Document-level Event Factuality Identification via Reinforced Multi-Granularity Hierarchical Attention Networks

Zhong Qian[1] , **Peifeng Li**[1,2] , **Qiaoming Zhu**[1,2*] and **Guodong Zhou**[1,2]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]AI Research Institute, Soochow University, Suzhou, China
qianzhongqz@163.com, {pfli, qmzhu, gdzhou}@suda.edu.cn

Abstract

Document-level Event Factuality Identification (DEFI) predicts the event factuality according to the current document, and mainly depends on event-related tokens and sentences. However, previous studies relied on annotated information and did not filter irrelevant and noisy texts. Therefore, this paper proposes a novel end-to-end model, i.e., Reinforced Multi-Granularity Hierarchical Attention Network (RMHAN), which can learn information at different levels of granularity from tokens and sentences hierarchically. Moreover, with hierarchical reinforcement learning, RMHAN first selects relevant and meaningful tokens, and then selects useful sentences for document-level encoding. Experimental results on DLEF-v2 corpus show that RMHAN model outperforms several state-of-the-art baselines and achieves the best performance.

1 Introduction

This paper focuses on Document-level Event Factuality Identification (DEFI), which predicts the factual nature of an event according to a document from which the event is derived, instead of a single sentence. DEFI is a crucial and fundamental task for information extraction in NLP. Therefore, we need to understand the event and document comprehensively, and pay more attentions to the tokens and sentences that are more relevant and useful with regard to the event.

As exemplified by Figure 1, the document-level factuality of event E1 is "certain negative"/CT- inferred from the document, where sentences S3, S4, S5, S6, and S7 refer to E1, i.e., contain event mentions of E1. In S4, S5 and S6, the negative cues (e.g., "denied" (S4), "fake" (S5), and "false" (S6)) and sentimental words (e.g., "disgraceful" and "offensive" (S4)) express the negative position to E1. However, some other sentences evaluate E1 as "possible positive"/PS+ (S3), "certain positive"/CT+ (S7), or do not give the factuality explicitly (S1 and S2), which may offer irrelevant or even wrong information for E1. Moreover, S8 mentions another irrelevant CT+ event "Mike Pence won election as Governor of Indiana in 2012" instead of E1, which may cause E1 to be identified as CT+ falsely. Therefore, it is essential for DEFI models to select the most relevant tokens and sentences to understand texts comprehensively and correctly for the event.

Event (E1): Mike Pence is preparing for the presidential election run in 2020.
Document-Level Event Factulaity: **CT-**
Document: (S1) On Sunday Mike Pence responded to a New York Times report that he was forming a shadow campaign as if President Trump were not involved. (S2) The report said Pence had not only kept a full political calendar but also had created his own power base. (S3) Hence, it is possible that Pence is getting ready for presidential election. (S4) But Mike Pence on Sunday denied that he is preparing for a presidential election, saying the suggestion is disgraceful and offensive. (S5) He called the article "fake news" ... (S6) "The allegations in this article are categorically false ..." Pence said. (S7) The New York Times spokeswoman claimed that, "We are confident in the accuracy of our reporting ..." (S8) Mike Pence was well known in Republican circles, and won election as Governor of Indiana in 2012.

Figure 1: An example of document-level event factuality, where speculative cues are blue, and negative cues (including negative sentimental words) are red.

Currently, most neural network EFI models focused on SEFI that predicts event factuality only considering the information within the current sentence [Saurí and Pustejovsky, 2012; Qian *et al.*, 2018a; Qian *et al.*, 2018b; Veyseh *et al.*, 2019; Cao *et al.*, 2020]. While DEFI is in the early stage, and previous models extracted knowledge from syntactic paths and sentences by LSTM [Qian *et al.*, 2019; Huang *et al.*, 2019] or modeled sentences via graph network [Cao *et al.*, 2021]. However, they relied on annotated information, e.g., event triggers, speculative and negative cues. Furthermore, they considered all the texts of documents and ignored to discard noise that may lead to wrong results, where noise refers to the irrelevant tokens and sentences.

According to above analysis, the challenges of DEFI mainly lie in those aspects: 1) End-to-end modelling, i.e., only using events and documents without any other explicitly annotated information; 2) Text selection, i.e., selecting the

*Corresponding Author

most relevant and useful tokens and sentences; 3) Comprehensive understanding, i.e., learning inherent and interactive semantics from events and documents effectively.

To solve these issues, we propose a novel end-to-end DEFI model, i.e., **R**einforced **M**ulti-Granularity **H**ierarchical **A**ttention **N**etwork (**RMHAN**), which integrates both text selection and comprehensive understanding for events and documents. Our contributions can be summarized as follows:

- RMHAN, to our best knowledge, is the first end-to-end DEFI model that does not rely on other annotated information (except for the event and document);
- RMHAN can learn semantical information of events at different levels of granularity, including sentences, topics, events, and documents, and considers both sentence and document-level encoding hierarchically;
- RMHAN integrates policy networks for both token and sentence selection with hierarchical reinforcement learning (HRL), and is the first DEFI model using HRL;
- The empirical results on DLEF-v2 corpus (both English and Chinese sub-corpora) demonstrate that RMHAN is superior to other state-of-the-art baselines.

2 Approach

2.1 Overview

The overall architecture of RMHAN is shown in Figure 2, which is composed of three sub-networks: 1) A Multi-Granularity Hierarchical Attention Network (MHAN) ϕ^c performing as the classification network, and produces rewards for policy networks; 2) A token selection policy network TS-PNet/π^t; 3) A sentence selection policy network SS-PNet/π^s.

Hence, RMHAN mainly enables these attributes to tackle the challenges in §1: 1) **R**einforced, i.e., hierarchical reinforcement learning for both token and sentence selection; 2) **M**ulti-Granularity, i.e., capturing semantics at different levels of granularity from texts; 3) **H**ierarchical, i.e., hierarchical structure for encoding the sentences and document; 4) **A**ttention, i.e., employing various attention networks to model the document.

