A Mixed Hierarchical Attention based Encoder-Decoder Approach for Standard Table Summarization

Parag Jain[†] **Anirban Laha**[†*] **Karthik Sankaranarayanan**[†]
Preksha Nema[*] **Mitesh M. Khapra**[*‡] **Shreyas Shetty**[*]
[†]IBM Research   [*]IIT Madras, India
[‡] Robert Bosch Center for Data Science and Artificial Intelligence, IIT Madras
{pajain34,anirlaha,kartsank}@in.ibm.com
{preksha,miteshk,shshett}@cse.iitm.ac.in

Abstract

Structured data summarization involves generation of natural language summaries from structured input data. In this work, we consider summarizing structured data occurring in the form of tables as they are prevalent across a wide variety of domains. We formulate the *standard table summarization* problem, which deals with tables conforming to a single predefined schema. To this end, we propose a *mixed hierarchical attention* based encoder-decoder model which is able to leverage the structure in addition to the content of the tables. Our experiments on the publicly available WEATHERGOV dataset show around 18 BLEU ($\sim 30\%$) improvement over the current state-of-the-art.

1 Introduction

Abstractive summarization techniques from structured data seek to exploit both structure and content of the input data. The type of structure on the input side can be highly varied ranging from key-value pairs (e.g. WIKIBIO (Lebret et al., 2016)), source code (Iyer et al., 2016), ontologies (Androutsopoulos et al., 2014; Colin et al., 2016), or tables (Wiseman et al., 2017), each of which require significantly varying approaches. In this paper, we focus on generating summaries from tabular data. Now, in most practical applications such as finance, healthcare or weather, data in a table are arranged in rows and columns where the schema is known beforehand. However, change in the actual data values can necessitate drastically different output summaries. Examples shown in the figure 1 have a predefined schema obtained from the WEATHERGOV dataset (Liang et al., 2009) and its corresponding weather report summary. Therefore, the problem that we seek to address in this paper is to generate abstractive summaries of tables conforming to a predefined fixed schema (as opposed to cases where the schema is unknown). We refer to this setting as **standard table summarization** problem. Another problem that could be formulated is one in which the output summary is generated from multiple tables as proposed in a recent challenge (Wiseman et al., 2017) (this setting is out of the scope of this paper). Now, as the schema is fixed, simple rule based techniques (Konstas and Lapata, 2013) or template based solutions could be employed. However, due to the vast space of selection (which attributes to use in the summary based on the current value it takes) and generation (how to express these selected attributes in natural language) choices possible, such approaches are not scalable in terms of the number of templates as they demand hand-crafted rules for both selection and generation.

We attempt to solve the problem of standard table summarization by leveraging the hierarchical nature of fixed-schema tables. In other words, rows consist of a fixed set of attributes and a table is defined by a set of rows. We cast this problem into a **mixed hierarchical attention model** following the encode-attend-decode (Cho et al., 2015) paradigm. In this approach, there is static attention on the attributes to compute the row representation followed by dynamic attention on the rows, which is subsequently fed to the decoder. This formulation is theoretically more efficient than the fully dynamic hierarchical attention framework followed by Nallapati et al. (2016). Also, our model does not need sophisticated sampling or sparsifying techniques like (Ling and Rush, 2017; Deng et al., 2017), thus, retaining differentiability. To demonstrate the efficacy of our approach, we transform the publicly available WEATHERGOV dataset (Liang et al., 2009) into fixed-schema tables, which is then used for our experiments. Our proposed *mixed hierarchi-*

Input : Table 1

TYPE	TIME	MIN	MAX	MEAN	MODE	MB100-4	MB20-2
temperature	17-30	33	60	44	-	-	-
windChill	17-30	0	47	31	-	-	-
windSpeed	17-30	5	9	7	-	-	0-10
windDir	17-30	-	-	-	NNW	-	-
gust	17-30	0	0	0	-	-	-
skyCover	17-30	-	-	-	-	0-25	-
skyCover	17-21	-	-	-	-	0-25	-
skyCover	17-26	-	-	-	-	0-25	-
skyCover	21-30	-	-	-	-	0-25	-
skyCover	26-30	-	-	-	-	0-25	-
precipPotential	17-30	0	0	0	-	-	-

