READ, HIGHLIGHT AND SUMMARIZE: A HIERARCHICAL NEURAL SEMANTIC ENCODER-BASED APPROACH

Anonymous authors
Paper under double-blind review

ABSTRACT

Traditional sequence-to-sequence (seq2seq) models and other variations of the attention-mechanism such as hierarchical attention have been applied to the text summarization problem. Though there is a hierarchy in the way humans use language by forming paragraphs from sentences and sentences from words, hierarchical models have usually not worked that much better than their traditional seq2seq counterparts. This effect is mainly because either the hierarchical attention mechanisms are too sparse using hard attention or noisy using soft attention. In this paper, we propose a method based on extracting the highlights of a document; a key concept that is conveyed in a few sentences. In a typical text summarization dataset consisting of documents that are 800 tokens in length (average), capturing long-term dependencies is very important, *e.g.*, the last sentence can be grouped with the first sentence of a document to form a summary. LSTMs (Long Short-Term Memory) proved useful for machine translation. However, they often fail to capture long-term dependencies while modeling long sequences. To address these issues, we have adapted Neural Semantic Encoders (NSE) to text summarization, a class of memory-augmented neural networks by improving its functionalities and proposed a novel hierarchical NSE that outperforms similar previous models significantly. The quality of summarization was improved by augmenting linguistic factors, namely lemma, and Part-of-Speech (PoS) tags, to each word in the dataset for improved vocabulary coverage and generalization. The hierarchical NSE model on factored dataset outperformed the state-of-the-art by nearly 4 ROUGE points. We further designed and used the first GPU-based self-critical Reinforcement Learning model.

1 INTRODUCTION

When there are a very large number of documents that need to be read in limited time, we often resort to reading summaries instead of the whole document. Automatically generating (abstractive) summaries is a problem with various applications, e.g., automatic authoring (Banerjee & Mitra, 2015). We have developed automatic text summarization systems that condense large documents into short and readable summaries. It can be used for both single (*e.g.*, Rush et al. (2015), See et al. (2017) and Nallapati et al. (2017)) and multi-document summarization (*e.g.*,Celikyilmaz et al. (2018), Nallapati et al. (2017), Henß et al. (2015)).

Text summarization is broadly classified into two categories: extractive (*e.g.*, Nallapati et al. (2017) and (Narayan et al., 2018)) and abstractive summarization (*e.g.*, Nallapati et al. (2016), Chopra et al. (2016) and Chen & Bansal (2018)). Extractive approaches select sentences from a given document and groups them to form concise summaries. By contrast, abstractive approaches generate human-readable summaries that primarily capture the semantics of input documents and contain rephrased key content. The former task falls under the classification paradigm, and the latter belongs to the generative modeling paradigm, and therefore, it is a much harder problem to solve. The backbone of state-of-the-art summarization models is a typical encoder-decoder (Sutskever et al., 2014) architecture that has proved to be effective for various sequential modeling tasks such as machine translation, sentiment analysis, and natural language generation. It contains an encoder that maps the raw input word vector representations to a latent vector. Then, the decoder usually equipped with a variant of the attention mechanism (Bahdanau et al., 2014) uses the latent vectors to generate the output sequence, which is the summary in our case. These models are trained in a supervised learning setting where we minimize the cross-entropy loss between the predicted and the target summary. Encoder-decoder models have proved effective for short sequence tasks such as machine translation where the length of a sequence is less than 120 tokens. However, in text summarization, the length of the sequences vary from 400 to 800 tokens, and modeling long-term dependencies becomes increasingly difficult.

Despite the metric's known drawbacks, text summarization models are evaluated using ROUGE (Lin, 2004), a discrete similarity score between predicted and target summaries based on 1-gram, 2-gram, and n-gram overlap. Cross-entropy loss would be a convenient objective on which to train the model since ROUGE is not differentiable, but doing so would create a mismatch between metrics used for training and evaluation. Though a particular summary scores well on ROUGE evaluation comparable to the target summary, it will be assigned lower probability by a supervised model. To tackle this problem, we have used a self-critic policy gradient method (Rennie et al., 2016) to train the models directly using the ROUGE score as a reward. In this paper, we propose an architecture that addresses the issues discussed above.

1.1 Problem Formulation

Let $D = \{d_1, d_2, ..., d_N\}$ be the set of document sentences where each sentence $d_i, 1 \leq i \leq N$ is a set of words and $S = \{s_1, s_2, ..., s_M\}$ be the set of summary sentences. In general, most of the sentences in D are a continuation of another sentence or related to each other, for example: in terms of factual details or pronouns used. So, dividing the document into multiple paragraphs as done by Celikyilmaz et al. (2018) leaves out the possibility of a sentence-level dependency between the start and end of a document. Similarly, abstracting a single document sentence as done by Chen & Bansal (2018) cannot include related information from multiple document sentences. In a good human-written summary, each summary sentence is a compressed version of a few document sentences. Mathematically,

$$\forall s \in S, \exists d_1, d_2, ..., d_K \in D, \mid C(d_1, d_2, ..., d_K) = s \tag{1}$$

Where C is a compressor we intend to learn. Figure 1 represents the fundamental idea when using a sequence-to-sequence architecture. For a sentence s in summary, the representations of all the related document sentences $d_1, d_2, ..., d_K$ are expected to form a cluster that represents a part of the highlight of the document.

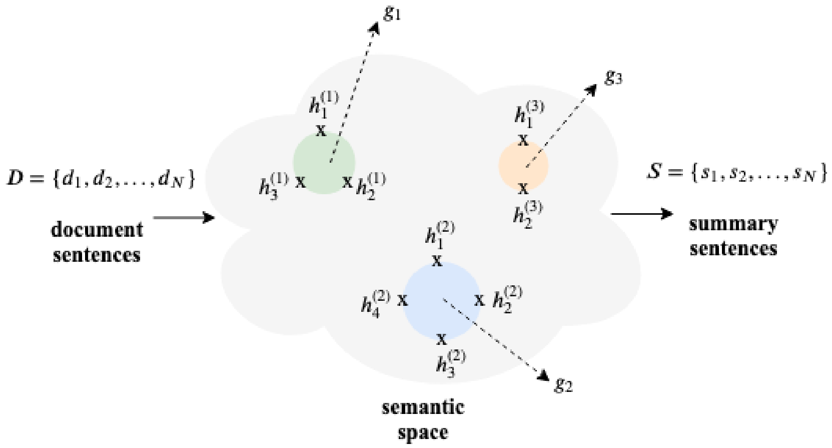

Figure 1: Document sentences are first projected into a semantic space typically by an encoder in a sequence-to-sequence model. g_1, g_2, g_3 are highlights of a document representing closely related sentence-semantics $\{h_1^{(1)}, h_2^{(1)}, h_3^{(1)}\}$, $\{h_1^{(2)}, h_2^{(2)}, h_3^{(2)}\}$, $\{h_1^{(3)}, h_2^{(3)}, h_3^{(3)}\}$ respectively. These highlights are then used by the decoder to form concise summaries.