In terms of **A**ttention, we consider Self-Attention (SA) [Vaswani *et al.*, 2017], bi-directional Co-Attention (CA) [Lu *et al.*, 2016; Xiong *et al.*, 2017], and Vanilla Attention (VA) pooling. Formally, SA is defined as:

$$\mathrm{SA}(\boldsymbol{U}) = \mathrm{FFN}(\mathrm{MHA}(\boldsymbol{U}, \boldsymbol{U}, \boldsymbol{U})) \tag{1}$$

where MHA is Multi-Head Attention, and FFN is Feed-Forward Network. While CA is computed as:

$$\hat{\boldsymbol{V}} = \mathrm{MHA}(\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{U}) \tag{2}$$

$$\mathrm{CA}(\boldsymbol{U}, \boldsymbol{V}) = \mathrm{FFN}(\mathrm{MHA}(\boldsymbol{U}, \hat{\boldsymbol{V}}, \hat{\boldsymbol{V}})) \tag{3}$$

To learn the vector representation of a matrix, VA is applied in our model:

$$\boldsymbol{\alpha} = \mathrm{softmax}(\boldsymbol{u}_s^{\mathsf{T}} \tanh(\boldsymbol{U})) \tag{4}$$

$$\mathrm{VA}(\boldsymbol{U}) = \boldsymbol{U}\boldsymbol{\alpha}^{\mathsf{T}} \tag{5}$$

where $\boldsymbol{u}_s$ is the parameter. Since RMHAN contains stacks of several sub-encoders defined in §2.2 and §2.3, we consider

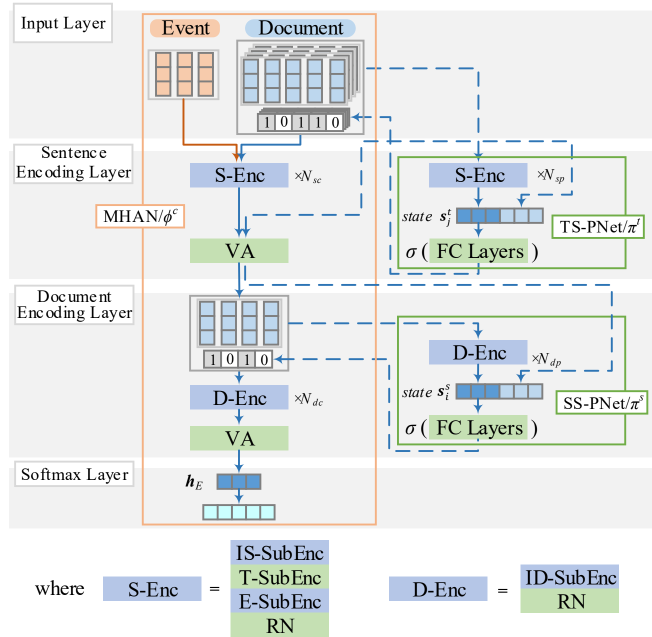

Figure 2: Overall architecture of Reinforced Multi-Granularity Hierarchical Attention Network (RMHAN) for end-to-end DEFI task.

Residual Networks (RN) to avoid the degradation of deep networks:

$$\mathrm{RN}'(\boldsymbol{U}) = \mathrm{LN}(\mathrm{FC}^{\mathrm{r}}(\boldsymbol{U}) + \boldsymbol{U}) \tag{6}$$

$$\mathrm{RN}(\boldsymbol{U}, \boldsymbol{V}) = \mathrm{LN}(\mathrm{RN}'(\boldsymbol{U}) + \boldsymbol{V}) \tag{7}$$

where LN is a normalization layer, and FC^{r} is a stack of fully connected layers.

In summary, the encoding procedures of RMHAN include: 1) selecting tokens from each sentence $\mathbb{S}_i$ by TS-PNet; 2) encoding sentences w.r.t the current event by the sentence encoding layer in MHAN; 3) selecting sentences from the document $\mathbb{D}$ by SS-PNet; 4) encoding the document by the document encoding layer in MHAN.

Firstly, we get the embedding of each token t_j^i in $\mathbb{S}_i$ and $\mathbb{E}$ by the sum of word and position embedding [Vaswani *et al.*, 2017], i.e., $\boldsymbol{S}_i^{(0)}$ and $\boldsymbol{E}^{(0)}$, and feed them into the following layers in MHAN.

2.2 Sentence Encoding Layer

This layer learns representations of each sentence $\mathbb{S}_i$, mainly containing a stack of sentence encoders, a token selection policy network, and a VA pooling layer, as defined below.

Sentence Encoders (S-Enc)

Concretely, an S-Enc consists of the following components:

Intra-Sentence Sub-Encoder (IS-SubEnc). Based on self-attention, IS-SubEnc learns semantic information within sentences, i.e.,

$$\boldsymbol{S}_i^{(1)} = \mathrm{SA}(\boldsymbol{S}_i^{(0)}) \tag{8}$$

$$\boldsymbol{E}^{(1)} = \mathrm{SA}(\boldsymbol{E}_i) \tag{9}$$

Topic Sub-Encoder (T-SubEnc). The first sentences in the document $\mathbb{D}$ are selected as topics for further encoding. The main motivation is that topics usually contain the main semantics of documents and more related information w.r.t the current event $\mathbb{E}$ than other sentences, e.g., tense of events, event triggers and arguments. Therefore, topics can offer beneficial supplementary semantics for the event. Just as illustrated in Figure 1, S1 involves several event arguments, e.g., named entities "Mike Pence" and "New York Times" that hold different document-level factuality of E1. Therefore, we integrate the topic into other sentences by CA to learn interactive semantics as below (where $i > 0$):

$$\boldsymbol{S}_i^{(2)} = \mathrm{CA}(\boldsymbol{S}_i^{(1)}, \boldsymbol{S}_0^{(1)}) \tag{10}$$

