


Day 145: NLP Papers Summary – SUPERT: Towards New Frontiers In Unsupervised Evaluation Metrics For Multi-Document Summarization

Objective and Contribution

Proposed SUPERT, an unsupervised evaluation metric for evaluating multi-document summary by measuring the semantic similarity between the summary and the pseudo reference summary. The pseudo reference summary is generated by selecting salient sentences from the source documents using contextualised embeddings and soft token alignment. SUPERT was able to achieve a better correlation with human evaluation of 18 – 39%. We used SL  with an reinforcement learning summariser and it yielded a strong performance in comparison to

SOTA unsupervised summarisers. This showcase the effectiveness of SUPERT and it means that we can create many reference summaries from the infinite number of documents to increase size of dataset.

Datasets and Evaluation Metrics

We used two multi-document summarisation datasets: TAC'08 and TAC'09. Both TAC datasets consist of roughly 45+ topics and each topic has ten news articles, four reference summaries and 55+ machine-generated summaries. Our evaluation metrics are three different correlation coefficients: Pearson's, Spearman's, and Kendall's.

MODELS COMPARISON

1. *TFIDF*
2. *JS divergence*. Measures the JS divergence between word distributions in source and summaries
3. *REAPER*
4. *Cosine-ELMo*. Contextualised word embeddings
5. *Bohm19*
6. *ROUGE-1 and ROUGE-2 and MoverScore*. Upper bounds performance measure

SUmmarisation evaluation with Pseudo references and bERT (SUPERT)

SUPERT measures the relevance of multi-document summaries, which measures how much salient information is included in the summary from the source document. We measure relevance in two steps:

1. Identify salient sentences from the source document
2. Measuring the semantic overlap between the pseudo reference (step 1) and the generated summary



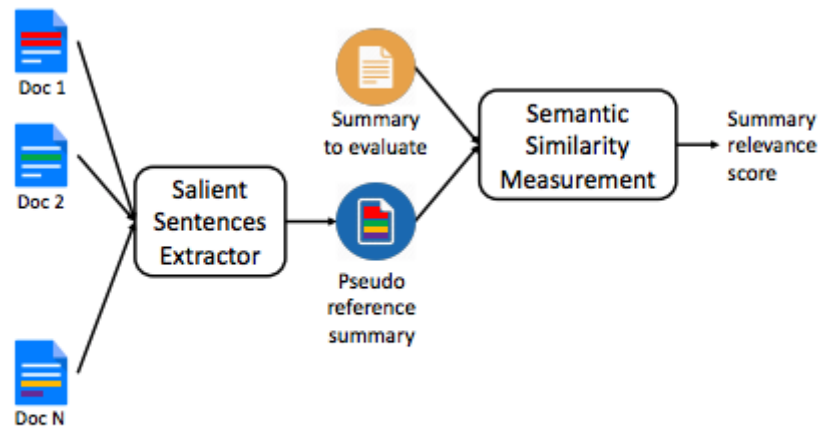


Figure 1: Workflow of SUPERT.

The results table below showcase how all the baseline methods performed significantly below the upper bound performance limit. Surprisingly, the embedding-based methods performed worse than the lexicon-based methods. This tells us that existing single document evaluation metrics are ineffective in evaluating multi-document summaries.

| | TAC'08 | | | TAC'09 | | |
|--------------------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | r | ρ | τ | r | ρ | τ |
| <i>Baselines (unsupervised evaluation)</i> | | | | | | |
| TF-IDF | .364 | .330 | .236 | .388 | .395 | .288 |
| JS | .381 | .333 | .238 | .388 | .386 | .283 |
| REAPER | .259 | .247 | .174 | .332 | .354 | .252 |
| C _{ELMo} | .139 | .108 | .076 | .334 | .255 | .183 |
| Böhm19 | .022 | -.001 | .001 | .075 | .043 | .031 |
| <i>Upper bounds (reference-based evaluation)</i> | | | | | | |
| Rouge1 | .747 | .632 | .501 | .808 | .692 | .533 |
| Rouge2 | .718 | .635 | .498 | .803 | .694 | .531 |
| Mover | .760 | .672 | .507 | .831 | .701 | .550 |

Table 1: Summary-level correlation between some popular evaluation metrics and human ratings. Unsupervised metrics (upper) measure the similarity between summaries and the source documents, while reference-based metrics (bottom) measure the similarity between summaries and human-written reference summaries.