First, we adapt the Neural Semantic Encoder (NSE) for text summarization by improving its attention mechanism and compose function. In a standard sequence-to-sequence model, the decoder has access to input sequence through hidden states of an LSTM (Hochreiter & Schmidhuber, 1997), which suffers from the difficulties that we discussed above. The NSE is equipped with an additional memory, which maintains a rich representation of words by evolving over time. We then propose a novel hierarchical NSE by using separate word memories for each sentence to enrich the word representations and a document memory to enrich the sentence representations, which performed better than its previous counterparts (Nallapati et al. (2016), Nallapati et al. (2017), Ling & Rush (2017)). Finally, we use a maximum-entropy self-critic model to achieve better performance using ROUGE evaluation.

2 Related Work

The first encoder-decoder for text summarziation is used by Rush et al. (2015) coupled with an attention mechanism. Though encoder-decoder models gave a state-of-the-art performance for Neural Machine Translation (NMT), the maximum sequence length used in NMT is just 100 tokens. Typical document lengths in text summarization vary from 400 to 800 tokens, and LSTM is not effective due to the loss in memory over time for very long sequences. Nallapati et al. (2016) used hierarchical attention(Yang et al., 2016) to mitigate this effect where, a word LSTM is used to encode (decode) words, and a sentence LSTM is used to encode (decode) sentences. The use of two LSTMs separately for words and sentences improves the ability of the model to retain its memory for longer sequences. Additionally, Nallapati et al. (2016) explored using a hierarchical model consisting of a feature-rich encoder incorporating position, Named Entity Recognition (NER) tag, Term Frequency (TF) and Inverse Document Frequency (IDF) scores. Since an RNN is a sequential model, computing at one time-step needs all of the previous time-steps to have computed before and is slow because the computation at all the time steps cannot be performed in parallel. Chopra et al. (2016) used convolutional layers coupled with an attention mechanism (Bahdanau et al., 2014) to increase the speed of the encoder. Since the input to an RNN is fed sequentially, it is expected to capture the positional information. But both works Nallapati et al. (2016) and Chopra et al. (2016) found positional embeddings to be quite useful for reasons unknown. Nallapati et al. (2017) proposed an extractive summarization model that classifies sentences based on content, saliency, novelty, and position. To deal with out-of-vocabulary (OOV) words and to facilitate copying salient information from input sequence to the output, See et al. (2017) proposed a pointer-generator network that combines pointing (Vinyals et al., 2015) with generation from vocabulary using a soft-switch. Attention models for longer sequences tend to be repetitive due to the decoder repeatedly attending to the same position from the encoder. To mitigate this issue, See et al. (2017) used a coverage mechanism to penalize a decoder from attending to same locations of an encoder. However, the pointer generator and the coverage model (See et al., 2017) are still highly extractive; copying the whole article sentences 35% of the time. Paulus et al. (2018) introduced an intra-attention model in which attention also depends on the predictions from previous time steps.

One of the main issues with sequence-to-sequence models is that optimization using the cross-entropy objective does not always provide excellent results because the models suffer from a mismatch between the training objective and the evaluation metrics such as ROUGE (Lin, 2004) and METEOR (Banerjee & Lavie, 2005). A popular algorithm to train a decoder is the teacher-forcing algorithm that minimizes the negative log-likelihood (cross-entropy loss) at each decoding time step given the previous ground-truth outputs. But during the testing stage, the prediction from the previous time-step is fed as input to the decoder instead of the ground truth. This exposure bias results in error accumulation over each time step because the model has never been exposed to its predictions during training. Instead, recent works show that summarization models can be trained using reinforcement learning (RL) where the ROUGE (Lin, 2004) score is used as the reward (Paulus et al. (2018), Chen & Bansal (2018) and Celikyilmaz et al. (2018)).

Henß et al. (2015) made such an earlier attempt by using Q-learning for single-and multi-document summarization. Later, Ling & Rush (2017) proposed a coarse-to-fine hierarchical attention model to select a salient sentence using sentence attention using REINFORCE (Williams, 1992) and feed it to the decoder. Narayan et al. (2018) used REINFORCE to rank sentences for extractive summarization. Celikyilmaz et al. (2018) proposed deep communicating agents that operate over small chunks of a document, which is learned using a self-critical (Rennie et al., 2016) training approach consisting of intermediate rewards. Chen & Bansal (2018) used a advantage actor-critic (A2C) method to extract sentences followed by a decoder to form abstractive summaries. Our model does not suffer from their limiting assumption that a summary sentence is an abstracted version of a single source sentence. Paulus et al. (2018) trained their intra-attention model using a self-critical policy gradient algorithm (Rennie et al., 2016). Though an RL objective gives a high ROUGE score, the output summaries are not readable by humans. To mitigate this problem, Paulus et al. (2018) used a weighted sum of supervised learning loss and RL loss.

Humans first form an abstractive representation of what they want to say and then try to put it into words while communicating. Though it seems intuitive that there is a hierarchy from sentence representation to words, as observed by both Nallapati et al. (2016) and Ling & Rush (2017), these hierarchical attention models failed to outperform a simple attention model (Rush et al., 2015). Unlike feedforward networks, RNNs are expected to capture the input sequence order. But strangely, positional embeddings are found to be effective (Nallapati et al. (2016), Chopra et al. (2016), Ling & Rush (2017) and Nallapati et al. (2017)). We explored a few approaches to solve these issues and improve the performance of neural models for abstractive summarization.

3 Proposed Models

In this section, we first describe the baseline Neural Semantic Encoder (NSE) class, discuss improvements to the compose function and attention mechanism, and then propose the Hierarchical NSE. Finally, we discuss the self-critic model that is used to boost the performance further using ROUGE evaluation.

3.1 Neural Semantic Encoder:

A Neural Semantic Encoder (Munkhdalai & Yu, 2017) is a memory augmented neural network augmented with an encoding memory that supports read, compose, and write operations. Unlike the traditional sequence-to-sequence models, using an additional memory relieves the LSTM of the burden to remember the whole input sequence. Even compared to the attention-model (Bahdanau et al., 2014) which uses an additional context vector, the NSE has anytime access to the full input sequence through a much larger memory. The encoding memory is evolved using basic operations described as follows:

$$o_t = f_{read}^{LSTM}(x_t) \tag{2}$$

$$z_t = softmax(o_t^T M_{t-1}) \tag{3}$$

$$m_{r,t} = z_t^T M_{t-1} \tag{4}$$

$$c_t = f_c^{MLP}(o_t, m_{t,t}) \tag{5}$$

$$h_t = f_w^{LSTM}(c_t) \tag{6}$$

$$M_t = M_{t-1}(\mathbf{1} - (z_t \otimes e_k)^T) + (h_t \otimes e_l)(z_t \otimes e_k)^T \tag{7}$$

Where, $x_t \in \mathbb{R}^D$ is the raw embedding vector at the current time-step. f_r^{LSTM} , f_c^{MLP} (Multi-Layer Perceptron), f_w^{LSTM} be the read, compose and write operations respectively. $e_l \in R^l$, $e_k \in R^k$ are vectors of ones, $\mathbf{1}$ is a matrix of ones and $\otimes$ is the outer product.

