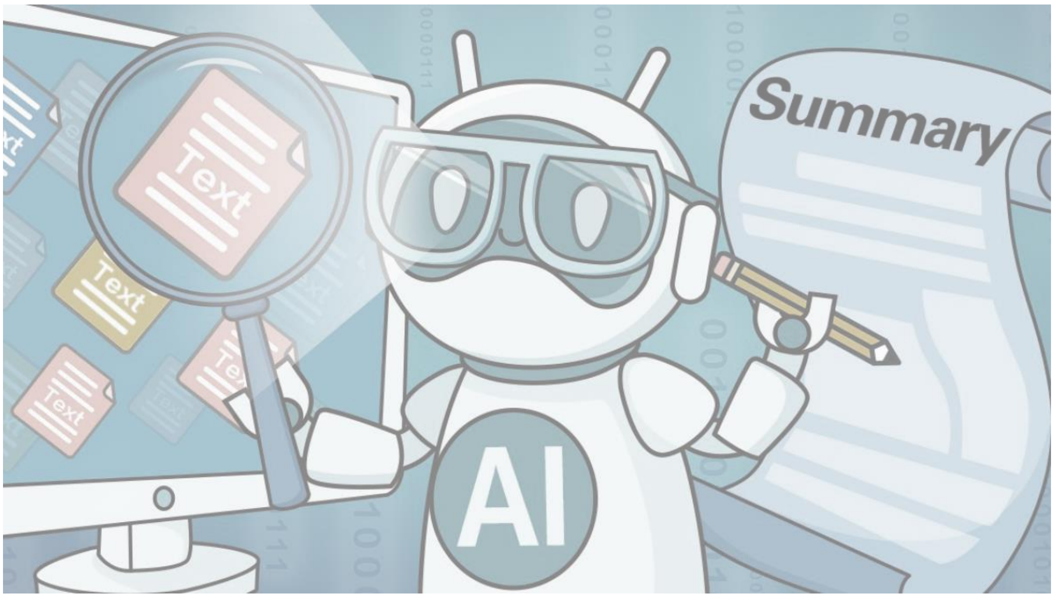


ENTITY-LEVEL FACTUAL CONSISTENCY OF ABSTRACTIVE TEXT SUMMARIZATION

GROUP NO : 05

CS/2017/041 – SIRIWARDHANA B.D.S.A

CS/2017/020 – MUTHUKUMARA M.G.L.

Entity-level Factual Consistency of Abstractive Text Summarization

Ensuring factual consistency of the generated summary with respect to the source document is a major challenge in abstractive text summarization. Recent researches reveal about 30% of the summaries generated by neural network sequence-to-sequence models have factual inconsistency. Factual inconsistency can happen at either the entity level or the relation level. This research paper focuses on entity-level factual consistency issue which is known as entity hallucination problem. An entity hallucination problem is named entities are that never contained in the source document that may be visible in the model generated summary. Although there is a widely used factual consistency metric which is known as the ROUGE score is inadequate to quantify factual consistency effectively. Three metrics are suggested to quantify factual consistency at the entity level in this research paper. The factual quality of summaries is analysed that is produced by the state-of-the-art BART model on three news data sets such as Newsroom, CNN/DailyMail and XSUM in this research. BART is a denoising autoencoder developed with a sequence-to-sequence model. Comparing to encoder-only pre-training such as in BERT or decoder-only pre-training such as in GPT-2, BART is an encoder-decoder transformer-based neural translation model that has been included five pre-taring techniques such as token masking, sentence permutation, document rotation, token deletion, and text infilling

Then certain strategies such as data filtering, multi-task learning and joint sequence generation are used to enhance performance on these metrics. The three metrics are proposed in this research are Precision-source(prec_s), Entity-based data filtering and Precision-target and recall-target. Also, these metrics rely on the Spacy tool to perform Named-Entity Recognition (NER).

Precision-source quantifies the degree of hallucination with respect to the source document.

$\text{prec}_s = N\ (h \cap s)/N(h)$

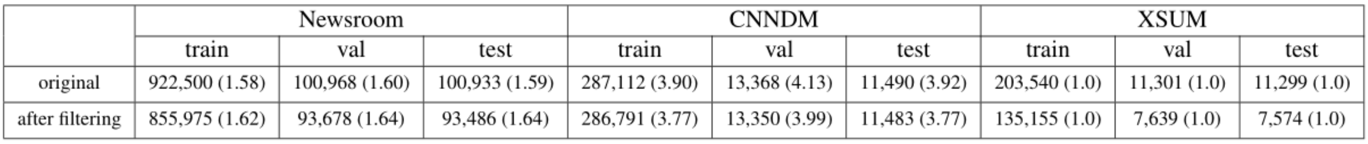
$N\ (h \cap s)$ is the number of entities in the target summary that can detect a match in the source document.