Event Sub-Encoder (E-SubEnc). According to the definition of DEFI, the event $\mathbb{E}$ is given explicitly, and contains the fundamental information of the current event. For example, in Figure 1, E1 mentions the event trigger "preparing" and arguments "Mike Pence", "presidential election", and also the time stamp "2020". Therefore, we integrate events into $\mathbb{D}$ to guide the model to capture event-related semantics, and E-SubEnc is defined as

$$\boldsymbol{S}_i^{(3)} = \mathrm{CA}(\boldsymbol{S}_i^{(2)}, \boldsymbol{E}^{(1)}) \tag{11}$$

S-Enc is followed by residual networks to control the output: $\boldsymbol{S}_i^{(4)} = \mathrm{RN}(\boldsymbol{S}_i^{(3)})$. After selecting tokens by TS-PNet, $\{\mathbb{S}_0, \mathbb{S}_1, \ldots, \mathbb{S}_{I-1}\}$ are fed into S-Enc in MHAN to obtain $\{\boldsymbol{S}_0^{(4)}, \boldsymbol{S}_1^{(4)}, \ldots, \boldsymbol{S}_{I-1}^{(4)}\}$. Then we get vector $\boldsymbol{h}_i^{(0)}$ of each sentence $\mathbb{S}_i$ by VA, i.e., $\boldsymbol{h}_i^{(0)} = \mathrm{VA}(\boldsymbol{S}_i^{(4)})$. The representation of $\mathbb{D}$ contains vectors of $\{\mathbb{S}_i\}$, i.e., $\boldsymbol{D}^{(0)} = \{\boldsymbol{h}_0^{(0)}, \boldsymbol{h}_1^{(0)}, \ldots, \boldsymbol{h}_{I-1}^{(0)}\}$, which is fed into document encoding layer in §2.3.

Token Selection Policy Network (TS-PNet)

Before encoding sentences, TS-PNet/π^t is used to select tokens for each sentence $\mathbb{S}_i$. Concretely, state, action and reward are:

State $\boldsymbol{s}_j^t$ of each token t_j^i consists of two vectors: 1) word embedding $\boldsymbol{t}_j^i \in \boldsymbol{S}_i^{(0)}$ that is the input for ϕ^c and π^t; 2) $\boldsymbol{v}_j^i \in \boldsymbol{S}_i^{(4)}$ computed by another S-Enc in π^t. Then we can get

$$\boldsymbol{s}_j^t = \tanh(\boldsymbol{t}_j^i \oplus \boldsymbol{v}_j^i) \tag{12}$$

where $\oplus$ is the concatenation operator.

Action $a_j^t \in \{0, 1\} \sim \pi^t$ represents whether token t_j^i is selected (1) or not (0), whose distribution is computed as:

$$\pi^t(a_j^t | \boldsymbol{s}_j^t, \theta_p^t) = \sigma(\boldsymbol{W}_p^t \mathrm{FC}^t(\boldsymbol{s}_j^t) + \boldsymbol{b}_p^t) \tag{13}$$

where $\mathrm{FC}^t(\cdot)$ is a stack of fully connected layers with $\tanh$ as the activation function, and σ is the logistic function.

Reward r_j^t is used to guide π^t for token selection. To reduce the variance, we compute r_j^t by the vector representation of the sentence rather than those of each token [Zhang *et al.*, 2018; Wang *et al.*, 2019]:

$$r_j^t = \log p_{\theta_c}(y | \mathrm{VA}(\boldsymbol{S}_i^{(4)})) - \epsilon^t (J^*/J) \tag{14}$$

where $p_{\theta_c}(\cdot)$ is the output layer of MHAN, y is the annotated label of the event $\mathbb{E}$ based on $\mathbb{D}$. The first term is a delay reward of $\mathbb{S}_i$ produced by MHAN, and we use VA pooling on $\boldsymbol{S}_i^{(4)}$ to get the vector for $\mathbb{S}_i$, then feed it to the softmax of MHAN to compute the probability according to y. J^* and J are the number of the selected and total tokens.

2.3 Document Encoding Layer

This layer aims to learn the vector representation for the document $\mathbb{D}$. Firstly, it captures inter-sentence knowledge by a stack of document encoders.

Document Encoders (D-Enc)

Each D-Enc contains the components below.

Intra-Document Sub-Encoder (ID-SubEnc). It learns intra-document information among the sentences, and updates the representations of $\mathbb{D}$ as

$$\boldsymbol{D}^{(1)} = \mathrm{SA}(\boldsymbol{D}^{(0)}) \tag{15}$$

Similar to S-Enc, we also consider the residual network to control the output of D-Enc, i.e., $\boldsymbol{D}^{(2)} = \mathrm{RN}(\boldsymbol{D}^{(1)})$. After selecting sentence by SS-PNet, $\boldsymbol{D}^{(0)}$ is fed into document encoding layer to compute $\boldsymbol{D}^{(2)}$, on which we apply VA pooling to obtain the final vector representation $\boldsymbol{h}_E$ of $\mathbb{D}$ w.r.t. $\mathbb{E}$: $\boldsymbol{h}_E = \mathrm{VA}(\boldsymbol{D}^{(2)})$.

Sentence Selection Policy Network (SS-PNet)

Before encoding the document $\mathbb{D}$, SS-PNet/π^s selects sentences from $\mathbb{D}$ w.r.t $\mathbb{E}$, and state, action and reward are:

State $\boldsymbol{s}_i^s$ of each sentence $\mathbb{S}_i$ is made up with two vectors: 1) $\boldsymbol{s}_i^{(0)}$ computed by VA on $\boldsymbol{S}_i^{(0)}$ that is the input for π^s; 2) $\boldsymbol{h}_i^{(2)} \in \boldsymbol{D}^{(2)}$ computed by a separate D-Enc in SS-PNet. We can obtain the state of each sentence $\mathbb{S}_i$ as

$$\boldsymbol{s}_i^s = \tanh(\boldsymbol{s}_i^{(0)} \oplus \boldsymbol{h}_i^{(2)}) \tag{16}$$

Action $a_i^s \in \{0, 1\} \sim \pi^s$ denotes whether the sentence $\mathbb{S}_i$ is selected (1) or not (0), and π^s is calculated as:

$$\pi^s(a_i^s | \boldsymbol{s}_i^s, \theta_p^s) = \sigma(\boldsymbol{W}_p^s \mathrm{FC}^s(\boldsymbol{s}_i^s) + \boldsymbol{b}_p^s) \tag{17}$$

where $\mathrm{FC}^s(\cdot)$ is a stack of fully connected layers.