Instead of using the raw input, the read function f_r^{LSTM} in equation 2 uses an LSTM to project the word embeddings to the internal space of memory M_{t-1} to obtain the hidden states o_t. Now, the alignment scores z_t of the past memory M_{t-1} are calculated using o_t as the key with a simple dot-product attention mechanism shown in equation 3. A weighted sum gives the retrieved input memory that is used in equation 5 by a Multi-Layer Perceptron in composing new information. Equation 6 uses an LSTM and projects the composed states into the internal space of memory M_{t-1}

to obtain the write states h_t. Finally, in equation 7, the memory is updated by erasing the retrieved memory as per z_t and writing as per the write vector h_t. This process is performed at each time-step throughout the input sequence. The encoded memories $\{M\}_{t=1}^{T}$ are similarly used by the decoder to obtain the write vectors $\{h\}_{t=1}^{T}$ that are eventually fed to projection and softmax layers to get the vocabulary distribution.

3.2 Improved NSE

Although the vanilla NSE described above performed well for machine translation, just a dot-product attention mechanism is too simplistic for text summarization. In machine translation, it is sufficient to compute the correlation between word-vectors from the semantic spaces of different languages. In contrast, text summarization also needs a word-sentence and sentence-sentence correlation along with the word-word correlation. So, in search of an attention mechanism with a better capacity to model the complex semantic relationships inherent in text summarization, we found that the additive attention mechanism (Bahdanau et al., 2014) given by the equation below performs well.

$$z_t = softmax(v^T \tanh(WM_{t-1} + Uo_t + b_{attn})) \tag{8}$$

Where, v, W, U, b_{attn} are learnable parameters. One other important difference is the compose function: a Multi-layer Perceptron (MLP) is enough for machine translation as the sequences are short in length. However, text summarization consists of longer sequences that have sentence-to-sentence dependencies, and a history of previously composed words is necessary for overcoming repetition (Rush et al., 2015) and thereby maintaining novelty. A powerful function already at our disposal is the LSTM; we replaced the MLP with an LSTM, as shown below:

$$h_t = f_w^{LSTM}(c_t) \tag{9}$$

In a standard text summarization task, due to the limited size of word vocabulary, out-of-vocabulary (OOV) words are replaced with [UNK] tokens. pointer-networks (Vinyals et al., 2015) facilitate the ability to copy words from the input sequence to the output via pointing. Later, See et al. (2017) proposed a hybrid pointer-generator mechanism to improve upon pointing by retaining the ability to generate new words. It points to the words from the input sequence and generates new words from the vocabulary. A generation probability $p_{gen} \in (0, 1)$ is calculated using the retrieved memories, attention distribution, current input hidden state o_t and write state h_t as follows:

$$p_{gen} = \sigma(W_m^T m_{r,t} + W_h^T h_t + W_o^T o_t + b_{ptr}) \tag{10}$$

Where, W_m, W_h, W_o, b_{ptr} are learnable parameters, and σ is the sigmoid activation function. Next, p_{gen} is used as a soft switch to choose between generating a word from the vocabulary by sampling from p_{vocab}, or copying a word from the input sequence by sampling from the attention distribution z_t. For each document, we maintain an auxiliary vocabulary of OOV words in the input sequence. We obtain the following final probability distribution over the total extended vocabulary:

$$p(w) = p_{gen} p_{vocab} + (1 - p_{gen}) \sum_{i:w=w_i} z_i^t \tag{11}$$

Note that if w is an OOV word, then $p_{vocab}(w)$ is zero; similarly, if w does not appear in the source document, then $\sum_{i:w=w_i} z_i^t$ is zero. The ability to produce OOV words is one of the primary advantages of the pointer-generator mechanism. We can also use a smaller vocabulary size and thereby speed up the computation of output projection and softmax layers.

3.3 Hierarchical NSE

When humans read a document, we organize it in terms of word semantics followed by sentence semantics and then document semantics. In a text summarization task, after reading a document, sentences that have similar meanings or continual information are grouped together and then expressed in words. Such a hierarchical model was first introduced by Yang et al. (2016) for document classification and later explored unsuccessfully for text summarization (Nallapati et al., 2017). In

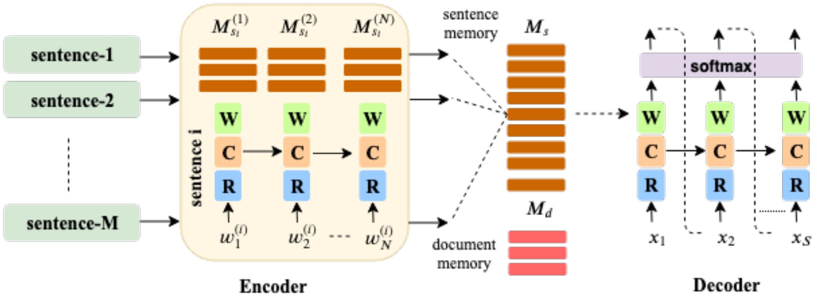

Figure 2: Hierarchical NSE: From a given article, all the M sentences consisting of N words each are processed by the NSE using read (R), compose (C) and write (W) operations. Each sentence memory is updated N times by each word in the sentence ($\{M_{s_i}^{(k)}\}_{k=1}^{N}$). After the last encoder step, all the updated sentence memories $M_{s_1}^N, M_{s_2}^N, ..., M_{s_M}^N$ are concatenated to form the cumulative sentence memory M_s. The decoder then uses the cumulative sentence memory M_s and document memory M_d in a similar fashion to produce the write vectors h_t that are passed through a softmax layer to obtain the vocabulary distribution.

this work, we propose to use a hierarchical model with improved NSE to take advantage of both augmented memory and also the hierarchical document representation. We use a separate memory for each sentence to represent all the words of a sentence and a document memory to represent all sentences. Word memory composes novel words, and document memory composes novel sentences in the encoding process that can be later used to extract highlights and decode to summaries as shown in Figure 2.