Input : Table 2

TYPE	TIME	MIN	MAX	MEAN	MODE	MB100-4	MB20-2
temperature	6-21	35	53	46	-	-	-
windChill	6-21	0	47	27	-	-	-
windSpeed	6-21	6	8	7	-	-	0-10
windDir	6-21	-	-	-	WSW	-	-
gust	6-21	0	0	0	-	-	-
skyCover	6-21	-	-	-	-	75-100	-
skyCover	6-9	-	-	-	-	75-100	-
skyCover	6-13	-	-	-	-	50-75	-
skyCover	9-21	-	-	-	-	75-100	-
skyCover	13-21	-	-	-	-	75-100	-
precipPotential	6-21	4	48	17	-	-	-
rainChance	13-21	-	-	-	Chc	-	-

Clear , with a low around 33 . North wind between 5 and 9 mph becoming calm .

Output : Summary 1

A slight chance of showers after 3pm . Mostly cloudy ,with a high near 54 . Southwest wind between 6 and 8 mph . Chance of precipitation is 20 % .
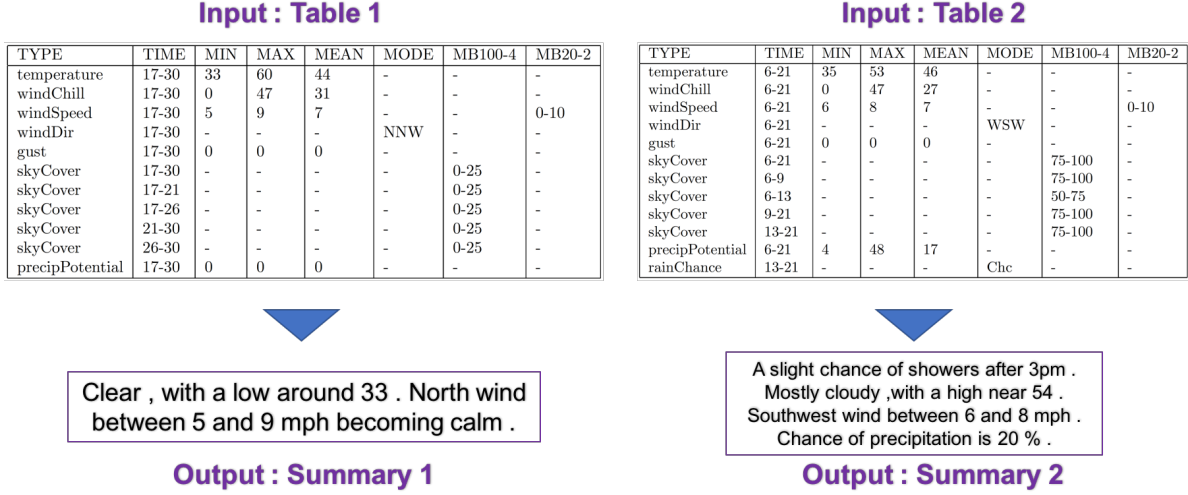
Output : Summary 2

Figure 1: Standard Table Summarization with fixed schema tables as input.

cal attention model provides an **improvement of around 18 BLEU (around 30%) over the current state-of-the-art result** by Mei et al. (2016).

2 Tabular Data Summarization

A standard table consist of set of records (or rows) $R = (r_1, r_2, ...r_T)$ and each record r has a fixed set of attributes (or columns) $A^r = (a_{r1}, a_{r2}, ...a_{rM})$. Tables in figure 1 have 7 columns (apart from 'TYPE') which correspond to different attributes. Also $U = (u_1, u_2, ...u_T)$ represents the type of each record where u_k is one-hot encoding for the record type for record r_k. Training data consists of instance pairs (X_i, Y_i) for $i = 1, 2, ..n$, where $X_i = (R_i, U_i)$ represents the input table and $Y_i = (y_1, ..., y_{T'})$ represents the corresponding natural language summary. In this paper, we propose an end-to-end model which takes in a table instance X to produce the output summary Y. This can be derived by solving in Y the following conditional probability objective:

