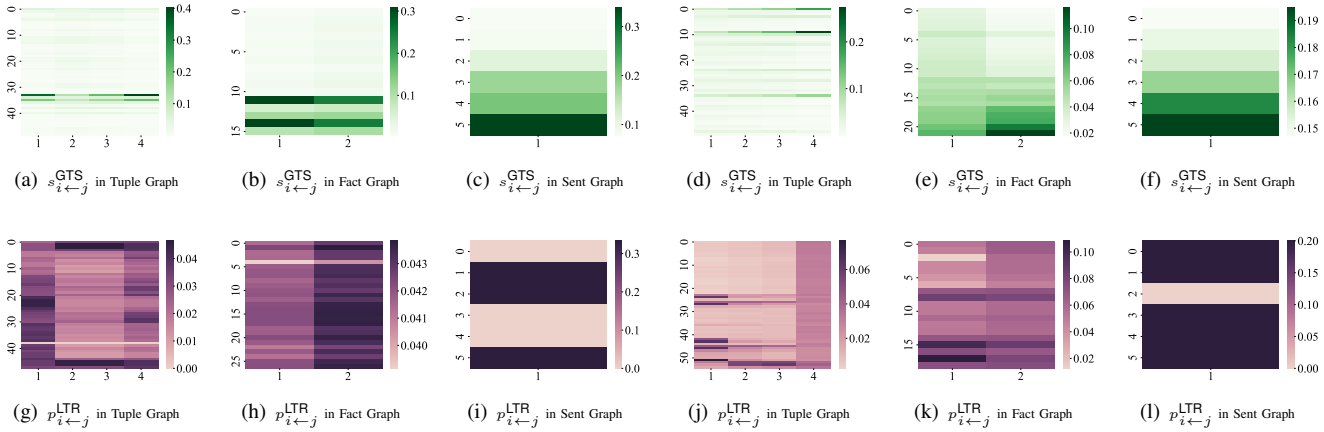
TABLE III
FACT-CHECKING PERFORMANCE OF LA AND FEVER ON THE DEV SET. WE UNDERLINE THE FACT-CHECKING RESULTS OF THE BEST GRAPH-BASED BASELINES.

Model	Nodes Initialization	LA	FEVER
GEAR♣ [17]	BERT-base	74.84	70.69
KGAT♣ [16]	BERT-base	77.80	75.64
KGAT♣ [19]	BERT-base	78.02	75.88
Trf-XH° [31]	BERT-base	78.05	74.98
KGAT♣ [19]	RoBERTa-large	78.29	76.11
DREAM° [18]	XLNet	79.16	77.01
KGAT♣ [16]	RoBERTa-base	79.98	77.66
KGAT♣ [16]	RoBERTa-large	80.77	<u>78.66</u>
TARSA° [12]	TFS	<u>81.24</u>	77.96
HiGIL♣	RoBERTa-base	**80.51**	**77.77**
HiGIL♣	RoBERTa-large	**81.34**	**78.82**

E. Main Results of Summarization

TableII reports the performance of different graph-augmented models on the blind TEST data. Our HiGIL achieves the best performance among graph-augmented models. It improves the FEVER score of 1.31 by modeling multi-granular graphs in the hierarchical reasoning process when compared with the salient graph-based KGAT [19]. Among these graph-augmented baselines, GEAR [17], Trf-XH [31], TARSA [12] and KGAT [19] are the three single-granular sentence-level graph neural models based on a fully-connected evidence graph or a hyperlink-connected evidence graph. DREAM [18] [9] employs a tuple-level graph for evidence reasoning. HiGIL outperforms these single-granular graph-augmented models by a large margin. These results show that fact-checking benefits from explicitly modeling the hierarchical structure's reasoning process among multi-granular semantic units. HiGIL is designed to provide such capabilities.

[9]DREAM is also augmented by XLNet's relative distance of words, making semantically related words have short distances.



(a) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Tuple Graph (b) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Fact Graph (c) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Sent Graph (d) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Tuple Graph (e) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Fact Graph (f) $s^{\mathsf{GTS}}_{i\leftarrow j}$ in Sent Graph

(g) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Tuple Graph (h) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Fact Graph (i) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Sent Graph (j) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Tuple Graph (k) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Fact Graph (l) $p^{\mathsf{LTR}}_{i\leftarrow j}$ in Sent Graph

Fig. 4. The attention matrix of the global textual similarity $s^{\mathsf{GTS}}_{i\leftarrow j}$ in GTSim and the local topological reachability $p^{\mathsf{LTR}}_{i\leftarrow j}$ in LTRch. The claim node id in each granular graph lies on X-axis and evidence node id lies on Y-axis. The similarity of tuple/fact/sentence graph is shown in the subfigure (a)/(b)/(c) which is related to a SUPPORTED case. The similarity of tuple/fact/sentence graph in a REFUTED case is shown in the subfigure (d)/(e)/(f). The reachability of tuple/fact/sentence graph is shown in the subfigure (g)/(h)/(i) in a SUPPORTED case and is shown in the subfigure (j)/(k)/(l) in a REFUTED case.