Let $D = \{(w_{ij})_{j=1}^{T_{in}}\}_{i=1}^{S_{in}}$ be the input document sequence, where S_{in} is the number of sentences in a document and T_{in} is the number of words per sentence. Let $\{M_i\}_{i=1}^{S_{in}}, M_i \in R^{T_{in} \times D}$ be the sentence memories that encode all the words in a sentence and $M^d, M^d \in R^{S_{in} \times D}$ be the document memory that encodes all the sentences present in the document. At each time-step, an input token x_t is read and is used to retrieve aligned content from both corresponding sentence memory $M_t^{i,s}$ and document memory M_t^d. Please note that the retrieved document memory, which is a weighted combination of all the sentence representations forms a highlight. After composition, both the sentence and document memories are written simultaneously. This way, the words are encoded with contextual meaning, and also new simpler sentences are formed. The functionality of the model is as follows:

$$o_t = f_r^{LSTM}(x_t) \tag{12}$$

$$z_t^s = f_{attn}(M_{t-1}^s, o_t) \tag{13}$$

$$z_t^d = f_{attn}(M_{t-1}^d, o_t) \tag{14}$$

$$m_{r,t}^s = z_t^s M_{t-1}^s \tag{15}$$

$$m_{r,t}^d = z_t^d M_{t-1}^d \tag{16}$$

$$c_t = f_c^{LSTM}(Concat(o_t, m_{r,t}^s, m_{r,t}^d)) \tag{17}$$

$$h_t = f_w^{LSTM}(c_t) \tag{18}$$

$$M_t^s = Update(M_{t-1}^s, z_t^s, h_t) \tag{19}$$

$$M_t^d = Update(M_{t-1}^d, z_t^d, h_t) \tag{20}$$

$$M^s = \begin{cases} M^{s_i}, 1 \leq i \leq S_{in} & \text{encoder-stage} \\ Concat(\{M^{s_i}\}_{i=1}^{S_{in}}) & \text{decoder-stage} \end{cases} \tag{21}$$

Where, f_{attn} is the attention mechanism given by equation(8). $Update$ remains the same as the vanilla NSE given by equation(7)and $Concat$ is the vector concatenation. Please note that NSE (Munkhdalai & Yu, 2017) has a concept of shared memory but we use multiple memories for representing words and a document memory for representing sentences, this is fundamentally different to a shared memory which does not have a concept of hierarchy.

3.4 Self-Critical Sequence Training

As discussed earlier, training in a supervised learning setting creates a mismatch between training and testing objectives. Also, feeding the ground-truth labels in training time-step creates an exposure bias while testing in which we feed the predictions from the previous time-step. Policy gradient methods overcome this by directly optimizing the non-differentiable metrics such as ROUGE (Lin, 2004) and METEOR (Banerjee & Lavie, 2005). It can be posed as a Markov Decision Process in which the set of actions $\mathcal{A}$ is the vocabulary and reward $\mathcal{R}$ is the ROUGE score itself. So, we should find a policy $\pi(\theta)$ such that the set of sampled words $\tilde{y} = \{\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_T\}$ achieves highest ROUGE score among all possible summaries.

We used the self-critical model of Rennie et al. (2016) proposed for image captioning. In self-critical sequence training, the REINFORCE algorithm (Williams, 1992) is used by modifying its baseline as the greedy output of the current model. At each time-step t, the model predicts two words: $\hat{y}_t$ sampled from $p(\hat{y}_t|\hat{y}_1, \hat{y}_2, ..., \hat{y}_{t-1}, x)$, the baseline output that is greedily generated by considering the most probable word from the vocabulary and $\tilde{y}_t$ sampled from the $p(\tilde{y}_t|\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_{t-1}, x)$. This model is trained using the following loss function:

$$L_{rl} = (r(\tilde{y}) - r(\hat{y})) \sum_{t=1}^{T} - \log(p(\tilde{y}_t|\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_{t-1}, x)) \tag{22}$$

Using the above training objective, the model learns to generate samples with high probability and thereby increasing $r(\tilde{y})$ above $r(\hat{y})$. Additionally, we have used enthttps://stackoverflow.com/questions/19053077/looping-over-data-and-creating-individual-figuresropy regularization.

$$\mathrm{H_t} = - \sum_{v=1}^{V} p(\tilde{y}_t = v) \log(p(\tilde{y}_t = v)) \tag{23}$$

$$L = L_{rl} - \alpha \sum_{t=1}^{T} H_t \tag{24}$$

Where, $p(\tilde{y}_t) = p(\tilde{y}_t|\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_{t-1}, x)$ is the sampling probability and V is the size of the vocabulary. It is similar to the exploration-exploitation trade-off. α is the regularization coefficient that explicitly controls this trade-off: a higher α corresponds to more exploration, and a lower α corresponds to more exploitation. We have found that all TensorFlow based open-source implementations of self-critic models use a function (**tf.py_func**) that runs only on CPU and it is very slow. To the best of our knowledge, ours is the first GPU based implementation.

4 Experiments and Results

4.1 Dataset

We used the CNN/Daily Mail dataset (Nallapati et al., 2016), which has been used as the standard benchmark to compare text summarization models. This corpus has 286,817 training pairs, 13,368 validation pairs, and 11,487 test pairs, as defined by their scripts. The source document in the training set has 766 words spanning 29.74 sentences on an average while the summaries consist of 53 words and 3.72 sentences (Nallapati et al., 2016). The unique characteristics of this dataset such as long documents, and ordered multi-sentence summaries present exciting challenges, mainly because the proven sequence-to-sequence LSTM based models find it hard to learn long-term dependencies in long documents. We have used the same train/validation/test split and examples for a fair comparison with the existing models.

The factoring of lemma and Part-of-Speech (PoS) tag of surface words, are observed (Bandyopadhyay, 2019) to increase the performance of NMT models in terms of BLEU score drastically. This is due to the improvement of the vocabulary coverage and better generalization. We have added a pre-processing step by incorporating the lemma and PoS tag to every word of the dataset and training the supervised model on the factored data. The process of extracting the lemma and the PoS tags has been described in Bandyopadhyay (2019). Please refer to the appendix for an example of factoring.

4.2 Training Settings

For all the plain NSE models, we have truncated the article to a maximum of 400 tokens and the summary to 100 tokens. For the hierarchical NSE models, articles are truncated to have a maximum of 20 sentences and 20 words per sentence each. Shorter sequences are padded with ‘PAD‘ tokens. Since the factored models have lemma, PoS tag and the separator ‘|‘ for each word, sequence lengths should be close to 3 times the non-factored counterparts. For practical reasons of memory and time, we have used 800 tokens per article and 300 tokens for the summary.

For all the models, including the pointer-generator model, we use a vocabulary size of 50,000 words for both source and target. Though some previous works (Nallapati et al., 2016) have used large vocabulary sizes of 150,000, since our models have a copy mechanism, smaller vocabulary is enough to obtain good performance. Large vocabularies increase the computation time. Since memory plays a prominent role in retrieval and update, it is vital to start with a good initialization. We have used 300-dimensional pre-trained GloVe (Pennington et al., 2014) word-vectors to represent the input sequence to a model. Sentence memories are initialized with GloVe word-vectors of all the words in that sentence. Document memories are initialized with vector representations of all the sentences where a sentence is represented with the average of the GloVe word-vectors of all its words. All the models are trained using the Adam optimizer with the default learning rate of 0.001. We have not applied any regularization as the usage of dropout, and L_2 penalty resulted in similar performance, however with a drastically increased training time.