$N(h)$ is named entities in the generated summary.

Low prec_s means high hallucination.

Entity-based Data Filtering

If any of the entities in the generated summary which are not appear in the source document, The specific sentence that contains the entity is discarded. If there is only one sentence in the ground truth summary, it is required to be discarded and the document-summary pair is removed from the data set. Filtered data set that does not contain hallucination of entities ($\text{prec}_s = 1$) is ensured from this technique. Here is the example below to the respective data set.



	Newsroom			CNNDM			XSUM		
	train	val	test	train	val	test	train	val	test
original	922,500 (1.58)	100,968 (1.60)	100,933 (1.59)	287,112 (3.90)	13,368 (4.13)	11,490 (3.92)	203,540 (1.0)	11,301 (1.0)	11,299 (1.0)
after filtering	855,975 (1.62)	93,678 (1.64)	93,486 (1.64)	286,791 (3.77)	13,350 (3.99)	11,483 (3.77)	135,155 (1.0)	7,639 (1.0)	7,574 (1.0)

Precision-target (prec_t) and Recall-target (recall_t)

Precision target (prec_t) and Recall-target (recall_t) is used to obtain a complete picture of entity-level accuracy of the generated summary since the precision source(prec_s) metric do not capture the entity level accuracy of the generated summary. precision source quantifies the degree of entity hallucination concerning the source document.

$\text{prec}_t = N(h \cap t)/N(h)$

$\text{recall}_t = N(h \cap t)/N(t)$

$N(t)$ is the number of named entities in the target summary

$N(h \cap t)$ is the number of named entities in the generated summary that can find a match in the ground truth summary.

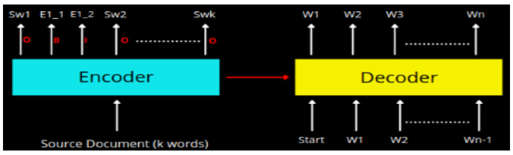
The F1 score can be computed as $F1_t = 2 \cdot \text{prec}_t \cdot \text{recall}_t /(\text{prec}_t + \text{recall}_t)$ to have a single quantifiable number.

Multi-Task Learning

Also, furthermore an enhancement to the entity based data filtering is explored which is a way of classifying summary-worthy named entities in the source document (encoder side). A summary worthy named entity from the source that is also present in the ground truth summary is a salient entity. To achieve this, they label every token in the original document with a BIO scheme (B-Begin, I-Inside, O-outside) This is a pretty standard technique for labelling single/multi-word entities in a text segment.

BOI scheme can be represented as mathematically below,

$$\mathcal{L}^i_{\text{BIO}}(\theta(\text{enc}), x^i, z^i) = -\sum_{t=1}^{ts(i)} \log p_{\theta(\text{enc})}(z^i_t|x^i)$$



Multi-task Learning

In the past, it has been shown that if a decoder was not able to model an entity's representation, it would lead to hallucinations. The idea behind this loss is that this loss would force the encoder to model its representation so that it captures relevant information about summary-worthy entities. When the decoder gets this entity enriched representation, it can be built more accurate generations with lesser hallucinations.

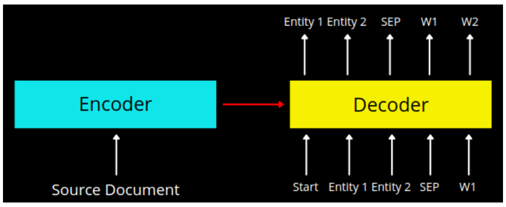
Apart from the BIO loss, they also use typical MLE loss for the training sequence generation model which would propagate loss from the decoder end.

$$\mathcal{L}^i_{\text{MLE}}(\theta, x^i, y^i) = -\sum_{t=1}^{tt(i)} \log p_\theta(y^i_t | x^i, y^i_{<t})$$

Here, theta, x, y, i are model parameters, input tokens, output tokens and ith token respectively. They minimize the joint loss L(i) = L(i)_MLE + αL(i)_ BIO, where a is a hyperparameter. They choose α between 0.1 to 0.5 based on the validation set.

