Appendix for Paper 4513

1 Comparing with Transformer and BERT

We conduct four experiments on TREC-RTS-16 dataset to evaluate the effectiveness of Transformer (Vaswani et al., 2017) and BERT (Devlin et al., 2018) in real-time event summarization: (1) we use the pre-trained BERT to obtain the representation of the query and document since end-to-end training BERT is unfeasible in most academic groups due to resource constraints. "[CLS]" token is added at the start of the sequence, whose embedding is treated as the representation of the text sequence. A binary classifier is then used to classify the input document as "skip" or "keep"; (2) we use Transformer to learn the representations of the query and document, and a binary classifier is then used to classify the document; (3) we replace the Knowledge-enhanced Document/Query Representation Learning module in DRES with pre-trained BERT (denoted as DRES-BERT) to learn the representations of query and document, and keep the remaining parts of DRES unchanged; (4) we replace the Knowledge-enhanced Document/Query Representation Learning module in DRES with Transformer (denoted as DRES-Transformer) to learn the representations of query and document, and keep the remaining parts of DRES unchanged.

The experimental results are reported in Table A.1. From the results we can observe that our model outperforms the Transformer and BERT by a large margin on TREC-RTS-16 dataset. This is because that Transformer and BERT cannot extract the interactive relations between the query and document. The results by DRES+BERT and DRES+Transformer are slightly worse that our results. This verifies the effectiveness of the knowledge-enhanced document/query representation learning module which incorporate the external knowledge from KB into the deep neural networks via hierarchical attention networks. The three-stage hierarchical attention mechanism can exploit the semantic compositionality of the input sequences and capture comprehensive information at different level of granularity.

Method	EG-0	nCG-0	EG-1	nCG-1	GMP	Latency
DRES	0.091	0.106	0.324	0.332	-0.065	68573
Transformer	0.062	0.074	0.263	0.274	-0.174	82545
BERT	0.057	0.069	0.267	0.269	-0.156	83824
DRES-Transformer	0.086	0.095	0.298	0.306	-0.116	73123
DRES-BERT	0.078	0.091	0.302	0.309	-0.103	72951

Table A.1: Experiments for Transformer and BERT on TREC-RTS-16.

2 Computational Cost

Since it is difficult to analyze the time complexity of deep learning models theoretically, we investigate the computational cost of baseline methods and the proposed DRES model on the TREC-RTS-16 dataset, instead of analyzing the time complexity. All the models run on a single NVIDIA Tesla P100 GPU. At the training phase, DRES takes about 105 seconds per training epoch. Based on our empirical observation, most compared baselines take about 10-60 seconds per training epoch on an average . All models typically converge within less than 20 epochs by using the early stopping criterion. At the testing phase, our model takes about 2.35 milliseconds to predict the label for each document, while the baselines take about 0.24-1.42 milliseconds on an average. The training time per epoch (seconds) and testing time (milliseconds) per instance on the DRES dataset are given in Table A.2.

3 Experimental Results with Different Balance Parameters for GMP

We conduct experiments by using 0.33, 0.5, and 0.66 as the balance parameters for calculating GMP. The experimental results are summarized in Table A.3. Our model significantly outperforms the compared methods on all the evaluation metrics.

Method	Training time per epoch(s)	Testing time per instance (ms)
IPS	10.5	0.24
AP	15.2	0.74
CST	11.3	0.43
LS	47.9	1.28
NNRL	59.6	1.42
DRES	105.3	2.35

Table A.2: Training time per epoch (seconds) and test time per instance (milliseconds) on TREC-RTS-16 dataset.