Reward r_i^s is to guide π^s to select sentences [Zhang *et al.*, 2018; Wang *et al.*, 2019], and is computed as:

$$r_i^s = \log p_{\theta_c}(y | \boldsymbol{h}_i^{(2)}) - \epsilon^s (I^*/I) \tag{18}$$

where the first term is a delay reward of sentence $\mathbb{S}_i$ provided by MHAN. After SS-PNet completes all the actions, we feed each sentence whose vector representation is $\boldsymbol{h}_i^{(2)}$ into the softmax of MHAN to compute the probability according to the annotated label y of $\mathbb{E}$, and I^* and I are the numbers of the selected and total sentences, respectively.

2.4 Model Output and Optimization

Finally, $\boldsymbol{h}_E$, which is the output of D-Enc in MHAN, is fed into the softmax layer to compute the distribution of the event factuality, i.e.,

$$\boldsymbol{p} = \mathrm{softmax}(\boldsymbol{W}_s \boldsymbol{h}_E + \boldsymbol{b}_s) \tag{19}$$

Specifically, the total parameters Θ of RMHAN mainly include two sets, i.e., θ_c of MHAN, θ_p^t and θ_p^s of TS-PNet and SS-PNet. For the optimization of policy networks, we update θ_p^t and θ_p^s by REINFORCE algorithm [Williams, 1992] and policy gradient [Sutton *et al.*, 1999] to maximize the expected reward, and policy networks π^t and π^s are optimized by the gradients:

$$\nabla_{\theta_p^t}\mathcal{J}(\theta_p^t)=\sum_{j=0}^{J-1}R_j^t\nabla_{\theta_p^t}\log\pi^t(a_j^t|\boldsymbol{s}_j^t,\theta_p^t) \tag{20}$$

$$\nabla_{\theta_p^s}\mathcal{J}(\theta_p^s)=\sum_{i=0}^{I-1}R_i^s\nabla_{\theta_p^s}\log\pi^s(a_i^s|\boldsymbol{s}_i^s,\theta_p^s) \tag{21}$$

where $R_j^t=r_j^t-\mathrm{b}(\tilde{r}^t)$ and $R_i^s=r_i^s-\mathrm{b}(\tilde{r}^s)$ estimates rewards of token selection r_j^t and sentence selection r_j^s, which can minimize the variance of the individual weight changes of rewards over time [Williams, 1992], i.e., we sample trajectories $\tilde{r}_0^t,\tilde{r}_1^t,\ldots,\tilde{r}_{m_1}^t$ and $\tilde{r}_0^s,\tilde{r}_1^s,\ldots,\tilde{r}_{m_2}^s$ over tokens and sentences according to π^t and π^s, and baseline values of rewards $\mathrm{b}(\tilde{r}^t)$ and $\mathrm{b}(\tilde{r}^s)$ are approximated by the average of all the previous rewards.

For the optimization of the classification network MHAN, we update the parameter set θ_c by back propagation algorithm, and the objective function is defined as:

$$\mathcal{L}(\theta)=-\frac{1}{N}\sum_{n=0}^{N-1}\log p(y_n|\theta_c) \tag{22}$$

where y_n is the annotated label of the event $\mathbb{E}$ w.r.t $\mathbb{D}$, and N is the number of samples.

3 Experimentation

In this section, firstly, we introduce the experimental settings, including corpus, event factuality, implementation details, and baselines. Then, we give the performance of our proposed model and the detailed analysis of experiments.

3.1 Corpus

Derived from DLEF [Qian *et al.*, 2019], DLEF-v2 corpus, whose statistics are presented in Table 1, is employed as the benchmark dataset to evaluate our models. Their main differences lie in:

- *Size*. Their Chinese sub-corpora are comparable in size. But DLEF contains more Chinese documents than English ones (4649 vs. 1727). To overcome the problem of the small size of English sub-corpus, DLEF-v2 annotates more English documents compared with DLEF;
- *Annotation*. DLEF annotates not only event triggers and their document- and sentence-level factuality, but also speculative and negative cues. While DLEF-v2 further annotates a document-level event $\mathbb{E}$ according to event triggers for each document;
- *Task*. Previous work relied on annotated information. But RMHAN only utilizes documents, document-level events and their factuality for training, and does not rely on any other annotated information.

Sub-Corpus	**Uu**	**CT-**	**PS-**	**PS+**	**CT+**	**Total**
English	38	671	46	594	3181	4530
Chinese	20	1358	38	860	2374	4650

Table 1: Statistics of the DLEF-v2 corpus.

	+	-	u
CT	CT+	CT-	CTu
PS	PS+	PS-	(NA)
U	(NA)	(NA)	Uu

Table 2: Event factuality values in the DLEF-v2 corpus.

3.2 Event Factuality Values

Event factuality values are composed of modality and polarity [Saurí and Pustejovsky, 2012]. Modality conveys the certainty degrees of events, including certain (CT), probable (PR), and possible (PS), while polarity expresses whether the event happened, containing positive (+) and negative (-). In this paper, we use the factuality values in Table 2 [Qian *et al.*, 2019], where underspecified (U/u) means unknown or uncommitted, which is a default value for both modality and polarity. PR and PS are merged into PS due to the similar semantics on modality. PSu and U+/- are not applicable (NA) [Saurí, 2008; Saurí and Pustejovsky, 2012]. No events can be annotated as CTu in DLEF-v2 though applicable. Therefore, there are five applicable factuality values in DLEF-v2 corpus, i.e., Uu, CT-, PS-, PS+, CT+.