Join Salient Entity and Summary Generation (JAENS)

The researchers have developed a new way of generating summaries, where they train a sequence model to generate the sequence of summary-worthy named entities, followed by a special token, and then the summary tokens instead of just generating the summary. The idea behind this approach is that while generating summary tokens, the decoder can attend to summary-worthy entities via the self-attention mechanism. This allows it to jointly learn to identify the descriptive entities while learning to generate summaries.



JAENS Architecture

Results

The table below reveals proposed exploration is great for hallucination alleviation. It demonstrates comparing models trained using original data with entity-based data filtering, with an additional classification task and with JAENS.

	training data	Rouge1	Rouge2	RougeL	macro $\textbf{prec}_s$	micro $\textbf{prec}_s$	macro $\textbf{prec}_t$	micro $\textbf{prec}_t$	macro $\textbf{recall}_t$	micro $\textbf{recall}_t$	macro $F1_t$	micro $F1_t$
Newsroom	original	47.7±0.2	35.0±0.3	44.1±0.2	97.2±0.1	97.0±0.1	65.4±0.3	62.9±0.4	70.8±0.3	68.5 ±0.2	68.0±0.2	65.6±0.3
	+ filtering	47.7±0.1	35.1±0.1	44.1 ±0.1	98.1±0.1	98.0±0.0	66.5±0.1	63.8±0.1	70.2 ±0.2	67.7±0.3	68.3±0.1	65.7±0.1
	+ classification	47.7±0.2	35.1±0.1	44.2±0.2	98.1±0.1	98.0±0.0	67.2±0.4	64.2±0.4	70.3±0.2	67.8±0.4	68.7±0.3	65.9±0.4
	JAENS	46.6 ±0.5	34.3±0.3	43.2±0.3	**98.3**±0.1	**98.3**±0.1	**69.5**±1.6	**67.3**±1.2	68.9±1.5	66.8±1.6	**69.2**±0.1	**67.0**±0.2
CNNDM	original	43.7±0.1	**21.1**±0.1	40.6±0.1	99.5±0.1	99.4±0.1	66.0±0.4	66.5±0.4	74.7±0.7	75.4±0.6	70.0±0.2	70.7±0.3
	+ filtering	43.4±0.2	20.8±0.1	40.3±0.2	**99.9**±0.0	**99.9**±0.0	66.2 ±0.4	66.6±0.3	74.1±0.6	74.9±0.6	69.9±0.2	70.5±0.2
	+ classification	43.5±0.2	20.8±0.2	40.4±0.2	**99.9**±0.0	**99.9**±0.0	**67.0**±0.6	**67.5**±0.5	74.7±0.2	75.5±0.1	70.6±0.3	71.3±0.3
	JAENS	42.4 ±0.6	20.2±0.2	39.5±0.5	**99.9**±0.0	**99.9**±0.0	**67.9**±0.7	**68.4**±0.6	**75.1**±0.7	**76.4**±0.7	**71.3**±0.2	**72.2**±0.2
XSUM	original	**45.6**±0.1	**22.5**±0.1	**37.2**±0.1	93.9±0.1	93.6±0.2	74.1±0.2	73.3±0.2	80.1±0.1	80.3±0.3	77.0±0.1	76.6±0.2
	+ filtering	45.4±0.1	22.2±0.1	36.9±0.1	98.2±0.0	98.2±0.1	77.9±0.2	77.3±0.2	79.4±0.2	79.6±0.2	78.6±0.1	78.4±0.2
	+ classification	45.3±0.1	22.1±0.0	36.9±0.1	98.3±0.1	98.2±0.1	78.6±0.3	**78.0**±0.3	79.5±0.3	79.8±0.4	**79.1**±0.1	**78.9**±0.1
	JAENS	43.4±0.7	21.0±0.3	35.5 ±0.4	**99.0**±0.1	**99.0**±0.1	77.6±0.9	77.1±0.6	79.5±0.6	80.0±0.5	78.5±0.2	78.5±0.1

data filtering facilitates higher prec$_s$ scores, and it reveals entity hallucination can be reduced by this simple technique. Data filtering mainly enhances other entity-level metrics: prec$_t$, recall$_t$ and F1$_t$. Including the classification task (multi-task) or JAENS to data filtering is a reason to further improvement of the performance on prec$_t$ and recall$_t$. The scores are shown in percentages, averages over 5 runs with standard deviations.