$$Y^* = \arg\max_Y \prod_{t=1}^{T'} p(y_t|y_1, ..., y_{t-1}, X) \quad (1)$$

2.1 Mixed Hierarchical Attention Model (MHAM)

Our model is based on the *encode-attend-decode* paradigm as defined by Cho et al. (2015). It consists of an *encoder* RNN which encodes a variable length input sequence $\mathbf{x} = (x_1, ..., x_T)$ into a representation sequence $\mathbf{c} = (c_1, ..., c_T)$. Another decoder RNN generates sequence of output symbols
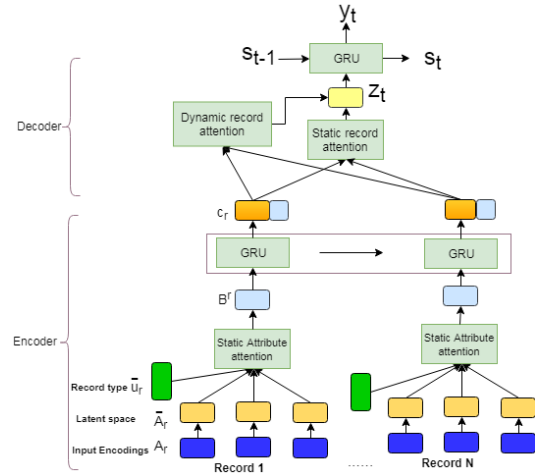

Figure 2: Proposed Architecture

$\mathbf{y} = (y_1, ..., y_{T'})$, attending to different combinations of c_i while generating different y_t.

As illustrated in figure 2, our encoder is not a single RNN. The encoder has a hierarchical structure to leverage the structural aspect of a *standard table*: a table consists of a set of records (or rows) and each record consists of values corresponding to a fixed set of attributes. We call it a *mixed hierarchical attention* based encoder, as it incorporates static attention and dynamic attention at two different levels of the encoder. At the record level, the attention over record representations is *dynamic* as it changes with each decoder time step. Whereas at the attribute level, since the schema is fixed, a record representation can be computed without the need of varying attention over attributes - thus *static* attention is used. For

623

example, with respect to WEATHERGOV dataset, a *temperature* record will always be defined by the attributes like min, max and mean irrespective of the decoder time step. So, attention over attributes can be static. On the other hand, while generating a word by the decoder, there can be a preference of focusing on a record type say, *temperature*, over some other type say, *windSpeed*. Thus, dynamic attention is used across records.

[!ht]

$$\bar{A}_j^r = W_j A_j^r, \ \forall j \in [1, M] \qquad (2)$$

$$I_j^r = \bar{A}_j^r \odot \bar{u}_r \quad , \quad \bar{u}_r = W_0 u_r \qquad (3)$$

$$\alpha_j^r = softmax_j(I^r) \qquad (4)$$

$$B^r = \sum_j \alpha_j^r \bar{A}_j^r \qquad (5)$$

$$c_r = [h_r; B^r] \quad , \quad h_r = GRU(B^r) \qquad (6)$$

$$g_r = \sigma(q^T tanh(P c_r)) \qquad (7)$$

$$\beta_t^r = v^T tanh(W_s s_{t-1} + W_c c_r) \qquad (8)$$

$$w_t^r = softmax_r(\beta_t) \qquad (9)$$

$$z_t = \sum_r \gamma_t^r c_r \quad , \quad \gamma_t^r = \frac{g_r w_t^r}{\sum_r g_r w_t^r} \qquad (10)$$

$$s_t = GRU(z_t, s_{t-1}) \qquad (11)$$

$$l_t = W_1 s_t + W_2 z_t + b_l \qquad (12)$$

$$p_t = softmax(l_t) \qquad (13)$$

Capturing attribute semantics: We learn record type embeddings and use them to calculate attentions over attributes. *For the trivial case of all records being same type, it boils down to having a single record type embedding.* Given attributes A^r for a record r, where each attribute a_i^r is encoded into a vector A_i^r based on the attribute type (discussed further in section 3), using equation 2 we embed each attribute where W_j is the embedding matrix for j^{th} attribute. We embed record type one-hot vector u_r through W_0, which is used to compute the importance score I_j^r for attribute j in record r according to equation 3.

