# Appendix for Paper 3534

## 1 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 as a case study and illustrate the attention weights with $b = 2$ (b is the number of hops of attention). Figure 1 shows the representation of how the attention focuses 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 1, we can observe that our model can capture the important information from different representation subspaces at different positions. For example, HRES 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"). 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 $A$.

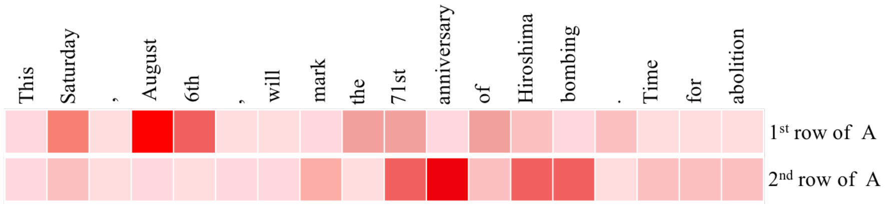

Figure 1: The attention weights for a input document of TREC-RTS by our model (with the number of attention hops is 2, i.e., $b = 2$).

## 2 The details of Evaluation Metrics EG and nCG

The assessment workflow proceeded in two major stages: relevance assessment and semantic clustering. Here we provide only a brief overview, referring the reader to previous work (Roegiest et al., 2017) for additional details.

documents returned by event summarization systems were judged for relevance by NIST assessors (Roegiest et al., 2017) via pooling. Note that this occurred after the live evaluation period ended, so it was possible to gather all documents pushed by all participating systems. NIST assessors began a few days after the end of the evaluation period to minimize the "staleness" of documents. Each document was assigned one of three judgments: not relevant, relevant, or highly-relevant. **After the relevance assessment process, the NIST assessors proceeded to perform semantic clustering on only the relevant and highly-relevant documents. Using a custom interface, they grouped documents into clusters in which documents share substantively similar content, or more colloquially, "say the same thing". The interpretation of what this means operationally was left to the discretion of the assessor. In particular, they were not given a particular target number of clusters to form; rather, they were asked to use their judgment, considering both the interest profile and the actual documents. The output of the cluster annotation process is a list of document clusters; each cluster contains documents that are assumed to convey the same information.**

As previously discussed, update summaries should be relevant, non-redundant, and timely. One challenge, however, is that there is little empirical work on how users perceive timeliness. Therefore, instead of devising a single-point metric that tries to combine all three characteristics, the organizers decided to separately capture output quality (relevance and redundancy) and timeliness (latency). In this paper, we only elaborate the output quality metrics EG and nCG.

### 2.1 Expected Gain (EG)

for an interest profile on a particular day is defined as $\frac{1}{N}\sum G(t)$, where $N$ is the number of documents returned and $G(t)$ is the gain of each document: not relevant documents receive a gain of 0; relevant documents receive a gain of 0.5; highly-relevant documents receive a gain of 1.0. Once a document from a cluster is retrieved, all other documents from the same cluster automatically become not relevant. This penalizes systems for returning redundant information. Expected gain can be interpreted as a precision metric.

### 2.2 Normalized Cumulative Gain (nCG)

For an interest profile on a particular day is defined as $\frac{1}{N}\sum G(t)$, where$Z$ is the maximum possible gain (given the ten document per day limit). The gain of each individual document is computed in the same way as above. Note that gain is not discounted (as in nDCG) because the notion of document ranks is not meaningful in this context. We can interpret nCG as a recall-like metric. The score for a run is the average over scores for each day over all interest profiles. An interesting question is how scores should be computed for days in which there are no relevant documents: for rhetorical convenience, we call days in which there are no relevant documents for a particular interest profile (in the pool) "silent days", in contrast to "eventful days" (when there are relevant documents). In the EG-1 and nCG-1 variants of the metrics, on a silent day, the system receives a score of one (i.e., a perfect score) if it does not push any documents, or a score of zero otherwise. Therefore, under EG-1 and nCG-1, systems are rewarded for recognizing that there are no relevant documents for an interest profile on a particular day and remaining silent (i.e., the system does not push any documents).

Thus, by taking EG and nCG as RL reward, our RL method can avoid redundancy and update more relevant documents.