3.3 Implementation Details

We focus on the performance of CT-, PS+ and CT+, since these events occupy 98.15%/98.75% in English/Chinese sub-corpus. Similar to [Qian *et al.*, 2019; Cao *et al.*, 2021], we do not consider PS- and Uu due to their small proportions. 10-fold cross validation are performed on English and Chinese sub-corpus. Word embeddings are pre-trained by GloVe [Pennington *et al.*, 2014]. To reduce the variance, we employ warm start, i.e., firstly train RMHAN without discarding any tokens or sentences using Adam, then consider both token and sentence selection and continue to optimize with SGD.

3.4 Baselines

For fair comparison with our model, we implement several baselines:

- SEFI model: 1) **SGCN** [Veyseh *et al.*, 2019] is an Sentence-level model with Graph Convolutional Network, and employing voting for DEFI;
- DEFI models: 2) **LSTM-A** [Qian *et al.*, 2019] adopts multi-layer LSTM with attention; 3) **ULGN** [Cao *et al.*, 2021] designs graph networks relying on event triggers, speculative and negative cues;
- Target-specific classification models: 4) **TEND-C** [Duan *et al.*, 2018] and **TEND-T** [Duan *et al.*, 2019] encodes document relying on targets (i.e., events in this paper). TEND-C utilizes LSTM, while TEND-T employs attention; 5) **RLSTM** [Wang *et al.*, 2019] is a reinforced LSTM network with token and sentence selection for aspect sentiment classification;

Models	CT-	PS+	CT+	Macro-Ave	Micro-Ave
SGCN	45.48 / 60.78	40.80 / 50.63	77.71 / 76.82	54.66 / 62.74	67.20 / 66.52
LSTM-A	42.05 / 59.08	41.07 / 54.68	78.43 / 77.04	53.85 / 63.60	67.23 / 67.53
ULGN	45.87 / 61.07	43.05 / 49.58	81.87 / 76.49	56.93 / 62.38	70.55 / 66.27
TEND-C	41.42 / 58.16	41.50 / 45.74	78.32 / 74.80	53.75 / 59.56	67.46 / 64.47
TEND-T	47.47 / 63.83	46.09 / 55.25	80.13 / 77.15	57.90 / 65.41	70.52 / 69.02
RLSTM	49.89 / 64.64	48.80 / 54.83	82.04 / 77.92	60.24 / 65.80	72.39 / 69.51
BERT-B	51.14 / 69.09	51.12 / 60.91	80.82 / 80.92	61.03 / 70.31	72.31 / 73.60
MHAN	50.76 / 68.16	50.93 / 60.86	82.19 / 80.06	61.29 / 69.69	73.32 / 73.28
RMLAN	46.39 / 63.86	45.96 / 54.06	81.08 / 77.88	57.81 / 65.27	71.06 / 69.03
RMHAN-SP	52.22 / 70.85	53.01 / 63.15	83.66 / 82.09	62.96 / 72.03	74.56 / 75.03
RMHAN	**56.43 / 73.83**	**55.13 / 65.55**	**84.35 / 82.60**	**65.30 / 73.99**	**76.38 / 77.07**

Table 3: Performance of various models on DEFI. Format: F1-scores for English / Chinese sub-corpus.

Models	CT-	PS+	CT+	Macro-Ave	Micro-Ave
RMHAN	56.43 / 73.83	55.13 / 65.55	84.35 / 82.60	65.30 / 73.99	76.38 / 77.07
w/o IS-SubEnc	-4.99 / -4.81	-3.85 / -4.95	-3.95 / -3.81	-4.26 / -4.52	-4.16 / -4.84
w/o T-SubEnc	-8.40 / -8.50	-7.32 / -8.20	-6.57 / -6.49	-7.43 / -7.72	-7.25 / -7.86
w/o E-SubEnc	-6.70 / -7.09	-6.83 / -6.49	-5.57 / -5.07	-6.36 / -6.21	-6.25 / -6.39
w/o ID-SubEnc	-3.79 / -4.02	-2.60 / -3.80	-2.10 / -3.46	-2.83 / -3.76	-2.74 / -4.05
w/o TS-PNet	-1.17 / -1.21	-1.18 / -1.51	+0.40 / -0.54	-0.65 / -1.08	-0.46 / -1.32
w/o SS-PNet	-4.96 / -3.66	-3.11 / -3.10	-1.24 / -1.39	-3.10 / -2.71	-2.30 / -2.85

Table 4: Performance of ablation study for RMHAN. Format: F1-scores for the complete RMHAN model, and improvement of F1-scores for other models w/o some components on English / Chinese sub-corpus.

- Pre-trained model: 6) **BERT-B** [Devlin *et al.*, 2019] is the base version of BERT;
- The variants of RMHAN model: 7) **MHAN** is the supervised classification network in RMHAN, i.e., does not consider policy networks with reinforcement learning; 8) **RMLAN** concatenates all the sentences into a single one as input and only considering token selection; 9) **RMHAN-SP** has simpler policy networks compared with RMHAN, i.e., does not consider S-Enc or D-Enc, and only employs fully connected layers in TS-PNet and SS-PNet.

Specially, the parameter size of BERT-B is at the same order of magnitude as RMHAN for fair comparison and real-world application. Large versions (e.g., BERT-Large and RoBERTa-Large) have much more parameters than BERT-B and RMHAN. Therefore, it is not a fair comparison for BERT-Large (or RoBERTa-Large, etc.) with RMHAN, since our model is task-specific. Although RMHAN has more layers, it has lower dimensions of hidden states and fewer parameters than BERT-Base and RoBERTa-Base.

3.5 Overall Results

Table 3 presents the results of RMHAN and several baselines on end-to-end DEFI task. RMHAN significantly outperforms other models (*t-test*: $p<0.05$ for RMHAN vs. RMHAN-SP, $p<0.01$ for RMHAN vs. others), proving the effectiveness of both text selection and comprehensive understanding integrated, whose attributes can be summarized as follows:

R/Reinforced. RMHAN is superior to the models without reinforcement learning, because policy networks can select the most relevant and meaningful tokens (e.g., event arguments, speculative and negative cues) and sentences with regard to the event. Moreover, RMHAN is better than RMHAN-SP, demonstrating the separate sentence and document encoders in policy networks TS-PNet and SS-PNet can learn useful semantics, and are beneficial to both token and sentence selection.