MEASURING SIMILARITY WITH CONTEXTUALISED EMBEDDINGS

We extended cosine-ELMo by exploring different text encoders such as BERT, ROBERTa, ALBERT and SBERT with cosine similarity. The results are displayed below. As shown, SBERT as the text encoder with cosine similarity yielded the highest relevance generated : 🔄 ries. However, this still performed poorly against the lexicon-based methods. Another extension we

explored is the use of word mover's distances (WMDs) to measure semantic similarity between two documents instead of using cosine similarity. Previous work has proven that WMDs yielded a stronger performance and our results below supported that as WMD with SBERT (M_SBERT) significantly outperformed its cosine similarity counterparts and all the lexicon-based methods. This led us to our ultimate method for computing semantic similarity between documents, which it's to use SBERT and WMD.

| | TAC'08 | | | TAC'09 | | |
|----------------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | τ | ρ | τ | τ | ρ | τ |
| C _{BERT} | .035 | .066 | .048 | .130 | .099 | .071 |
| C _{RoBERTa} | .100 | .126 | .091 | .262 | .233 | .165 |
| C _{ALBERT} | .152 | .122 | .086 | .247 | .219 | .157 |
| C _{SBERT} | .304 | .269 | .191 | .371 | .319 | .229 |
| M _{RoBERTa} | .366 | .326 | .235 | .357 | .316 | .229 |
| M _{SBERT} | .466 | .428 | .311 | .436 | .435 | .320 |

Table 2: Performance of contextual-embedding-based metrics. Soft aligning the embeddings of the source documents and the summaries (the bottom part) yields higher correlation than simply computing the embeddings cosine similarity (the upper part).

BUILDING PSEUDO REFERENCES


Results from previous tables showcase a large difference in performance between unsupervised evaluation and reference-based evaluation. This argues that we still need reference summaries and so we explore different methods of building pseudo references.

Firstly, we explored two simple strategies to establish baseline results: choose N random sentences or top N sentences. The results are displayed below. The results showcase the poor performance of randomly selected sentences and we should be selecting the top 10 – 15 sentences as pseudo references as it outperformed the lexical-based and our M_SBERT methods. This also illustrate the position bias in news articles.



Secondly, we explored two graph-based approach to building pseudo references: position-agnostic and position-aware graphs. For position-agnostic graphs, we extended LexRank using SBERT (SLR) to measure the cosine similarity. We also explore the affinity propagation clustering algorithm (SC) which clusters the sentences and the center of each cluster is selected to build pseudo reference. This clustering algorithm doesn't require us to preset the number of clusters. For SLR and SC, we have two variations: individual graph and global graph. The individual graph builds a graph for each source document and selects top K sentences. The global graph builds a graph using all the sentences from all the source documents of the same topic and selects the top M sentences.

For position-aware graphs, we extended PacSum using SBERT (SPS) to measure sentences similarity and similarly, consider both individual and global-graph versions. PacSum selects sentences that are semantically central meaning it has high average similarity with succeeding sentences and low average similarity with preceding sentences. In addition, we also proposed Top + Clique (TC), which selects top N sentences and semantically central sentences to build pseudo references. Here's how TC works:

1. Label top N sentences from each document as salient
2. Build a graph that connects highly similar non-top-N sentences
3. Identify the cliques from the graph and select the semantically central sentence from each clique as potential salient sentences
4. For each potential salient sentence, compare it to the top N sentences and label it  salient if it's not highly similar to the top N sentences

The table below showcase the results of the position-agnostic and position-aware graphs. All the methods (except SC_G) outperformed the baseline models in table 1 above. Our position-agnostic graphs underperformed the position-aware graphs. In addition, our position-aware graphs underperformed simple sentence extraction method of selecting top N sentences in table 3. This shows us that the position bias is very strong in news and it remains the most effective approach in selecting the positive information.

GUIDING REINFORCEMENT LEARNING

We use our new unsupervised evaluation metric to guide the training of a RL-based multi-document summariser, Neural Temporal Difference (NTD). We considered three unsupervised reward functions: JS, REAPER, and SUPERT (SP). SUPERT selects the top 10 – 15 sentences from each source document as pseudo references and uses SBERT to measure semantic similarity between summaries and pseudo references. The results are shown below and NTD with SUPERT yielded the strongest results.



Source: <https://arxiv.org/pdf/2005.03724.pdf>



Ryan

Data Scientist

Previous Post

< Day 144: NLP
Papers Summary -
Attend to Medical
Ontologies:
Content Selection
for Clinical
Abstractive
Summarization

Next Post

Day 146: NLP
Papers Summary -
Exploring Content
Selection in
Summarization of
Novel Chapters





[Data Science](#) [Natural Language Processing](#) [NLP Papers Summary](#)

Day 365: NLP Papers Summary – A Survey on Knowledge Graph Embedding

Ryan

30th December 2020





[Data Science](#) [Implementation](#) [Natural Language Processing](#)

Day 364: Ryan's PhD Journey – OpenKE-PyTorch Library Analysis + code snippets for 11 KE models

Ryan

29th December 2020