TABLE IV
ABLATION STUDIES OF THE PROPOSED COMPONENTS ON THE DEV SET WITH ROBERTA-BASE.

Model	LA	FEVER
HiGIL (full)	**81.34**	**78.82**
w/o GTSim	80.38 ↓0.96	77.81 ↓1.01
w/o LTRch	80.69 ↓0.65	78.11 ↓0.71
w/ Tuple graph	80.41 ↓0.93	78.02 ↓0.80
w/ Tuple graph & Fact graph	80.57 ↓0.77	78.34 ↓0.48
w/ Tuple graph & Sent graph	80.50 ↓0.84	77.96 ↓0.86

TABLE V
ADDITIONAL STUDIES OF DIFFERENT GRAPH POOLING STRATEGIES OF HIGIL WITH ROBERTA-LARGE ON THE DEV SET.

Model	LA	FEVER
w/ GMeanPool	80.73	78.32
w/ GMaxPool	81.16	78.45
w/ GWghtPool	81.34	78.82

TABLE VI
ADDITIONAL STUDIES OF DIFFERENT GRAPH POOLING STRATEGIES OF HIGIL WITH ROBERTA-LARGE ON THE DEV SET.

Model	LA	FEVER
w/ GAT	79.32	76.12
w/ GCN	81.31	78.74
w/ GCNII	81.34	78.82

In an attempt to illustrate the working mechanism of our techniques, we visualize the attention matrix of the global textual similarity $s^{\mathsf{GTS}}_{i\leftarrow j}$ of GTSim, and the local topological reachability $p^{\mathsf{LTR}}_{i\leftarrow j}$ of LTRch. The results are given in Fig.4, where subfigures a-c show the semantic similarity in tuple/fact/sentence graphs of a 'SUPPORTED' case. By observing the scale value of the color bar, we find that HiGIL can capture high similarity (e.g., some large attention score existed in the matrix) between the claim (X-axis) and evidence nodes (Y-axis). In the 'SUPPORTED' case, this phenomenon indicates that the claim and evidence are close in semantic space; hence, the claim is 'SUPPORTED' by the evidence. In contrast, the similarity score is relatively low in the 'REFUTED', as shown from subfigures d-f in each granular graph. Moreover, the path reachability matrices have a clear difference in the three graphs no matter in the SUPPORTED case or 'REFUTED' case, as shown in subfigures g-i and j-l, respectively. This distinction makes it easier for claim nodes to find important adjacent evidence nodes.

F. Ablation Study

We conduct two ablation studies on the DEV set using HiGIL with RoBERTa-large. First, we examine the effect of our proposed two graph inference mechanisms, and the results are shown in TableIV. Without the GTSim mechanism, HiGIL uses local topological reachability to bridge the connection between the claim and evidence. Without the LTRch mechanism, HiGIL optimizes the global textual similarity between the claim and evidence. Compared to the full HiGIL, label accuracy drops by 0.96 after removing the GTSim and the drop is greater than that caused by removing LTRch. Incorporating LTRch brings a 0.65 improvement in label accuracy.

Second, we explore several strategies for exploiting the performance using hierarchical graphs. Removing the sentence-level reasoning module drops 0.48 FEVER score, and removing the fact-level reasoning module drops 0.86, which indicates the importance of the fact-level graph. These results suggest that these two graph inference modules are the two most essential components. Our hierarchical graph modeling method considers more complex fact augments on multi-granular semantic units, such as semantic entailment and topical coherence in their hierarchical topology. These fact augments exploit more consolidated relations between the claim and evidence, leading to steady improvements in label accuracy and FEVER score.

As shown in TableVI, HiGIL implemented in GCN or GCNII have a slight difference in performance. However, using graph

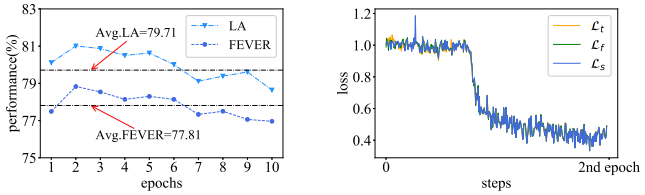
TABLE VII
A CASE STUDY OF CORRECT AND FALSE PREDICTIONS IN WHICH CLAIMS REQUIRE COMPLEX REASONING.