H/Hierarchical. RMHAN gets higher performance than the models concatenating all the texts into a single sentence as the input, e.g., BERT-B and RMLAN. Since sentences may hold different factuality values w.r.t. the events, the hierarchical structured model can extract meaningful semantics from them, especially from the core sentences of documents. Due to the hierarchical framework, RMHAN can select the most useful sentences by sentence selection policy network as well.

A/Attention. RMHAN obtains better results than those models with simpler structures, e.g, TEND-C, TEND-T and RLSTM. Based on attention, RMHAN can not only learn contextual information within and among sentences, but also integrate topics and events into documents to capture event-related semantics. Moreover, Table 3 also demonstrates that attention is better at modelling documents than LSTM on DEFI, e.g., MHAN/TEND-T vs TEND-C.

The performance of CT- and PS+ is lower on English sub-corpus than that of Chinese one, indicating it is more difficult to identify speculation and negation for events in English documents due to fewer CT- and PS+ samples. In addition, the low results of other baselines are due to their limitations, e.g., based on LSTM, TEND-C is much simpler. SGCN, LSTM-A and ULGN rely on annotated information. To make them comparable with our end-to-end model, we use raw texts as the input, i.e., first detecting event triggers

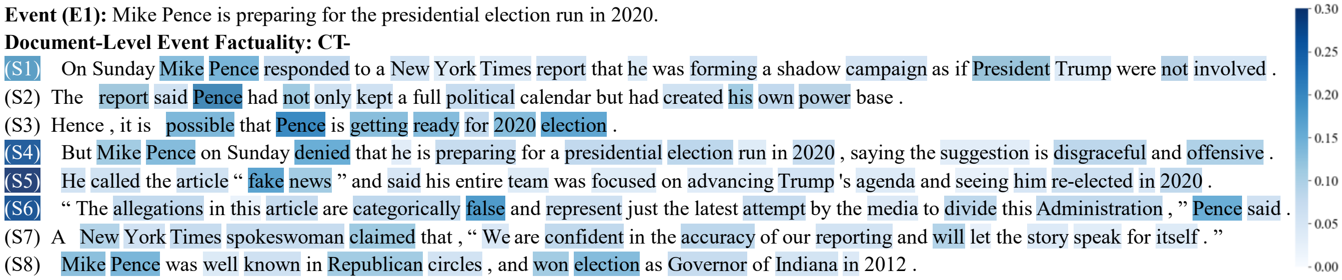

Figure 3: Visualizations of tokens and sentences selected by RMHAN for event E1. Attention weights are computed by VA pooling, and are visualized as background colors of tokens and sentence IDs, while no background color means this token or sentence has been discarded.

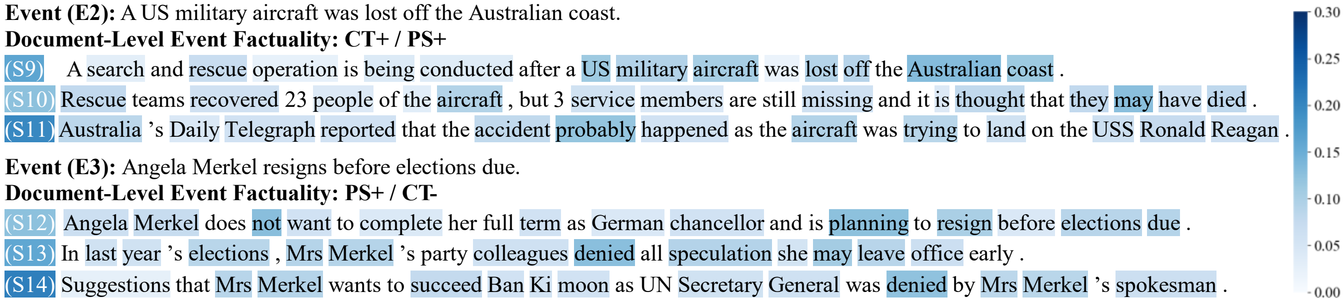

Figure 4: Visualizations of some error cases predicted by RMHAN. Format of labels: **Annotated / Predicted**. For each document, only several sentences selected by the sentence selection policy network (SS-PNet) with higher attention weights are listed.

(F1=83.19%/79.65% for English/Chinese sub-corpus), speculative and negative cues (F1=68.80%/75.42%), then identifying event factuality. Hence, their low results are mainly due to cascade errors. The low performance of SGCN shows that voting is less effective, because DEFI depends on the main semantics of documents, rather than simply aggregating or counting factuality values of sentences, since it does not conform to the mechanism of document understanding. TEND-T is superior to LSTM-A and TEND-C, mainly attributed to the hierarchical architecture and attention. Compared with most models (except for RMHAN-SP and RMHAN), BERT-B gets better results because of multi-head attention layers.

3.6 Ablation Study

To further evaluate the **M**/Multi-Granularity of RMHAN, i.e., the contributions of each component, we launch ablation study and present the results in Table 4, which manifests all the sub-encoders and policy networks are beneficial to RMHAN on DEFI. To be specific:

Sentence Encoder. Both topic and event sub-encoders are more important than intra-sentence sub-encoder. Topics and events usually contain event-related information, e.g., event triggers and arguments. Hence, T-SubEnc and E-SubEnc can capture correlative semantics about events and integrate them into the document. Specially, events, which can be seen as "weak" topics in most documents, usually contain less information and improve lower performance than topics. Meanwhile, IS-SubEnc is also useful, since it is mainly employed to learn information within sentences, including speculative and negative information.

