GENERATION OF TEXT FROM STRUCTURED DATA

Anusha Prakash
anushap@andrew.cmu.edu

Mansi Gupta
mansig1@andrew.cmu.edu

Raghuveer Chanda
rchanda@andrew.cmu.edu

Srividya Potharaju
spothara@andrew.cmu.edu

ABSTRACT

Neural models have shown significant progress in the generation of short descriptions of text over unstructured data. We wish to tackle text generation using structured data like tables, info-boxes, databases, etc to present meaningful information. We use a dataset, ROTOWIRE, that consists of basketball game statistics and corresponding document level textual description for each game. We use neural encoder-decoder model, and experiment with variations of encoder units, attention and copy mechanism. We implement hierarchical attention with pointer generator network to leverage the hierarchical structure of the table and copy elements from the table. We compare our results with the state of the art conditional copy and joint copy mechanism. We find that although neural models perform well in generating fluent summaries, they perform poorly in capturing facts from the table.

1 INTRODUCTION

A classic problem in Natural Language Generation (NLG) is generating text that adequately and clearly describes the structured data, such as table, graph, info-box or any other knowledge base. According to Mei et al. this form of NLG imposes two goals, *what to say*, the selection of relevant subset of the input data to present, and, *how to say it*, the surface realization of a generation.

Generating documents using structured data is significantly harder than generating short summaries, where the majority of work in this area lie. The generated text might not be able to capture all the important facts in the structured input and contains factual errors or hallucinate factual statements. Techniques like attention and copy mechanism help to mitigate these challenges, but only to a limited extent.

In the ROTOWIRE dataset that we use, the input is a set of records and hence it cannot be naturally modeled as a sequence. To encode all the inputs into a single encoding vector, fixed sized models whose memory does not increase on increasing size of input, such as MLP, or strictly sequential models, such as RNNs can not be used. An important invariance property that must be satisfied is that swapping two elements x_i and x_j in the set X should not alter its encoding.

Secondly, the input is structured hierarchically. Each record has a type, for instance, PLAYER_NAME or POINTS, and the corresponding key-value pair. There are NUM_PLAYERS number of records under each type. The idea is that while generating the output, the decoder should first decide the type, for instance, number of goals, team name, points, etc and then decide which player scored the maximum score or goals.

To tackle these challenges, we use an order invariant encoder that takes the input records and produces an embedding per record, and LSTM decoder attends over the encoder as specified by Wiseman et al.. To leverage the hierarchical structure of the input which the above models are not capable of, we propose a hierarchical attention (Yang et al. (2016)) based pointer-generator network (See et al. (2017)) that can copy words from the input table via pointing, which aids accurate reproduction of information, while retaining the ability to produce novel words through the generator. The decoder attends over the type, as well as the key-value pair. We compare the proposed approach with the state of the art joint and conditional copy mechanism over it.

2 Related Work

Traditionally, content selection and surface realization have been handled as two separate modules. For instance, Liang et al. propose a hierarchical hidden semi-markov model where, first, records are chosen and ordered, then fields are chosen for each record, and finally, words are chosen for each field. They also experimented with the ROTOWIRE dataset primarily.

More recently, neural model based approaches have been used for the task of generating text from structured data. Decoding using recurrent networks, attention over the structured data and copy mechanism are some common components of these models.

Chisholm et al. use sequence-to-sequence model with attention to generate single sentence descriptions from facts derived from Wikidata slot-value pairs. Liu et al. go a step further and propose a novel structure-aware seq2seq model to encode both the content and the structure of a table. They introduce field-gating encoder in the GRU unit. They employ hierarchical (global and local) attention in the description generator to attend on the facts. They experiment on the WIKIBIO dataset and show that their model is capable of generating coherent and informative descriptions.

Although neural systems perform fairly well in generating fluent outputs, Wiseman et al. find that they perform quite poorly in terms of content selection and capturing long-term structure on a more challenging dataset like RotoWire. The authors experiment with combinations of encoder-decoder networks with extensions like, copy mechanism such as joint copy and conditional copy; additional loss terms such as reconstruction loss and total variation distance (TVD) over the predicted distributions and finally, varying size of beam search for decoding. They use a simple non-neural template-based approach as their baseline. Neural models improve significantly over baseline on BLEU scores, that particularly measures fluency and coherence of generated text, but they perform poorly in objective fact-based metrics.