Static Attribute attention: Not all attribute values contribute equally to the record. Hence, we introduce attention weights for attributes of each record. These attention weights are static and does not change with decoder time step. We calculate the attention probability vector α^r over attributes using the attribute importance vector I^r. The attention weights can then be used to calculate the

record representation B^r for record r by using equations 4 and 5.

Record Encoder: A GRU based RNN encoder takes as input a sequence of attribute attended records $B^{1:N}$ and returns a sequence of hidden states $h_{1:N}$, where h_r is the encoded vector for record B^r. We obtain the final record encoding c_r (equation 6) by concatenating the GRU hidden states with the embedded record encodings B^r.

Static Record attention: In a table, a subset of record types can always be more salient compared to other record types. This is captured by learning a static set of weights over all the records. These weights regulate the dynamic attention weights computed during decoding at each time step. Equation 7 performs this step where g_r is the static record attention weight for r^{th} record and q and P are weights to be learnt. We do not have any constraints on static attention vector.

Dynamic Record attention for Decoder: Our decoder is a GRU based decoder with dynamic attention mechanism similar to (Mei et al., 2016) with modifications to modulate attention weights at each time step using static record attentions. At each time step t attention weights are calculated using 8, 9, 10, where γ_t^r is the aggregated attention weight of record r at time step t. We use the *soft attention* over input encoder sequences c_r to calculate the weighted average, which is passed to the GRU. GRU hidden state s_t is used to calculate output probabilities p_t by using a softmax as described by equation 11, 12, 13, which is then used to get output word y_t.

Due to the static attention at attribute level, the time complexity of a single pass is $O(TM + TT')$, where T is the number of records, M is the number of attributes and T' is the number of decoder steps. In case of dynamic attention at both levels (as in Nallapati et al. (2016)), the time complexity is much higher $O(TMT')$. Thus, *mixed hierarchical attention* model is faster than fully dynamic hierarchical attention. For better understanding of the contribution of hierarchical attention(MHAM), we propose a simpler non-hierarchical (NHM) architecture with attention only at record level. In NHM, B^r is calculated by concatenating all the record attributes along with corresponding record type.

				Input Table				**Generated Output**

TYPE	TIME	MIN	MAX	MEAN	MODE	MB100-4	MB20-2
temperature	17-30	27	36	29	-	-	-
windChill	17-30	14	26	18	-	-	-
windSpeed	17-30	14	20	16	-	-	10-20
windDir	17-30	-	-	-	SSW	-	-
gust	17-30	0	0	0	-	-	-
skyCover	17-30	-	-	-	-	75-100	-
skyCover	17-21	-	-	-	-	75-100	-
skyCover	17-26	-	-	-	-	75-100	-
skyCover	21-30	-	-	-	-	75-100	-
skyCover	26-30	-	-	-	-	75-100	-
precipPotential	17-30	26	58	43	-	-	-
rainChance	17-21	-	-	-	Lkly	-	-
snowChance	17-30	-	-	-	Chc	-	-
snowChance	17-26	-	-	-	Lkly	-	-

Reference: Periods of rain and possibly a thunderstorm . Some of the storms could produce heavy rain . Temperature rising to near 51 by 10am , then falling to around 44 during the remainder of the day . Breezy , with a north northwest wind between 10 and 20 mph . Chance of precipitation is 90 % . New rainfall amounts between one and two inches possible .

NHM: Periods of rain and possibly a thunderstorm . Some of the storms could produce heavy rain . Temperature rising to near 51 by 8am , then falling to around 6 during the remainder of the day . Breezy , with a north northwest wind 10 to 15 mph increasing to between 20 and 25 mph . Chance of precipitation is 90 % . New rainfall amounts between one and two inches possible .

MHAM: Periods of rain and possibly a *thunderstorm* . Some of the storms could produce *heavy rain* . Temperature rising to *near 51 by 8am* , then falling to *around 44* during the remainder of the day . Breezy , with a *north northwest wind* between *10 and 20 mph* . Chance of precipitation is *90 %* . New rainfall amounts between *one and two inches* possible .