Method	EG-0	nCG-0	EG-1	nCG-1	$GMP_{0.33}$	$GMP_{0.5}$	$GMP_{0.66}$	Latency
IPS	0.033	0.039	0.201	0.213	-0.536	-0.324	-0.273	192344
AP	0.037	0.031	0.232	0.235	-0.145	-0.103	-0.081	134077
CST	0.048	0.063	0.262	0.254	-0.407	-0.321	-0.193	91456
LS	0.070	0.081	0.271	0.297	-0.242	-0.185	-0.083	85665
NNRL	0.069	0.085	0.282	0.288	-0.297	-0.236	-0.105	84343
COMP2016 (winner)	0.048	0.069	0.270	0.291	-0.326	-0.205	-0.092	91549
DRES	**0.091***	**0.106***	**0.324***	**0.332***	**-0.124***	**-0.065***	**-0.059***	**68573***

Table A.3: Event summarization results on TREC-RTS-16. Numbers with * mean that improvement from our model is statistically significant over the baseline methods (t-test, p-value < 0.05).

4 Experimental Results on TREC-RTS-17

TREC-RTS-17 consists of 203,344 documents which are provided by Twitter API (approximately 1% tweets of all documents) from July 29, 2017 to August 5, 2017. To identify relevant tweets, a set of profiles are provided. Each profile (called topic in the TREC jargon) is composed of a title, a description and a narrative of the interest profile. Each document has a relevance label (i.e., *highly relevant*, *relevant* and *non-relevant*) with respect to the given interest profile. There are 97 interest topics being monitored during the evaluation period. We randomly choose the tweets of 80 interest topics as training set, the tweets of 7 interest topics as the validation set, and the rest samples are used for testing.

The experimental results are summarized in TableA.4. From the results, we can observe that DRES consistently and substantially surpasses the compared models by a large margin on most of the evaluation metrics. The improvement from DRES is statistically significant over the compared models (t-test, p-value < 0.05).

Method	EG-0	nCG-0	EG-1	nCG-1	$GMP_{0.33}$	$GMP_{0.5}$	$GMP_{0.66}$	Latency
IPS	0.185	0.154	0.147	0.138	-0.326	-0.218	-0.159	139273
AP	0.247	0.149	0.197	0.142	-0.314	-0.183	-0.113	126575
CST	0.253	0.184	0.175	0.126	-0.313	-0.189	-0.108	68112
DES	0.257	0.153	0.223	0.157	-0.286	-0.154	-0.091	77427
LS	0.373	0.295	0.246	0.215	-0.238	-0.087	-0.073	58392
NNRL	0.318	0.262	0.194	0.227	-0.285	-0.152	-0.054	64758
HLJIT (winner)	0.363	0.281	0.209	0.127	-0.272	-0.157	-0.048	119374
DRES	**0.389***	**0.296**	**0.264***	**0.259***	**-0.189***	**-0.124***	**-0.032***	**44827***

Table A.4: Event summarization results on TREC-RST-17. Numbers with * mean that improvement from our model is statistically significant over the baseline methods (t-test, p-value < 0.05)

5 Hyperparameter Sensitivity Analysis

In this section, we analyze how the hyperparameters λ in Eq.(21) and γ in Eq.(23) affect the performance of DRES.

5.1 Sensitivity Analysis of Hyperparameter λ for Reward

λ is the hyperparamter that controls the effect of EG, cCG and $Latency$ in calculating the reward for reinforcement learning. In this experiment, we analyze the impact of λ on the overall performance of

DRES by using different values of λ_1, λ_2 and λ_3 while keeping $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Due to the limited time, we choose five combinations of λ_1, λ_2 and λ_3 to analyze the sensitivity of λ. The experimental results on TREC-RST-16 dataset are shown in Table A.5. Generally speaking, the performance of DRES is not very sensitive to the hyperparameter λ over a wide range. However, a large λ_1 or λ_3 will hurt the overall performance.