These limitations motivate us to use RotoWire dataset for the task of structured data to document generation and improve upon the results obtained by Wiseman et al.. We find that the authors do not leverage the hierarchical structure of the input, which might improve the facts presented in the summary. We derive the idea of Hierarchical Pointer Generator model for this task from a combination of previous work in hierarchical attention networks and Pointer Generator network.

Hierarchical attention networks have been previously used for document classification tasks in Yang et al. (2016). As documents form a hierarchical structure, (words form sentences, sentences form a document), and different words and sentences are differentially informative. Hence, their model includes two levels of attention mechanisms one at the word level and one at the sentence level that lets the model pay more or less attention to individual words and sentences when constructing the representation of the document. We can use this method to pay first pay attention to the *type* and then to the key-value pair.

Pointer-generator networks introduced by See et al. (2017) have provided a viable new approach for abstractive text summarization. These networks can copy words from the source text via pointing, which aids accurate reproduction of information, while retaining the ability to produce novel words through the generator. It makes sense to use this approach in our problem as the network has shown to reduce factual inaccuracies, which is our primary goal as well.

3 DATA OVERVIEW

Most of the above mentioned neural approaches are tested on WIKIBIO dataset that contain only few records per examples and short, single sentence summaries. In contrast, the datasets, ROTOWIRE and SBNATION used by Wiseman et al., are more challenging in terms of larger number of records per example, larger number of unique records, longer target texts and larger vocabulary size. Hence, along with fluency metrics like BLEU, we also use a set of extractive evaluation metrics. The data statistics are shown in the Table 1.

Category	RotoWire
Vocabulary	11.3K
Tokens	1.6M
Examples	4.9K
Avg Len	337.1
Record Types	39
Average Records	628
Max Records	664

Table 1: Statistics of ROTOWIRE dataset

The ROTOWIRE dataset that we use for this task uses professionally written, medium length game summaries which are targeted at basketball fans who are primarily interested in game statistics. Figure 1 shows an instance of input and output pair from the ROTOWIRE dataset.

Let $s = \{r_i\}_{i=1}^{S}$ be a set of records, where for each $r \in s$ we define $r.t \in T$ to be the type of r, and we assume each r to be a binarized relation, where $r.e$ and $r.v$ are a records entity and value, respectively. So a record r is characterized by $\{r.t, r.e, r.v\}$. For example, a record statistics for a basketball game might have a record r such that $r.t =$ `POINTS`, $r.e =$ `RUSSELL WESTBROOK`, and $r.v = 50$. In this case, $r.e$ gives the player in question, and $r.v$ gives the number of points the player scored. Each *type* consists of `NUM_PLAYERS=31` players each.

From these records, we plan to generate a descriptive text, $\hat{y}_{1:T} = \hat{y}_1, ..., \hat{y}_T$ of T words such that $\hat{y}$ is an adequate and fluent summary of s. Training dataset consists of (s, y) pairs where y is a document consisting of a gold summary for databases.

TEAM	WIN	LOSS	PTS	FG_PCT	RB	AS ...
Heat	11	12	103	49	47	27
Hawks	7	15	95	43	33	20

PLAYER	AS	RB	PT	FG	FGA	CITY ...
Tyler Johnson	5	2	27	8	16	Miami
Dwight Howard	4	17	23	9	11	Atlanta
Paul Millsap	2	9	21	8	12	Atlanta
Goran Dragic	4	2	21	8	17	Miami
Wayne Ellington	2	3	19	7	15	Miami
Dennis Schroder	7	4	17	8	15	Atlanta
Rodney McGruder	5	5	11	3	8	Miami
Thabo Sefolosha	5	5	10	5	11	Atlanta
Kyle Korver	5	3	9	3	9	Atlanta
...						

The Atlanta Hawks defeated the Miami Heat , 103 - 95 , at Philips Arena on Wednesday . Atlanta was in desperate need of a win and they were able to take care of a shorthanded Miami team here . Defense was key for the Hawks , as they held the Heat to 42 percent shooting and forced them to commit 16 turnovers . Atlanta also dominated in the paint , winning the rebounding battle , 47 - 34 , and outscoring them in the paint 58 - 26.The Hawks shot 49 percent from the field and assisted on 27 of their 43 made baskets . This was a near wire - to - wire win for the Hawks , as Miami held just one lead in the first five minutes . Miami (7 - 15) are as beat - up as anyone right now and it 's taking a toll on the heavily used starters . Hassan Whiteside really struggled in this game , as he amassed eight points , 12 rebounds and one blocks on 4 - of - 12 shooting ...