Document Encoder. The removal of ID-SubEnc also leads to lower performance, especially on CT- and PS+. Actually, ID-SubEnc can learn interactive information among sentences, including those with speculative and negative semantics, and the ones that summarize the main idea of the document. In addition, ID-SubEnc, whose main function is learning inter-sentence knowledge, is also effective on deciding which sentences contribute to correct results.

Policy Networks. We can conclude that sentence selection with SS-PNet is more important and meaningful for end-to-end DEFI task than token selection with TS-PNet. We argue that even if we do not consider token selection, those irrelevant or useless words are assigned lower attention weights by vanilla attention pooling, and hence have fewer effects on the semantics of sentences and documents. Compared to tokens, sentences contain more information, and performance will drop if SS-PNet is not considered, especially when sentences hold different factuality of the event, and irrelevant and noisy sentences can interfere with event-related and core ones that are crucial to the results of DEFI.

3.7 Case Study

To illustrate which tokens and sentences are more relevant and meaningful for end-to-end DEFI, we provide a qualitative analysis of RMHAN on event E1 in Figure 1, whose attention weights are visualized in Figure 3.

In general, selected tokens and sentences with higher attention weights contribute to the correct identification of the document-level factuality of E1. For token selection, there are mainly two types of tokens that are selected and assigned higher weights, i.e., 1) event arguments in topic and event sentences, including entities (e.g., "Mike Pence"); 2) negative words (e.g., "denied" in S4, "fake" in S5, and "false" in S6) in selected sentences that contain event mentions and can determine the factuality of E1. These results verifies the capability of S-Enc and TS-PNet in the sentence encoding layer.

While for sentence selection, those sentences containing event mentions and negative information are selected and have higher weights, i.e., S4, S5, and S6. It is worth noting that S1 is also selected, which can prove the usefulness of T-SubEnc. These four sentences summarize the semantics of the document w.r.t. E1, manifesting the validity of SS-PNet. Finally, the vector representation of E1 is also based on selected sentences encoded by document encoding layer, and the correct result is also attributed to ID-SubEnc in D-Enc.

3.8 Error Analysis

Since the performance of CT- and PS+ is lower than that of CT+, errors mainly derive from the wrong identification of speculative and negative information. To illustrate the production of errors, we list two error cases in Figure 4. Event E2 is annotated as CT+ according to sentence S9, but it is predicted as PS+ mistakenly, mainly attributed to the speculative information in S10 and S11. Actually, speculative cue "may" in S10 and "probably" in S11 cannot affect E2 grammatically. But RMHAN does not filter S10 and S11.

While event E3 is PS+ according to sentence S12. However, E3 gets a wrong predicted value CT- mainly due to sentence S13 and S14. On the timeline, S13 is before the current document, and cannot negate document-level factuality of E3, while S14 denies another event "Mrs Merkel wants to succeed Ban Ki moon as UN Secretary General" rather than E3. But RMHAN does not filter S13 and S14, assigning them considerable attention weights mistakenly.

These two examples illustrate DEFI models need to judge whether speculative and negative information can effect the event or not.

4 Related Work

4.1 Event Factuality Identification

Event Factuality Identification (EFI) is primarily concerned with Sentence-level Event Factuality Identification (SEFI) and Document-level Event Factuality Identification (DEFI), and most previous studies considered SEFI, e.g., rule-based models [Saurí and Pustejovsky, 2012], traditional machine learning models [de Marneffe *et al.*, 2012; Lee *et al.*, 2015; Qian *et al.*, 2015], or hybrid approaches of them [Qian *et al.*, 2015]. Recently, neural networks has been applied to SFEI, e.g., learning plain lexical and semantic information [He *et al.*, 2017; Sheng *et al.*, 2019], considering dependency trees [Rudinger *et al.*, 2018; Qian *et al.*, 2018a; Qian *et al.*, 2018b; Cao *et al.*, 2020; Zhang *et al.*, 2020] or graph based network [Veyseh *et al.*, 2019; Le and Nguyen, 2021]. Specially, [Qian *et al.*, 2018a] designed a generative adversarial network to produce more syntactic information.

Compared with SEFI, DEFI is still in its preliminary stage. Based on DLEF corpus [Qian *et al.*, 2019], [Qian *et al.*, 2019; Huang *et al.*, 2019] employed multi-layer LSTM with attention pooling to learn features from dependency paths and sentences. [Cao *et al.*, 2021] extracted local and global information for events with a graph convolution network. However, the main limitations of them are that they depended heavily on annotated information, e.g., event triggers, speculative and negative cues, and did not discard any noisy texts.

4.2 Reinforcement Learning

Reinforcement Learning (RL) is effective on optimization for non-differentiable objectives, and obtains satisfactory performance in classification tasks in NLP [Zhang *et al.*, 2018; Fei *et al.*, 2019; Ye *et al.*, 2020]. Hierarchical Reinforcement Learning (HRL) usually integrates two-level networks to capture different level information, e.g., [Wang *et al.*, 2019] incorporated clause and word selection for document-level aspect sentiment classification. [Xiao *et al.*, 2020] proposed a hybrid two-step framework for summarization switching between copying and rewriting sentences. [Wan *et al.*, 2020] built a hierarchical two-level policies for structured encoding on link predictions.

This paper studies end-to-end DEFI task for the first time, and designs RMHAN model with HRL that can select relevant and meaningful tokens and sentences, and learn different levels of features from them hierarchically for events.

5 Conclusion

This paper proposes a novel multi-granularity model for end-to-end DEFI, i.e., RMHAN, which can not only learn contextual information from tokens and sentences with multi-granularity encoders hierarchically, but also select the most useful tokens and sentences with hierarchical reinforcement learning. Experimental results on DLEF-v2 corpus show that RMHAN is superior to several state-of-the-arts on DEFI. In the future work, we plan to launch more fine-grained end-to-end DEFI task, e.g., identifying factuality of several events separately in one document and evidential sentences of them, and exploring cross-document task.

A Resources

For more details of the resources and reproducibility of this paper, please refer to https://github.com/qz011/rmhan.