The Hierarchical models process one sentence at a time, and hence attention distributions need less memory, and therefore, a larger batch size can be used, which in turn speeds up the training process. The non-factored model is trained on 7-NVIDIA Tesla-P100 GPUs with a batch size of 448 (64 examples per GPU); it takes approximately 45 minutes per epoch. Since the factored sequences are long, we used a batch size of 96 (12 examples per GPU) on 8-NVIDIA Tesla-V100 GPUs. The Hier model reaches optimal cross-entropy loss in just 8 epochs, unlike 33-35 epochs for both Nallapati et al. (2016) and See et al. (2017). For the self-critical model, training is started from the best supervised model with a learning rate of 0.00005 and manually changed to 0.00001 when needed with $\alpha = 0.0001$ and the reported results are obtained after training for 15 days.

Table 1: ROUGE F_1 scores on the test set. Our hierarchical (Hier-NSE) model outperform previous hierarchical and pointer-generator models. Hier-NSE-factor is the factored model and Hier-NSE-SC is the self-critic model.

Paradigm	Models	ROUGE (% F-score)		
		1	2	L
Supervised Learning	HierAttn (Nallapati et al., 2016)	32.75	12.21	29.01
	abstractive model (Nallapati et al., 2016)	35.46	13.30	32.65
	Pointer Generator (See et al., 2017)	36.44	15.66	33.42
	Pointer Generator + coverage (See et al., 2017)	39.53	17.28	36.38
	Hier-NSE (ours)	38.31	16.34	35.26
	Hier-NSE-factor (ours)	**45.58**	**26.81**	**41.17**
Reinforcement Learning	MLE+RL, with intra-attention (Paulus et al., 2018)	39.87	15.82	36.90
	DCA, MLE+RL (Celikyilmaz et al., 2018)	41.69	19.47	37.92
	Hier-NSE-SC (ours)	39.42	16.46	**36.93**

4.3 Evaluation

All the models are evaluated using the standard metric ROUGE; we report the F1 scores for ROUGE-1, ROUGE-2, and ROUGE-L, which quantitively represent word-overlap, bigram-overlap, and longest common subsequence between reference summary and the summary that is to be evaluated. The results are obtained using *pyrouge* package[1]. The performance of various models and our

[1] https://pypi.org/project/pyrouge/

improvements are summarized in Table 2. A direct implementation of NSE performed very poorly due to the simple dot-product attention mechanism. In NMT, a transformation from word-vectors in one language to another one (say English to French) using a mere matrix multiplication is enough because of the one-to-one correspondence between words and the underlying linear structure imposed in learning the word vectors (Pennington et al., 2014). However, in text summarization a word (sentence) could be a condensation of a group of words (sentences). Therefore, using a complex neural network-based attention mechanism proposed improved the performance. Both dot-product and additive (Bahdanau et al., 2014) mechanisms perform similarly for the NMT task, but the difference is more pronounced for the text summarization task simply because of the nature of the problem as described earlier. Replacing Multi-Layered Perceptron (MLP) in the NSE with an LSTM further improved the performance because it remembers what was previously composed and facilitates the composition of novel words. This also eliminates the need for additional mechanisms to penalize repetitions such as coverage (See et al., 2017) and intra-attention (Paulus et al., 2018). Finally, using memories for each sentence enriches the corresponding word representation, and the document memory enriches the sentence representation that help the decoder. Please refer to the appendix for a few example outputs. Table 1 shows the results in comparison to the previous methods. Our hierarchical model outperforms Nallapati et al. (2016) (HIER) by 5 ROUGE points. Our factored model achieves the new state-of-the-art (SoTA) result, outperforming Celikyilmaz et al. (2018) by almost 4 ROUGE points.

Table 2: Performance of various NSE models on CNN/Daily Mail corpus. Please note that the data is not factored here.

Model	ROUGE (% F-score)		
	1	2	L
Plain NSE	7.99	0.86	7.52
NSE - improved attention	25.47	8.96	24.01
NSE - improved compose	30.86	11.42	29.04
Hierarchical NSE	38.31	16.34	35.26

5 Conclusion

In this work, we presented a memory augmented neural network for the text summarization task that addresses the shortcomings of LSTM-based models. We applied a critical pre-processing step by factoring the dataset with inherent linguistic information that outperforms the state-of-the-art by a large margin. In the future, we will explore new sparse functions (Martins & Astudillo, 2016) to enforce strict sparsity in selecting highlights out of sentences. The general framework of pre-processing, and extracting highlights can also be used with powerful pre-trained models like BERT (Devlin et al., 2018) and XLNet (Yang et al., 2019).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014. URL `http://arxiv.org/abs/1409.0473`. cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

Saptarashmi Bandyopadhyay. Factored neural machine translation at LoResMT 2019. In *Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages*, pp. 68–71, Dublin, Ireland, 20 August 2019. European Association for Machine Translation. URL `https://www.aclweb.org/anthology/W19-6811`.

Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W05-0909`.

Siddhartha Banerjee and Prasenjit Mitra. WikiKreator: Improving Wikipedia stubs automatically. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 867–877, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1084. URL `https://www.aclweb.org/anthology/P15-1084`.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1662–1675, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1150. URL `https://www.aclweb.org/anthology/N18-1150`.

Yen-Chun Chen and Mohit Bansal. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 675–686, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1063. URL `https://www.aclweb.org/anthology/P18-1063`.

Sumit Chopra, Michael Auli, and Alexander M. Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 93–98, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1012. URL `https://www.aclweb.org/anthology/N16-1012`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL `http://arxiv.org/abs/1810.04805`.

Stefan Henß, Margot Mieskes, and Iryna Gurevych. A reinforcement learning approach for adaptive single- and multi-document summarization. In Bernhard Fisseni, Bernhard Schröder, and Torsten Zesch (eds.), *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology, GSCL 2015, University of Duisburg-Essen, Germany, 30th September - 2nd October 2015*, pp. 3–12. GSCL e.V., 2015. URL `http://gscl2015.inf.uni-due.de/wp-content/uploads/2016/02/GSCL-201503.pdf`.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL `http://dx.doi.org/10.1162/neco.1997.9.8.1735`.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W04-1013`.

Jeffrey Ling and Alexander Rush. Coarse-to-fine attention models for document summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pp. 33–42, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4505. URL `https://www.aclweb.org/anthology/W17-4505`.