Figure 1: An example data-record and document pair from the ROTOWIRE dataset. We show a subset of a game's records (out of 628 records), and a selection from the gold document.

4 EXPERIMENTAL SETUP

4.1 EVALUATION METRICS

To realize the goals of subset selection and surface realization, an end-to-end system should perform the tasks of content selection (CS), content ordering (CO) and precisely generating the relations (RG) from the database effectively. Hence, along with fluency metrics like BLEU and METEOR, these three tasks are the basis of evaluation for text generation using structured input (Wiseman et al.). Facts are extracted from input records, gold summary and generated summary and the following metrics are calculated[1] -

- **Content Selection (CS):** Precision and recall of number of unique relations extracted from gold summary that are present in the generated summary. It measures how well generated text matches actual text in terms of unique relations.
- **Relation Generation (RG):** Precision and recall of number of unique relation extracted from generated summary that are present in the input records. It measures how well the system generates text containing factual records.
- **Content Ordering (CO):** It is the normalized Damerau-Levenshtein Distance between relation extracted from gold and generated summary. It measures how well the system orders records it chooses to discuss.

4.2 BASELINE

We start with a base attention-based encoder-decoder model. For S input records, we map each record $r \in s$ into a single vector **e**. We concatenate the embeddings of $r.t$, $r.e$, and $r.v$ and then apply a 1-layer MLP to convert them into a vector **e** of size 600. Our input records are then represented as $E = \{e_j\}_{j=1}^{S}$. Given E, an LSTM decoder attends over these embeddings to compute the probability of each target word, conditioned on the previous words and on E. The model is trained end-to-end to minimize the negative log-likelihood of the words in the gold text $y_{1:T}$. We call this model as Linear.

We then replace the MLP unit of the encoder with simple RNN and BiLSTM unit, keeping the decoder same as before. Note that all these models are order invariant with respect to the input. We observe that BiLSTM encoder performs the best, and hence we use that for our further experiments.

4.3 COPY MECHANISM

Additionally, we implement two variations of copy mechanism (Gu et al.), joint copy and conditional copy, which form the state of the art on this dataset. Copy mechanism introduces a binary vector z of size T, that decides whether to copy the generated word from source or not. The words are copied from the value portion of a record r; that is, $z_t = 1$ implies $\hat{y} = r.v$ for some r and t.

$$p(\hat{y}_t, \hat{z}_t | \hat{y}_{1:t-1}, s) \propto \begin{cases} copy(\hat{y}_t, \hat{y}_{1:t-1}, s) & z_t = 1, \hat{y}_t \in s \\ 0 & z_t = 1, \hat{y}_t \notin s \\ gen(\hat{y}_t, \hat{y}_{1:t-1}, s) & z_t = 0 \end{cases} \tag{1}$$

While Joint Copy mechanism parameterize the joint distribution table over $\hat{y}$ and z_t directly while the Conditional Copy mechanism decomposes the joint probability.

$$p(\hat{y}_t, \hat{z}_t | \hat{y}_{1:t-1}, s) = \begin{cases} p_{copy}(\hat{y}_t | z_t, \hat{y}_{1:t-1}, s).p(z_t | \hat{y}_{1:t-1}, s) & z_t = 1 \\ p_{gen}(\hat{y}_t | z_t, \hat{y}_{1:t-1}, s).p(z_t | \hat{y}_{1:t-1}, s) & z_t = 0 \end{cases} \tag{2}$$

We note here that the key distinction between the Joint Copy model and the Conditional Copy model is that the latter conditions on the output whether there is a copy or not, and hence, in p_{copy},

[1] the evaluation scripts are taken from https://github.com/harvardnlp/data2text/blob/master/extractor.lua

the source records compete only with each other. In the Joint Copy model, however, the source records also compete with the words that cannot be copied. The proposed Pointer Generator network described in the following section, soft matches the distribution for copying the input and distribution for generating the output.