Acknowledgments

The authors would like to thank the three anonymous reviewers for their comments on this paper. This work was supported by national Natural Science Foundation of China (NSFC) via Grant Nos. 62006167 and 61836007, and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

[Cao *et al.*, 2020] Jinjuan Cao, Zhong Qian, Peifeng Li, Xiaoxu Zhu, and Qiaoming Zhu. End-to-end event factuality identification via hybrid neural networks. In *CCKS*, 2020.

[Cao *et al.*, 2021] Pengfei Cao, Yubo Chen, Yuqing Yang, Kang Liu, and Jun Zhao. Uncertain local-to-global networks for document-level event factuality identification. In *EMNLP*, 2021.

[de Marneffe *et al.*, 2012] Marie-Catherine de Marneffe, Christopher D. Manning, and Christopher Potts. Did it happen? the pragmatic complexity of veridicality assessment. *Comput. Linguistics*, 38(2):301–333, 2012.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[Duan *et al.*, 2018] Junwen Duan, Yue Zhang, Xiao Ding, Ching-Yun Chang, and Ting Liu. Learning target-specific representations of financial news documents for cumulative abnormal return prediction. In *COLING*, 2018.

[Duan *et al.*, 2019] Junwen Duan, Xiao Ding, Yue Zhang, and Ting Liu. TEND: A target-dependent representation learning framework for news document. *IEEE ACM Trans. Audio Speech Lang. Process.*, 27(12):2313–2325, 2019.

[Fei *et al.*, 2019] Hongliang Fei, Xu Li, Dingcheng Li, and Ping Li. End-to-end deep reinforcement learning based coreference resolution. In *ACL*, 2019.

[He *et al.*, 2017] Tianxiong He, Peifeng Li, and Qiaoming Zhu. Identifying chinese event factuality with convolutional neural networks. In *CLSW*, 2017.

[Huang *et al.*, 2019] Rongtao Huang, Bowei Zou, Hongling Wang, Peifeng Li, and Guodong Zhou. Event factuality detection in discourse. In *NLPCC*, 2019.

[Le and Nguyen, 2021] Duong Le and Thien Huu Nguyen. Does it happen? multi-hop path structures for event factuality prediction with graph transformer networks. In *Proceedings of W-NUT 2021*, 2021.

[Lee *et al.*, 2015] Kenton Lee, Yoav Artzi, Yejin Choi, and Luke Zettlemoyer. Event detection and factuality assessment with non-expert supervision. In *EMNLP*, 2015.

[Lu *et al.*, 2016] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, 2016.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[Qian *et al.*, 2015] Zhong Qian, Peifeng Li, and Qiaoming Zhu. A two-step approach for event factuality identification. In *IALP*, 2015.

[Qian *et al.*, 2018a] Zhong Qian, Peifeng Li, Yue Zhang, Guodong Zhou, and Qiaoming Zhu. Event factuality identification via generative adversarial networks with auxiliary classification. In *IJCAI*, 2018.

[Qian *et al.*, 2018b] Zhong Qian, Peifeng Li, Guodong Zhou, and Qiaoming Zhu. Event factuality identification via hybrid neural networks. In *ICONIP*, 2018.

[Qian *et al.*, 2019] Zhong Qian, Peifeng Li, Qiaoming Zhu, and Guodong Zhou. Document-level event factuality identification via adversarial neural network. In *NAACL*, 2019.

[Rudinger *et al.*, 2018] Rachel Rudinger, Aaron Steven White, and Benjamin Van Durme. Neural models of factuality. In *NAACL-HLT*, 2018.

[Saurí and Pustejovsky, 2012] Roser Saurí and James Pustejovsky. Are you sure that this happened? assessing the factuality degree of events in text. *Comput. Linguistics*, 38(2):261–299, 2012.

[Saurí, 2008] Roser Saurí. *A Factuality Profiler for Eventualities in Text*. PhD thesis, Brandeis University, Waltham, MA, USA, 2008.

[Sheng *et al.*, 2019] Jiaxuan Sheng, Bowei Zou, Zhengxian Gong, Yu Hong, and Guodong Zhou. Chinese event factuality detection. In *NLPCC*, 2019.

[Sutton *et al.*, 1999] Richard S. Sutton, David A. McAllester, Satinder P. Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *NIPS*, 1999.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[Veyseh *et al.*, 2019] Amir Pouran Ben Veyseh, Thien Huu Nguyen, and Dejing Dou. Graph based neural networks for event factuality prediction using syntactic and semantic structures. In *ACL*, 2019.

[Wan *et al.*, 2020] Guojia Wan, Shirui Pan, Chen Gong, Chuan Zhou, and Gholamreza Haffari. Reasoning like human: Hierarchical reinforcement learning for knowledge graph reasoning. In *IJCAI*, 2020.

[Wang *et al.*, 2019] Jingjing Wang, Changlong Sun, Shoushan Li, Jiancheng Wang, Luo Si, Min Zhang, Xiaozhong Liu, and Guodong Zhou. Human-like decision making: Document-level aspect sentiment classification via hierarchical reinforcement learning. In *EMNLP*, 2019.

[Williams, 1992] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8:229–256, 1992.

[Xiao *et al.*, 2020] Liqiang Xiao, Lu Wang, Hao He, and Yaohui Jin. Copy or rewrite: Hybrid summarization with hierarchical reinforcement learning. In *AAAI*, 2020.

[Xiong *et al.*, 2017] Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. In *ICLR*, 2017.

[Ye *et al.*, 2020] Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Jingmin Chen, Xiaoxiao Xu, Suhang Zheng, Feng Wang, Jun Zhang, and Huajun Chen. Zero-shot text classification via reinforced self-training. In *ACL*, 2020.

[Zhang *et al.*, 2018] Tianyang Zhang, Minlie Huang, and Li Zhao. Learning structured representation for text classification via reinforcement learning. In *AAAI*, 2018.

[Zhang *et al.*, 2020] Liumin Zhang, Peifeng Li, Zhong Qian, Xiaoxu Zhu, and Qiaoming Zhu. Employing multi-cues to identify event factuality. In *CCKS*, 2020.