$(\lambda_1, \lambda_2, \lambda_3)$	EG-0	nCG-0	EG-1	nCG-1	$GMP_{0.33}$	$GMP_{0.5}$	$GMP_{0.66}$	Latency
(0.15, 0.8, 0.05)	0.091	0.106	0.324	0.332	-0.081	-0.065	-0.013	68573
(0.2, 0.7, 0.1)	0.092	0.105	0.326	0.331	-0.079	-0.064	-0.016	68824
(0.3, 0.6, 0.1)	0.095	0.102	0.327	0.328	-0.081	-0.068	-0.022	69172
(0.1, 0.7, 0.2)	0.092	0.107	0.325	0.329	-0.083	-0.063	-0.015	68455
(0.3, 0.4, 0.3)	0.089	0.102	0.322	0.326	-0.087	-0.071	-0.023	68013

Table A.5: Sensitivity analysis for hyperparameter λ on TREC-RST-16 dataset.

5.2 Sensitivity Analysis of Hyperparameter γ for Joint Training

γ defined in Eq.(23) is the hyper-parameter that determines the weights of L_1 and L_2. In this experiment, we analyze the impact of γ on the overall performance of DRES by varying the value of γ_2 from 0.1 to 1 with step size 0.1 on TREC-RST-16 dataset while keeping $\gamma_1 + \gamma_2 = 1$. The special case of $\gamma_1 = 0$ indicates that the relevance prediction task will not be trained jointly with event summarization task. We illustrate the experimental results in Figure 1. As γ_2 rises from 0.1 to 1, the EG-0 and nCG-1 scores increase sharply till $\gamma_2 = 0.8$, after which the results decrease.

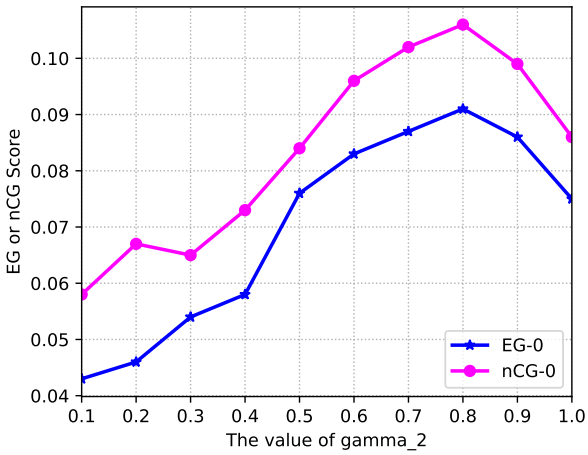

Figure 1: EG-0 and nCG-0 scores on TREC-RST-16 dataset dataset with different values of γ.

6 Effect of Multi-head Attention Mechanism

In order to further investigate how the multi-head attention mechanism works qualitatively, we choose one document "*This Saturday, August 6th, will mark the 71st anniversary of the Hiroshima bombing. Time for abolition.*" from TREC-RTS-16 as a case study and illustrate the attention weights with $b = 2$ (b is the number of heads of multi-head attention). Figure 2 visualizes the attention weights on the input document with respect to the query. The color depth indicates the importance degree of the attention. The darker the color, the more important the word. From Figure 2, we can see that our model can identify the essential content from different representation subspaces. For example, DRES firstly notices the words about time (i.e., "Saturday" and "August 6th") via the first hop of attention. Then, it notices the words about the event (i.e., "71th anniversary" and "Hiroshima bombing") via the second hop of attention. The multi-head attention can model the overall semantics of the input text by combining the attention weights of each row of the attention matrix Σ.

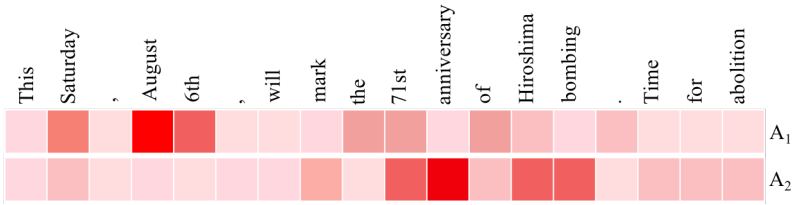

Figure 2: The attention weights for a input document of TREC-RTS-16 by our model (with the number of attention heads is 2, i.e., $b = 2$).

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* .

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. pages 5998–6008.