4.4 PROPOSED APPROACH - HIERARCHICAL POINTER GENERATOR NETWORK

On the top of the baseline model, we implement Hierarchical Pointer Generator model. The encoder uses the embeddings, E to generate local and global encoded hidden representation. Local representations are created for each input record whereas global representations are created for each block or type only. The meta-data about the game, such as city in which the game was played, the teams that are playing the game, etc, is represented as one block, while all other types that consists of 31 players are represented as one block per type.

At each timestep, the decoder attends over the local and global encoder hidden representations respectively to generate local and global attention vectors. Let hl_i be the local sequence and hg_i be the global sequence of encoder hidden states, and s_{t-1} be the previous word emitted by the decoder, the corresponding local and global attentions are calculated as -

$$al_i^t = softmax(tanh(Wl_h hl_i + Wl_s s_{t-1} + bl_{attn})) \tag{3}$$

$$ag_i^t = softmax(\nu * tanh(Wg_h hg_i + Wg_s s_{t-1} + bg_{attn})) \tag{4}$$

where ν, Wl_h, Wl_s, bl_{attn}, Wg_h, Wg_s and bg_{attn} are learnable parameters.

Next, hadamard product (element-wise product) of local and global attention distribution is calculated, and is used to produce a weighted sum of the encoder hidden states, known as the *context vector* h_t^* -

$$h_t^* = \sum_i \{al_i^t h_i \circ ag_i^t h_i\} \tag{5}$$

where $\circ$ implies hadamard product.

The context vector, h_t^*, which can be seen as a fixed size representation of what has been read from the source for this step, is concatenated with the decoder state s_t and fed through two linear layers to produce the vocabulary distribution P_{vocab} -

$$P_{vocab} = softmax(V^{'}(V(s_{t-1} \oplus h_t^*) + b) + b^{'}) \tag{6}$$

where $\oplus$ implies concatenation, $V^{'}$, V, b, $b^{'}$ are learnable parameters and P_{vocab} is a probability distribution over all words in the vocabulary, and provides us with our final distribution from which to predict words w.

Next, the generation probability $p_{gen} \in [0, 1]$ for timestep t is calculated from the context vector h_t, the decoder state s_{t-1} and the decoder input x_t -

$$p_{gen} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr}) \tag{7}$$

where vectors w_h, w_s, w_x and scalar b_{ptr} are learnable parameters and σ is the sigmoid 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 a_t. For each document let the extended vocabulary denote the union of the vocabulary, and all words appearing in the source document. We obtain the following probability distribution over the extended vocabulary -

$$P(w) = (p_{gen})(P_{vocab}(w)) + (1 - p_{gen})(\sum_{i:w_i=w} a_i^t) \tag{8}$$

Figure 2 gives the logical view of the above mechanism.

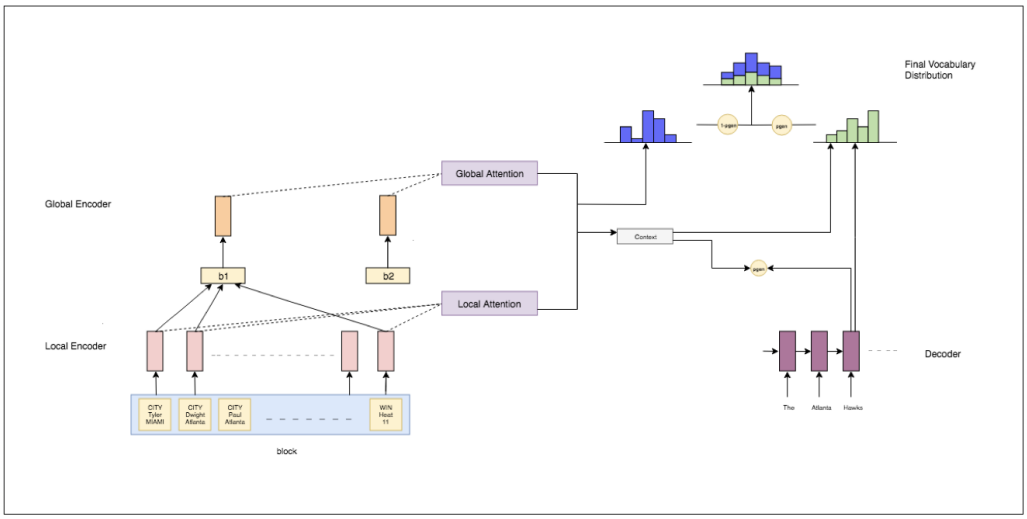

Figure 2: Hierarchical Attention with Pointer Generator Network

4.5 Parameters used

The paramters used for training the proposed model are mentioned in table 2. We trained the network for 220 epochs using Adam optimizer with default parameters. For the baseline models of copy and joint copy the model was trained with a batch size of 16 and the Encoder's hidden dimension and word embedding size of 600.