ID:4083 **Claim:** Bethany Hamilton's biopic was directed by Sean McNamara. **Evidence sentences:** She wrote about her experience in the 2004 autobiography Soul Surfer: A True Story of Faith, Family, and Fighting to Get Back on the Board. Soul Surfer is a 2011 American biographical drama film directed by Sean McNamara, based on the 2004 autobiography Soul Surfer: A True Story of Faith, Family, and Fighting to Get Back on the Board by Bethany Hamilton about her life as a surfer after a horrific shark attack and her recovery. **Annotated label:** SUPPORTED **Predicted label:** SUPPORTED ✓
ID:130576 **Claim:** Bruce Shand died on a ranch. **Evidence sentences:** Major Bruce Middleton Hope Shand MC and bar (22 January 1917–11 June 2006) was an officer in the British Army . He is best known as the father of Camilla, Duchess of Cornwall, the second wife of Charles, Prince of Wales. The Military Cross (MC) is the third-level military decoration awarded to officers and (since 1993) other ranks of the British Armed Forces, and used to be awarded to officers of other Commonwealth countries. Major is a military rank which is used by both the British Army and Royal Marines. The equivalent rank in the Royal Navy is lieutenant commander, and squadron leader in the Royal Air Force. **Annotated label:** NOTENOUGHINFO **Predicted label:** NOTENOUGHINFO ✓
ID: 149051 **Claim:** Margaret Thatcher avoids all involvement in politics. **Evidence sentences:** In 1975, Thatcher defeated Heath in the Conservative Party leadership election to become Leader of the Opposition and became the first woman to lead a major political party in the United Kingdom. **Annotated label:** REFUTED **Predicted label:** REFUTED ✓
ID: 3111 **Claim:** Luis Fonsi was born in the eightie. **Evidence sentences:** Luis Alfonso Rodrguez Lopez Cepero, more commonly known by his stage name Luis Fonsi, (born April 15, 1978) is a Puerto Rican singer, songwriter and actor. **Annotated label:** REFUTED **Predicted label:** SUPPORTED ✗
ID: 4414 **Claim:** The current Chief Executive Officer of Lockheed Martin is Kansas native Marillyn Hewson . **Evidence sentences:** Marillyn A. Hewson (born December 27, 1953) is chairwoman, president and chief executive officer of Lockheed Martin. Marillyn Hewson is the current President and Chief Executive Officer. Lockheed Martin is an American global aerospace, defense, security and advanced technologies company with worldwide interests. Lockheed Martin is one of the largest companies in the aerospace, defense, security, and technologies industry. Kansas is a U.S. state in the Midwestern United States . **Annotated label:** NOTENOUGHINFO **Predicted label:**SUPPORTED ✗
ID: 16306 **Claim:** Chadwick Boseman refused to ever portray a character in any Marvel Studios film. **Evidence sentences:** He will reprise his Marvel role in Black Panther, scheduled for a 2018 release as well as in Avengers: Infinity War. Black Panther is an upcoming American superhero film based on the Marvel Comics character of the same name. Produced by Marvel Studios and distributed by Walt Disney Studios Motion Pictures, it is intended to be the eighteenth film installment of the Marvel Cinematic Universe. **Annotated label:** REFUTED **Predicted label:** NOTENOUGHINFO ✗

convolutional networks is better than using graph attention networks.

G. Additional Analysis

Analysis of graph pooling in graph representation learning. The graph pooling method captures the important node information in the mapping process from a fine-grained graph to a coarse-grained graph. This experiment investigates three graph coarsening approaches: GMeanPool, GMaxPool, and GWghtPool. As shown in TableV, GWghtPool seems to be a decent method given its much better performance than other graph pooling methods.



(a) Performance trends (b) Loss Curves

Fig. 5. The performance trends w.r.t epochs on the official DEV set are shown in the subfigure (a). The loss changing trends w.r.t steps (average results of 100 steps are reported as a point) are shown in the subfigure (b).

Analysis of training procedure in graph inference learning. As shown in Fig.5, we have tried 10 epoch experiments and got an average LA score of 79.71 on the DEV set. From the loss curves on the training steps shown in subfigure (b), we find that HiGIL basically keeps convergence and obtains the best performance in the second epoch (36360 steps).