André F. T. Martins and Ramón F. Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pp. 1614–1623. JMLR.org, 2016. URL `http://dl.acm.org/citation.cfm?id=3045390.3045561`.

Tsendsuren Munkhdalai and Hong Yu. Neural semantic encoders. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 397–407, Valencia, Spain, April 2017. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/E17-1038`.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The*

20th SIGNLL Conference on Computational Natural Language Learning, pp. 280–290, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/K16-1028. URL `https://www.aclweb.org/anthology/K16-1028`.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pp. 3075–3081. AAAI Press, 2017. URL `http://dl.acm.org/citation.cfm?id=3298483.3298681`.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1747–1759, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1158. URL `https://www.aclweb.org/anthology/N18-1158`.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=HkAClQgA-`.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL `https://www.aclweb.org/anthology/D14-1162`.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1179–1195, 2016.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 379–389, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1044. URL `https://www.aclweb.org/anthology/D15-1044`.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1073–1083, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1099. URL `https://www.aclweb.org/anthology/P17-1099`.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 27*, pp. 3104–3112. Curran Associates, Inc., 2014. URL `http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf`.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pp. 2692–2700, Cambridge, MA, USA, 2015. MIT Press. URL `http://dl.acm.org/citation.cfm?id=2969442.2969540`.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3-4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL `https://doi.org/10.1007/BF00992696`.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237, 2019. URL `http://arxiv.org/abs/1906.08237`.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1174. URL `https://www.aclweb.org/anthology/N16-1174`.

A Appendix

Figure 3 below shows the self-critical model. All the examples shown in Tables 3-8 are chosen as per the shortest article lengths available due to space constraints.

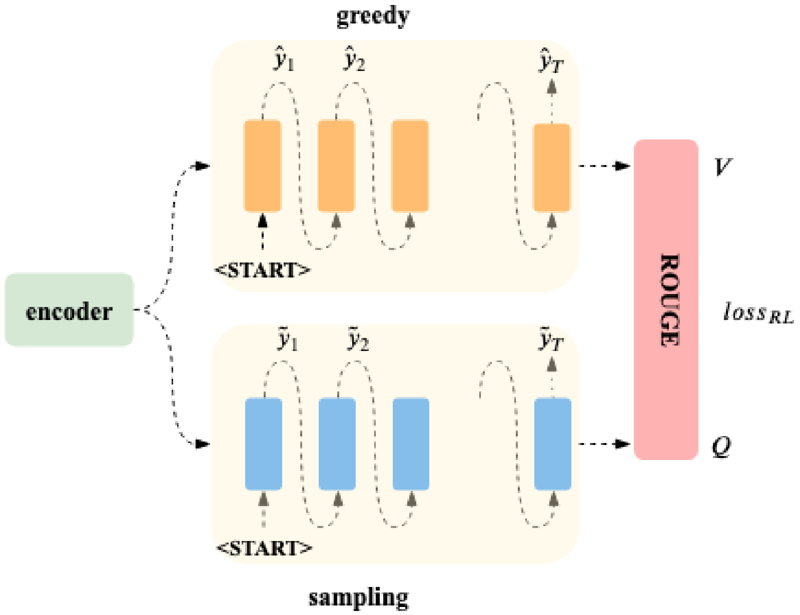

Figure 3: Self-Critic training reduces exposure bias and by learning a policy whose samples score better than the greedy samples that are used during test time in a supervised learning setting.

Table 3: Sample outputs for both non-factored and factored input articles. While factoring, each surface word is augmented with lemma and PoS tag separated by |.

<table>
<tr><td><b>Original Article</b></td></tr>
<tr><td>The build-up for the blockbuster fight between Floyd Mayweather and Manny Pacquiao in Las Vegas on May 2 steps up a gear on Tuesday night when the American holds an open workout for the media. The session will be streamed live across the world and you can watch it here from 12am UK.</td></tr>
<tr><td><b>Factored Article</b></td></tr>
<tr><td>The | the | DT build-up | build-up | NN for | for | IN the | the | DT blockbuster | blockbust | NN fight | fight | NN between | between | IN Floyd | floyd | NNP Mayweather | mayweath | NNP and | and | CC Manny | manni | NNP Pacquiao | pacquiao | NNP in | in | IN Las | la | NNP Vegas | vega | NNP on | on | IN May | may | NNP 2 | 2 | CD steps | step | NNS up | up | RB a | a | DT gear | gear | NN on | on | IN Tuesday | tuesday | NNP night | night | NN when | when | WRB the | the | DT American | american | NNP holds | hold | VBZ an | an | DT open | open | JJ workout | workout | NN for | for | IN the | the | DT media | media | NNS . | . | . The | the | DT session | session | NN will | will | MD be | be | VB streamed | stream | VBN live | live | JJ across | across | IN the | the | DT world | world | NN and | and | CC you | you | PRP can | can | MD watch | watch | VB it | it | PRP here | here | RB from | from | IN 12am | 12am | . | .</td></tr>
<tr><td><b>GT Summary</b></td></tr>
<tr><td>floyd mayweather holds an open media workout from 12am uk -lrb- 7pm edt -rrb- . the american takes on manny pacquiao in las vegas on may 2 . mayweather 's training is being streamed live across the world .</td></tr>
<tr><td><b>Hier-NSE output</b></td></tr>
<tr><td>the build-up for the blockbuster fight between floyd mayweather and manny pacquiao in las vegas on may 2 steps up a gear on tuesday night . the session will be streamed live across the world and you can watch it here from uk . the session will be the media 's open workout for the media .</td></tr>
<tr><td><b>Hier-NSE-SC</b></td></tr>
<tr><td>the floyd mayweather and manny pacquiao in las vegas . the american holds an open workout for the media . will be streamed live across the world and .</td></tr>
<tr><td><b>GT summary (factored)</b></td></tr>
<tr><td>floyd | floyd | nnp mayweather | mayweath | nnp holds | hold | vbz an | an | dt open | open | jj media | media | nns workout | workout | nn from | from | in 12am | 12am | cd uk | uk | nnp -lrb- | -lrb- | vbd 7pm | 7pm | cd edt | edt | nnp -rrb- | -rrb- | nn . the | the | dt american | american | jj takes | take | vbz on | on | in manny | manni | nnp pacquiao | pacquiao | nnp in | in | in las | la | nnp vegas | vega | nnp on | on | in may | may | nnp 2 | 2 | cd . mayweather | mayweath | nnp 's | 's | pos training | train | nn is | is | vbz being | be | vbg streamed | stream | vbn live | live | jj across | across | in the | the | dt world | world | nn .</td></tr>
<tr><td><b>Hier-NSE output (factored)</b></td></tr>
<tr><td>the | the | dt session | session | nn will | will | md be | be | vb streamed | stream | vbn live | live | jj across | across | in the | the | dt world | world | nn and | and | cc you | you | prp can | can | md watch | watch | vb it | it | prp here | here | rb from | from | in 12am | 12am | nnp nnp | nnp | nnp . the | the | dt american | american | nnp holds | hold | vbz an | an | dt open | open | jj workout | workout | nn for | for | in the | the | dt media | media | nns .</td></tr>
</table>

Table 4: Sample outputs for both non-factored and factored input articles. While factoring, each surface word is augmented with lemma and PoS tag separated by |.