Parameter	Value
Embedding dimension	600
Hidden Dimension	1800
Number of Layers	4
Max players	31
Max blocklength	800
Gradient Clipping	5
Batch Size	2
Learning Rate	0.01
Epochs	220
beam size	5

Table 2: Parameters used for training

5 Results and Analysis

Table 3 shows the results of three base attention-based encoder-decoder models. Among the three, Linear, RNN and BiLSTM units, we observe that BiLSTM gives the best precision and recall for Content Selection as it encodes the entity, type and value together into a single vector more efficiently. The BLEU scores are acceptable for all the three models and we also observe the generated text to be very close to natural English sentences. Since the BLEU score is highest for BiLSTM encoder, we use it in the further implementations.

Model	RG (P% / R%)	CS (P% / R%)	CO	BLEU
Linear	0.29 / 0.08	0.13 / 0.20	0.23	8.87
RNN	0.41 / 0.08	0.24 / 0.30	0.28	6.95
BiLSTM	0.32 / 0.11	0.29 / 0.41	0.30	8.96
Joint copy	62.7 / 16.5	21.4 / 31.8	10.4	13.61
Conditional copy	75.0 / 16.7	25.9 / 31.9	11.0	14.19
Hierarchical Attention Pointer Networks	**61.07 / 12.12**	**20.2 / 29.5**	**9.98**	**11.56**

Table 3: Results of baseline models, state of the art copy models, and proposed Hierarchical Attention Pointer model on the ROTOWIRE dataset

The precision and recall of the extractive evaluation metrics are very low, implying that the model is not able to generate the facts properly. We also observe the same in the generated text where lot of noun phrases are repeated and incorrect match score values are generated. For example, few of the generated sentences mention that a single team defeating itself i.e "`The Phoenix Suns ( 10 - 7 ) defeated the Phoenix Suns`". So the generated sentences are grammatically correct but factually incorrect. A similar example is provided in the Table 4 where the initial sentence matches but all of the remaining facts don't match with the original text.

Type	Text
Original Text	The Atlanta Hawks (46 - 12) beat the Orlando Magic (19 - 41) 95 - 88 on Friday. Al Horford had a good all - around game , putting up 17 points , 13 rebounds , four assists and two steals in a tough matchup against Nikola Vucevic .... They blew a seven point lead with 43 seconds remaining and they might have carried that with them into Friday 's contest against the Hawks .... The Magic will host the Charlotte Hornets on Sunday , and the Hawks with take on the Heat in Miami on Saturday.
Generated Text	The Atlanta Hawks (6 - 2) defeated the Orlando Magic (7 - 12) 104 - 98 on Wednesday at the Amway Center in Atlanta . The Hawks got off to a quick start in this one , out - scoring the Magic 36 - 18 in the first quarter alone .... The Hawks also secured a 43 - point advantage into the half , while the Magic went just 42 percent from the floor and a meager 26 percent from beyond the arc .... The Magic 's next game will be at home against the Atlanta Hawks on Wednesday , while the Magic will be at home against the Atlanta Hawks on Wednesday .

Table 4: Comparison between gold output and generated text for given set of input records. The blue text shows the parts the generated output gets factually correct and red shows the parts that are factually incorrect.

We then use copy mechanism to obtain the state of the art results as described in the original paper. The extractive evaluation metrics shows a huge improvement thus strongly support our hypothesis that copy mechanism is best suitable for generating text with a lot of factual data and applying just simple attention is not sufficient. It can also be observed that Conditional Copy has a superior performance compared to Joint Copy indicating that relationships are often generated incorrectly in Joint Copy.