## 3 Updating the RL Reward Function

The policy gradient algorithm is a type of reinforcement learning methods, which relies upon optimizing parametrized policy with respect to the expected return (long-term cumulative rewards) by gradient descent. When we reach the end of the sequence of document-query representations, the expected reward (RT) will be calculated from the predicted distribution, which represents the score for producing the global action sequence $a_{1:T}$ given document stream and the update summary. This is a typical delayed reward since we cannot obtain it until the final action distribution is predicted. In order to update documents that are relevant, non-redundant, and timely, we define the delayed final reward $R_T$ as follows:

$$R_T = r(a_{1:T}) = \lambda_1 EG(a_{1:T}) + \lambda_2 nCG(a_{1:T}) + \lambda_3 Latency. \tag{1}$$

where $r(\cdot)$ is the reward function; $\lambda$ controls the effect of $EG$, $cCG$ and Latency, and we empirically demonstrate that we can achieve best results when $\lambda_1 = 0.15, \lambda_2 = 0.8, \lambda_3 = 0.05$. EG and nCG can capture the output quality (relevance and redundancy), while Latency can provide guidance of timeliness. Here, the calculation of EG and nCG is described above. The latency score only computed on the documents that are relevant. Latency is defined as the mean difference between the time the document was pushed and the first document in the semantic cluster that the document belongs is reported.

Table A.1 demonstrates the experimental results of HRES by employing the updated RL reward function (see Equation 1) on TREC-RTS dataset. HRES with new RL reward function (denoted as HRES-new) achieves comparable or slightly better results than HRES. Especially, the Latency by HRES-new is significantly better than that by HRES.

| Method | EG-0 | nCG-0 | EG-1 | nCG-1 | GMP | Latency |
|---|---|---|---|---|---|---|
| HRES | 0.082 | 0.097 | 0.309 | 0.315 | -0.071 | 70573 |
| HRES-new | 0.086 | 0.099 | 0.314 | 0.317 | -0.070 | 70158 |

Table A.1: Event summarization results on TREC-RTS.

## 4 Statistical Significance Tests

In this experiment, we also perform the statistical significance tests. The experimental results are summarized in Table A.2 and Table A.3, which show that our model is statistically significantly better the compared baseline methods on both datasets (t-test, p-value $<$ 0.05).

| Method | EG-0 | nCG-0 | EG-1 | nCG-1 | GMP | Latency |
|---|---|---|---|---|---|---|
| IPS | 0.033 | 0.039 | 0.201 | 0.213 | -0.324 | 192344 |
| AP | 0.037 | 0.031 | 0.232 | 0.235 | -0.103 | 134077 |
| CST | 0.048 | 0.063 | 0.262 | 0.254 | -0.321 | 91456 |
| LS | 0.070 | 0.081 | 0.271 | 0.297 | -0.185 | 85665 |
| NNRL | 0.069 | 0.085 | 0.282 | 0.288 | -0.236 | 84343 |
| HRES | **0.082*** | **0.097*** | **0.309*** | **0.315*** | **-0.071*** | **70573*** |

Table A.2: Event summarization results on TREC-RTS. Numbers with * mean that improvement from our model is statistically significant over the baseline methods (t-test, p-value $<$ 0.05).

## References

Adam Roegiest, Luchen Tan, and Jimmy Lin. 2017. Online in-situ interleaved evaluation of real-time push notification systems. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pages 415–424.

| Method | EG-0 | nCG-0 | EG-1 | nCG-1 | GMP | Latency |
|---|---|---|---|---|---|---|
| IPS | 0.042 | 0.472 | 0.237 | 0.234 | -0.256 | 12632 |
| AP | 0.039 | 0.448 | 0.255 | 0.212 | -0.143 | 135334 |
| CST | 0.046 | 0.501 | 0.239 | 0.250 | -0.331 | 98045 |
| LS | 0.052 | 0.521 | 0.260 | 0.260 | -0.097 | 78433 |
| NNRL | 0.051 | 0.501 | 0.271 | 0.254 | -0.109 | 74257 |
| HRES | **0.063*** | **0.572*** | **0.293*** | **0.276** | **-0.087*** | **65916*** |

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