<table>
<tr><td><b>Original Article</b></td></tr>
<tr><td>-LRB- CNN -RRB- Justin Timberlake and Jessica Biel , welcome to parenthood . The celebrity couple announced the arrival of their son , Silas Randall Timberlake , in statements to People . “ Silas was the middle name of Timberlake ’s maternal grandfather Bill Bomar , who died in 2012 , while Randall is the musician ’s own middle name , as well as his father ’s first , ” People reports . The couple announced the pregnancy in January , with an Instagram post . It is the first baby for both .</td></tr>
<tr><td><b>Factored Article</b></td></tr>
<tr><td>-LRB- | -lrb- | JJ CNN | cnn | NNP -RRB- | -rrb- | NNP Justin | justin | NNP Timberlake | timberlak | NNP and | and | CC Jessica | jessica | NNP Biel | biel | NNP , | , | , welcome | welcom | NN to | to | TO parenthood | parenthood | NN . | . | . The | the | DT celebrity | celebr | NN couple | coupl | NN announced | announc | VBD the | the | DT arrival | arriv | NN of | of | IN their | their | PRP son | son | NN , | , | , Silas | sila | NNP Randall | randal | NNP Timberlake | timberlak | NNP , | , | , in | in | IN statements | statement | NNS to | to | TO People | peopl | NNS . | . | . “ | “ | “ Silas | sila | NNP was | wa | VBD the | the | DT middle | middl | JJ name | name | NN of | of | IN Timberlake | timberlak | NNP ’s | ’s | POS maternal | matern | JJ grandfather | grandfath | NN Bill | bill | NNP Bomar | bomar | NNP , | , | , who | who | WP died | die | VBD in | in | IN 2012 | 2012 | CD , | , | , while | while | IN Randall | randal | NNP is | is | VBZ the | the | DT musician | musician | NN ’s | ’s | POS own | own | JJ middle | middl | NN name | name | NN , | , | , as | as | RB well | well | RB as | as | IN his | hi | PRP father | father | NN ’s | ’s | POS first | first | JJ , | , | , ” | ” | ” People | peopl | NNP reports | report | NNS . | . | . The | the | DT couple | coupl | NN announced | announc | VBD the | the | DT pregnancy | pregnanc | NN in | in | IN January | januari | NNP , | , | , with | with | IN an | an | DT Instagram | instagram | NNP post | post | NN . | . | . It | It | PRP is | is | VBZ the | the | DT first | first | JJ baby | babi | NN for | for | IN both | both | DT . — . | .</td></tr>
<tr><td><b>GT Summary</b></td></tr>
<tr><td>timberlake and biel welcome son silas randall timberlake . the couple announced the pregnancy in january .</td></tr>
<tr><td><b>Hier-NSE Output</b></td></tr>
<tr><td>“ silas was the middle name of timberlake ’s maternal grandfather bill bomar ’ the couple announced the pregnancy in january , with an instagram post . it is the first baby for both .</td></tr>
<tr><td><b>Hier-NSE-SC</b></td></tr>
<tr><td>justin timberlake and jessica biel the couple of their son , . silas randall timberlake , in . the first baby for both .</td></tr>
<tr><td><b>GT summary (factored)</b></td></tr>
<tr><td>timberlake | timberlak | nnp and | and | cc biel | biel | nnp welcome | welcom | vbp son | son | nn silas | sila | nnp randall | randal | nnp timberlake | timberlak | nnp . the | the | dt couple | coupl | nn announced | announc | vbd the | the | dt pregnancy | pregnanc | nn in | in | in january | januari | nnp .</td></tr>
<tr><td><b>Hier-NSE Output (factored)</b></td></tr>
<tr><td>justin | justin | nnp timberlake | nnp | nnp and | and | cc jessica | jessica | nnp nnp | nnp | nnp are | are | [UNK] in | in | in statements | statement | nns to | to | to people | peopl | nns . he | he | nnp is | is | vbz the | the | dt first | first | jj baby | vbz | nn for | for | in both | both | dt . timberlake | [UNK] | jj bill | bill | nn , | , | , the | the | dt couple | | nn ’s | ’s | pos son | son | nn .</td></tr>
</table>

Table 5: Sample outputs from the hierarchical NSE and self-critical model.

Original Article
-LRB- CNN -RRB- Once Hillary Clinton ’s official announcement went online , social media responded in a big way , with terms like “ Hillary Clinton , ” ” Hillary2016 , ” and yes , even ” WhyImNotVotingforHillary ” trending . Certainly , you could n’t go far on Twitter -LRB- even before Clinton tweeted her announcement -RRB- , without an opinion or thought on her new campaign -LRB- there were over 3 million views of her announcment tweets in one hour , and 750,000 Facebook video views so far by Sunday evening -RRB- . Some tweeted their immediate support , with one word :
GroundTruth Summary
response across social media led to multiple trending topics for hillary clinton ’s presidential announcement . some responded to her video and her new campaign logo .
Hier-NSE Output
hillary clinton tweeted her announcement without an opinion or thought on her new campaign . some tweeted their immediate support , with one word : “ hillary clinton , ” yes .
Hier-NSE-SC
hillary clinton ’s official announcement . clinton “ hillary clinton , ” . “ ’ ’ in the .

Table 6: Factored input and outputs for the same example used in Table 5.