As indicated by the results, the performance of the proposed Hierarchical Pointer Generator Network is much superior to the base attention-based encoder-decoder models. But, the performance is very comparable to the conditional and joint copy models. This can be attributed to the fact that just conditioning on hierarchical attention might not be enough to perform better than the joint / conditional copy models. Rather, a more aggressive strategy of copying between levels, that is hierarchical copy might be needed, which we intend to experiment with in future.

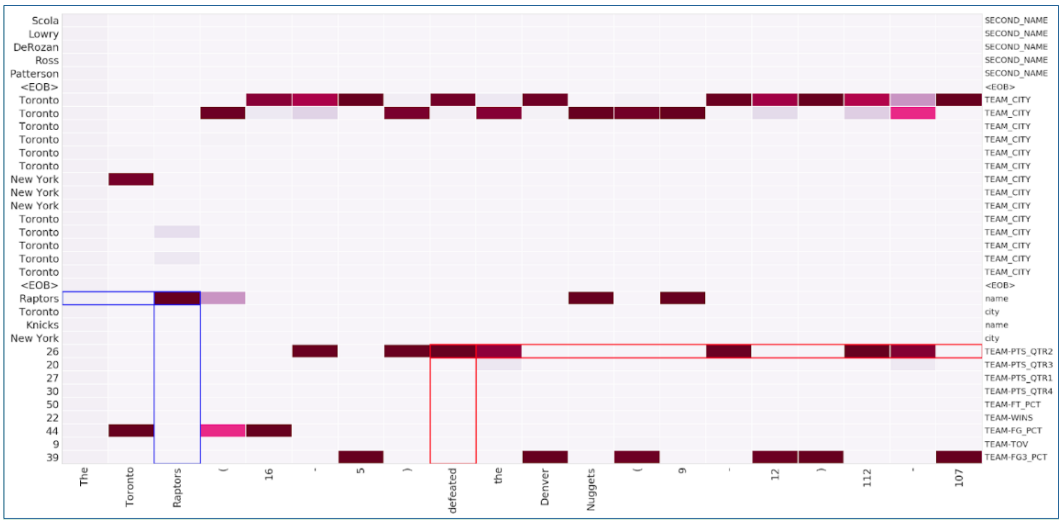

Figure 3: Visualization of combines global and local attention weights

Attention weights learnt by our model are visualized in Figure 3 where the Y-axis represents input tuples with values on left (local attention) and types on right global attention). The X-axis represents the generated sentence from the decoder. The blocks are colored as per the value of combined (local + global) attention. Notice that the output `Raptors` is generated by attending on the value `Raptors`, indicated by the local attention, while the output `defeated` is generated by attending on `TEAM_PTS_QTR2`, indicated by the global attention, as it can be concluded that a team defeated another team based on points in second quarter only. These instances show that our network has learnt attention well.

6 Conclusion and Future Work

The problem of document generation using nothing but the match statistics is significantly challenging. Despite adopting pointer generator networks and hierarchical attention mechanism the facts are not represented accurately in the generated summaries. To copy values at different levels more efficiently, we plan to incorporate hierarchical copy network in addition to hierarchical attention mechanism. Also, we observe that a common problem in the generated summaries is repetitive sentences. To tackle this, we wish to add *coverage* mechanism as described by See et al. (2017). They use a *coverage vector* to keep track of what has been generated and add a penalty term in loss to penalize repetition. Another thing that we would like to experiment with is to pretrain the decoder with the language model of the output.

References

Ai-writer-data2doc. `https://github.com/gau820827/AI-writer_Data2Doc`, 2018.

Andrew Chisholm, Will Radford, and Ben Hachey. Learning to generate one-sentence biographies from wikidata. *arXiv preprint arXiv:1702.06235*, 2017.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*, 2016.

Percy Liang, Michael I Jordan, and Dan Klein. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pp. 91–99. Association for Computational Linguistics, 2009.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. Table-to-text generation by structure-aware seq2seq learning. *arXiv preprint arXiv:1711.09724*, 2017.

Hongyuan Mei, Mohit Bansal, and Matthew R Walter. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. *arXiv preprint arXiv:1509.00838*, 2015.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

Sam Wiseman, Stuart M Shieber, and Alexander M Rush. Challenges in data-to-document generation. *arXiv preprint arXiv:1707.08052*, 2017.

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, 2016.