H. Case Study

Correct predictions. TableVII shows the correct prediction requires multiple nested facts to make the right inference. As shown in the case (ID:4083), to verify the SUPPORTED claim of '*Bethany Hamilton's biopic was directed by Sean McNamara*', our model needs to explore implicit clues, such as the fact '*Soul Surfer is a 2011 American biographical drama film*' and the details of the fact, such as '*film directed by Sean McNamara*', '*based on the 2004 autobiography Soul Surfer*' and '*by Bethany Hamilton about her life*'. Moreover, our HiGIL can also predict the second REFUTED case accurately.

False predictions. To better understand the limitations of our method, we have thoroughly examined 100 prediction mistakes that HiGIL fails to predict the veracity of relation labels on the DEV set. Our HiGIL is still inadequate in some complex and detailed reasoning, such as semantic understanding of numbers and reasoning of unmentioned information. In the last case, our model needs to understand that '*1978*' is not '*eightie*' and to note that '*Marillyn Hewson is Kansas native*' is not in the mentioned evidence.

Besides, we find there are some examples of possible annotation errors. As shown in the last case (ID:16306), the evidence does not mention who does '*he*' refer in the first sentence. Thus, the evidence does not directly support the claim, and the label should not be the REFUTED. Our model predicts NOTENOUGHINFO which could be a correct prediction. These cases indicate the superiority of our model, which can capitalize on the nested relationship among complex information pieces in a multi-granular manner.

VI. CONCLUSIONS

In this paper, we formally model claim-evidence graphs in a multi-granular manner for fact-checking. We design the inference learning procedure to optimize the graphs' global textual similarity and local topological reachability, so that both

local and global information in multiple pieces of evidence can be captured. Results on a large-scale FEVER dataset show that our model outperforms the existing graph-based approaches. Our future direction is to continue applying the HiGIL to other fact-checking datasets and to explore the properties of HiGIL in effectiveness and explainability.

Please also be aware of some known risks and limitations of our framework. The strategies for constructing graphs are carefully designed. However, we cannot completely avoid that important facts loss, due to the problems existed in even the best-performing information extraction tools. This is the same problem that all existing graph reasoning methods cannot avoid. To mitigate the risks and limitations and improve the real-world usability, we also welcome all kinds of improvements and enhancements from any research field by using our framework.

ACKNOWLEDGMENT

The authors are supported by the NSFC through grants (No.U20B2053).

REFERENCES

[1] R. Faris, H. Roberts, B. Etling, N. Bourassa, E. Zuckerman, and Y. Benkler, "Partisanship, propaganda, and disinformation: Online media and the 2016 us presidential election," *Berkman Klein Center Research Publication*, vol. 6, 2017.

[2] V. Soroush, R. Deb, and A. Sinan, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[3] P. Nakov and G. D. S. Martino, "Fact-checking, fake news, propaganda, and media bias: Truth seeking in the post-truth era," in *EMNLP*. Association for Computational Linguistics, 2020, pp. 7–19.

[4] G. Bekoulis, C. Papagiannopoulou, and N. Deligiannis, "A review on fact extraction and verification: The FEVER case," *CoRR*, vol. abs/2010.03001, 2020.

[5] A. Vlachos and S. Riedel, "Fact checking: Task definition and dataset construction," in *LTCSS@ACL*. Association for Computational Linguistics, 2014, pp. 18–22.

[6] I. Dagan, O. Glickman, and B. Magnini, "The PASCAL recognising textual entailment challenge," in *PASCAL*, ser. Lecture Notes in Computer Science, vol. 3944. Springer, 2005, pp. 177–190.

[7] G. Angeli and C. D. Manning, "Naturalli: Natural logic inference for common sense reasoning," in *EMNLP*. ACL, 2014, pp. 534–545.

[8] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "FEVER: a large-scale dataset for fact extraction and verification," in *NAACL-HLT, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2018, pp. 809–819.

[9] A. Hanselowski, H. Zhang, Z. Li, D. Sorokin, B. Schiller, C. Schulz, and I. Gurevych, "Ukp-athene: Multi-sentence textual entailment for claim verification," *CoRR*, vol. abs/1809.01479, 2018.

[10] C. Hidey and M. Diab, "Team SWEEPer: Joint sentence extraction and fact checking with pointer networks," in *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. Association for Computational Linguistics, 2018.

[11] T. Yoneda, J. Mitchell, J. Welbl, P. Stenetorp, and S. Riedel, "UCL machine reading group: Four factor framework for fact finding (HexaF)," in *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. Association for Computational Linguistics, 2018.