<table>
<tr><th>Article (Factored)</th></tr>
<tr><td>-LRB- | -lrb- | JJ CNN | cnn | NNP -RRB- | -rrb- | NNP Once | onc | NNP Hillary | hillari | NNP Clinton | clinton | NNP ’s | ’s | POS official | offici | JJ announcement | announc | NN went | went | VBD online | onlin | NN , | , | , social | social | JJ media | media | NNS responded | respond | VBD in | in | IN a | a | DT big | big | JJ way | way | NN , | , | , with | with | IN terms | term | NNS like | like | IN “ | “ | “ Hillary | hillari | NNP Clinton | clinton | NNP , | , | , ” | ” | ” ” | ” | ” | | Hillary2016 | hillary2016 | NNP , | , | , ” | ” | ” and | and | CC yes | ye | UH , | , | , even | even | RB ” | ” | ” | | WhyImNotVotingforHillary | whyimnotvotingforhillari | NNP ” | ” | ” trending | trend | NN . | . | . Certainly | certainli | RB , | , | , you | you | PRP could | could | MD n’t | n’t | RB go | go | VB far | far | RB on | on | IN Twitter | twitter | NNP -LRB- | -lrb- | NNP even | even | RB before | befor | IN Clinton | clinton | NNP tweeted | tweet | VBD her | her | PRP announcement | announc | NN -RRB- | -rrb- | NN , | , | , without | without | IN an | an | DT opinion | opinion | NN or | or | CC thought | thought | NN on | on | IN her | her | PRP new | new | JJ campaign | campaign | NN -LRB- | -lrb- | NN there | there | EX were | were | VBD over | over | IN 3 | 3 | CD million | million | CD views | view | NNS of | of | IN her | her | PRP announcment | announc | JJ tweets | tweet | NNS in | in | IN one | one | CD hour | hour | NN , | , | , and | and | CC 750,000 | 750,000 | CD Facebook | facebook | NNP video | video | NN views | view | NNS so | so | RB far | far | RB by | by | IN Sunday | sunday | NNP evening | even | VBG -RRB- | -rrb- | NN . | . | . Some | some | DT tweeted | tweet | VBD their | their | PRP immediate | immedi | JJ support | support | NN , | , | , with | with | IN one | one | CD word | word | NN : | : | :</td></tr>
<tr><th>GT summary (factored)</th></tr>
<tr><td>response | respons | nnp across | across | in social | social | jj media | media | nns led | led | vbd to | to | to multiple | multipl | vb trending | trend | vbg topics | topic | nns for | for | in hillary | hillari | nnp clinton | clinton | nnp ’s | ’s | pos presidential | presidenti | jj announcement | announc | nn . some | some | dt responded | respond | vbd to | to | to her | her | prp video | video | nn and | and | cc her | her | prp new | new | jj campaign | campaign | nn logo | logo | nn .</td></tr>
<tr><th>Hier-NSE Output (factored)</th></tr>
<tr><td>hillary | nnp | nnp clinton | clinton | nnp ’s | ’s | pos official | [UNK] | jj announcement | announc | nn went | went | vbd online | onlin | nn . clinton | clinton | nnp tweeted | tweet | vbd her | her | prp new | new | jj campaign | campaign | nn , | , | , without | without | in an | an | dt opinion | opinion | nn or | or | cc thought | thought | vbd on | on | in twitter | twitter | nn , | , | , with | with | in terms | term | nns like | like | in “ | “ | “ hillary | nnp | nnp clinton | clinton | nnp , | , | , ” | ” | ” | | nnp | nnp | jj .</td></tr>
</table>

Table 7: Sample outputs from the hierarchical NSE and self-critical model.

Original Article
Blackpool are in talks to sign Austria defender Thomas Piermayr . The 25-year-old has been training with the Championship club this week and they are keen to get him on board for what is expected to be confirmed as a campaign in League One next season . Piermayr is a free agent and had been playing for Colorado Rapids . The former Austria U21 international had a spell with Inverness Caledonian Thistle in 2011 . Thomas Piermayr -LRB- left , in action for the Colorado Rapids -RRB- tries to tackle Obafemi Martins last year
GroundTruth Summary
thomas piermayr has been training with blackpool this week . austrian defender is a free agent after leaving mls side colorado rapids . blackpool are bottom of the championship and look set to be relegated .
Hier-NSE Output
thomas has been training with the championship club this week . the former austria u21 international had a spell with inverness caledonian thistle . blackpool are in talks to sign austria defender thomas .
Hier-NSE-SC
blackpool are in talks to sign austria defender thomas . has been training with the championship club this week . is a free agent and .

Table 8: Factored input and outputs for the same example used in Table 7.

<table>
<tr><th>Factored Article</th></tr>
<tr><td>Blackpool | blackpool | NNP are | are | VBP in | in | IN talks | talk | NNS to | to | TO sign | sign | VB Austria | austria | NNP defender | defend | NN Thomas | thoma | NNP Piermayr | piermayr | NNP . | . | . The | the | DT 25-year-old | 25-year-old | JJ has | ha | VBZ been | been | VBN training | train | VBG with | with | IN the | the | DT Championship | championship | NNP club | club | NN this | thi | DT week | week | NN and | and | CC they | they | PRP are | are | VBP keen | keen | JJ to | to | TO get | get | VB him | him | PRP on | on | IN board | board | NN for | for | IN what | what | WP is | is | VBZ expected | expect | VBN to | to | TO be | be | VB confirmed | confirm | VBN as | as | IN a | a | DT campaign | campaign | NN in | in | IN League | leagu | NNP One | one | NNP next | next | JJ season | season | NN . | . | . Piermayr | piermayr | NNP is | is | VBZ a | a | DT free | free | JJ agent | agent | NN and | and | CC had | had | VBD been | been | VBN playing | play | VBG for | for | IN Colorado | colorado | NNP Rapids | rapid | NNP . | . | . The | the | DT former | former | JJ Austria | austria | NNP U21 | u21 | NNP international | intern | JJ had | had | VBD a | a | DT spell | spell | NN with | with | IN Inverness | inver | NNP Caledonian | caledonian | NNP Thistle | thistl | NNP in | in | IN 2011 | 2011 | CD . | . | . Thomas | thoma | NNP Piermayr | piermayr | NNP -LRB- | -lrb- | NNP left | left | VBD , | , | , in | in | IN action | action | NN for | for | IN the | the | DT Colorado | colorado | NNP Rapids | rapid | NNP -RRB- | -rrb- | NNP tries | tri | VBZ to | to | TO tackle | tackl | VB Obafemi | obafemi | NNP Martins | martin | NNP last | last | JJ year | year | NN</td></tr>
<tr><th>GT summary (factored)</th></tr>
<tr><td>thomas | thoma | nnp piermayr | piermayr | nnp has | ha | vbz been | been | vbn training | train | vbg with | with | in blackpool | blackpool | nnp this | thi | dt week | week | nn . austrian | austrian | jj defender | defend | nn is | is | vbz a | a | dt free | free | jj agent | agent | nn after | after | in leaving | leav | vbg mls | ml | nnp side | side | nn colorado | colorado | nnp rapids | rapid | nnp . blackpool | blackpool | nnp are | are | vbp bottom | bottom | nn of | of | in the | the | dt championship | championship | nnp and | and | cc look | look | vb set | set | vbn to | to | to be | be | vb relegated | releg | vbn .</td></tr>
<tr><th>Hier-NSE Output (factored)</th></tr>
<tr><td>the | the | dt 25-year-old | 25-year-old | jj has | ha | vbz been | been | vbn training | train | nnp with | with | in the | the | dt championship | championship | nnp club | club | nnp this | thi | dt week | week | nn . the | the | dt former | former | jj austria | austria | nnp u21 | u21 | nnp international | intern | jj had | had | vbd a | a | dt spell | spell | nn with | with | in nnp | nnp | nnp nnp | nnp | nnp .</td></tr>
</table>