[12] J. Si, D. Zhou, T. Li, X. Shi, and Y. He, "Topic-aware evidence reasoning and stance-aware aggregation for fact verification," in *ACL/IJCNLP*. Association for Computational Linguistics, 2021, pp. 1612–1622.

[13] H. Wan, H. Chen, J. Du, W. Luo, and R. Ye, "A dqn-based approach to finding precise evidences for fact verification," in *ACL/IJCNLP 2021, Volume 1: Long Papers*. Association for Computational Linguistics, 2021, pp. 1030–1039.

[14] Y. Nie, H. Chen, and M. Bansal, "Combining fact extraction and verification with neural semantic matching networks," in *AAAI*. AAAI Press, 2019, pp. 6859–6866.

[15] N. Lee, Y. Bang, A. Madotto, and P. Fung, "Towards few-shot fact-checking via perplexity," in *NAACL-HLT*. Association for Computational Linguistics, 2021, pp. 1971–1981.

[16] K. Tymoshenko and A. Moschitti, "Strong and light baseline models for fact-checking joint inference," in *ACL/IJCNLP*. Association for Computational Linguistics, 2021, pp. 4824–4830.

[17] J. Zhou, X. Han, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "GEAR: graph-based evidence aggregating and reasoning for fact verification," in *ACL*. Association for Computational Linguistics, 2019, pp. 892–901.

[18] W. Zhong, J. Xu, D. Tang, Z. Xu, N. Duan, M. Zhou, J. Wang, and J. Yin, "Reasoning over semantic-level graph for fact checking," in *ACL*. Association for Computational Linguistics, 2020, pp. 6170–6180.

[19] Z. Liu, C. Xiong, M. Sun, and Z. Liu, "Fine-grained fact verification with kernel graph attention network," in *ACL*. Association for Computational Linguistics, 2020, pp. 7342–7351.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*. OpenReview.net, 2017.

[21] M. Chen, Z. Wei, Z. Huang, B. Ding, and Y. Li, "Simple and deep graph convolutional networks," in *ICML*. PMLR, 2020, pp. 1725–1735.

[22] N. Vedula and S. Parthasarathy, "FACE-KEG: fact checking explained using knowledge graphs," in *WSDM*. ACM, 2021, pp. 526–534.

[23] J. Li, H. Peng, Y. Cao, Y. Dou, H. Zhang, P. Yu, and L. He, "Higher-order attribute-enhancing heterogeneous graph neural networks," *IEEE Transactions on Knowledge & Data Engineering*, no. 01, pp. 1–1, 2021.

[24] L. Wu, Y. Chen, H. Ji, and B. Liu, "Deep learning on graphs for natural language processing," in *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 2021, pp. 4084–4085.

[25] Q. Sun, J. Li, H. Peng, J. Wu, Y. Ning, P. S. Yu, and L. He, "SUGAR: subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism," in *WWW '21: The Web Conference 2021*. ACM / IW3C2, 2021, pp. 2081–2091.

[26] J. Shen, C. Wang, L. Gong, and D. Song, "Joint language semantic and structure embedding for knowledge graph completion," in *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022*. International Committee on Computational Linguistics, 2022, pp. 1965–1978.

[27] Q. Mao, J. Li, H. Peng, S. He, L. Wang, P. S. Yu, and Z. Wang, "Fact-driven abstractive summarization by utilizing multi-granular multi-relational knowledge," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 30, pp. 1665–1678, 2022.

[28] Z. Ying, J. You, C. Morris, X. Ren, W. L. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *NeurIPS*, 2018, pp. 4805–4815.

[29] C. Xu, Z. Cui, X. Hong, T. Zhang, J. Yang, and W. Liu, "Graph inference learning for semi-supervised classification," in *ICLR*. OpenReview.net, 2020.

[30] D. Shen, X. Zhang, R. Henao, and L. Carin, "Improved semantic-aware network embedding with fine-grained word alignment," in *EMNLP*. Association for Computational Linguistics, 2018, pp. 1829–1838.

[31] C. Zhao, C. Xiong, C. Rosset, X. Song, P. N. Bennett, and S. Tiwary, "Transformer-xh: Multi-evidence reasoning with extra hop attention," in *ICLR*. OpenReview.net, 2020.

[32] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," in *ACL, Volume 1: Long Papers*. Association for Computational Linguistics, 2019, pp. 2978–2988.

[33] D. Ye, Y. Lin, J. Du, Z. Liu, P. Li, M. Sun, and Z. Liu, "Coreferential reasoning learning for language representation," in *EMNLP*. Association for Computational Linguistics, 2020, pp. 7170–7186.