TYPE	TIME	MIN	MAX	MEAN	MODE	MB100-4	MB20-2
temperature	17-30	27	36	29	-	-	-
windChill	17-30	14	26	18	-	-	-
windSpeed	17-30	14	20	16	-	-	10-20
windDir	17-30	-	-	-	SSW	-	-
gust	17-30	0	0	0	-	-	-
skyCover	17-30	-	-	-	-	75-100	-
skyCover	17-21	-	-	-	-	75-100	-
skyCover	17-26	-	-	-	-	75-100	-
skyCover	21-30	-	-	-	-	75-100	-
skyCover	26-30	-	-	-	-	75-100	-
precipPotential	17-30	26	58	43	-	-	-
rainChance	17-21	-	-	-	Lkly	-	-
snowChance	17-30	-	-	-	Chc	-	-
snowChance	17-26	-	-	-	Lkly	-	-

Reference: A chance of rain and snow . Snow level 5500 feet . Mostly cloudy , with a low around 31 . Calm wind becoming north northeast around 6 mph . Chance of precipitation is 40% .

NHM: A chance of rain and snow . Mostly cloudy , with a low around 31 . North northwest wind at 6 mph becoming east southeast . Chance of precipitation is 40% .

MHAM: A chance of *rain and snow* . Snow level *5800 feet lowering to 5300 feet* after midnight . Mostly cloudy , with a low *around 31* . *North northwest wind* at *6 mph* becoming south southwest . Chance of precipitation is *40%* .

TYPE	TIME	MIN	MAX	MEAN	MODE	MB100-4	MB20-2
temperature	17-30	27	36	29	-	-	-
windChill	17-30	14	26	18	-	-	-
windSpeed	17-30	14	20	16	-	-	10-20
windDir	17-30	-	-	-	SSW	-	-
gust	17-30	0	0	0	-	-	-
skyCover	17-30	-	-	-	-	75-100	-
skyCover	17-21	-	-	-	-	75-100	-
skyCover	17-26	-	-	-	-	75-100	-
skyCover	21-30	-	-	-	-	75-100	-
skyCover	26-30	-	-	-	-	75-100	-
precipPotential	17-30	26	58	43	-	-	-
rainChance	17-21	-	-	-	Lkly	-	-
snowChance	17-30	-	-	-	Chc	-	-
snowChance	17-26	-	-	-	Lkly	-	-

Reference: Rain and snow likely , becoming all snow after 8pm . Cloudy , with a low around 22 . South southwest wind around 15 mph . Chance of precipitation is 60% . New snow accumulation of less than one inch possible .

NHM : Rain or freezing rain likely before 8pm , then snow after 11pm , snow showers and sleet likely before 8pm , then a chance of rain or freezing rain after 3am . Mostly cloudy , with a low around 27 . South southeast wind between 15 and 17 mph . Chance of precipitation is 80% . Little or no ice accumulation expected . Little or no snow accumulation expected .

MHAM: Snow , and freezing rain , *snow after 9pm* . Cloudy , with a steady temperature *around 23* . Breezy , with a *south wind* between *15 and 20 mph* . Chance of precipitation is *60%* . New snow accumulation of *around an inch* possible .

Table 1: Anecdotal example. Records which contain all null attributes are not shown in the example. MB100-4 and MB20-2 correspond to mode-bucket-0-100-4 & mode-bucket-0-20-2 resp. in the dataset.

3 Experiments

Dataset and methodology: To evaluate our model we have used WEATHERGOV dataset (Liang et al., 2009) which is the standard benchmark dataset to evaluate tabular data summarization techniques. We compared the performance of our model against the state-of-the-art work of MBW (Mei et al., 2016), as well as two other baseline models KL (Konstas and Lapata, 2013) and ALK (Angeli et al., 2010). Dataset consists of a total of 29,528 tables (25000:1000:3528 ratio for train:validation:test splits) corresponding to scenarios created by collecting weather forecasts for 3,753 cities in the U.S.A over three days. There are 12 record types consisting of both numeric and categorical values. Each table contains 36 weather records (e.g., temperature, wind direction etc.) along with a corresponding natural language summary.

Input Encodings: Attributes were encoded based on the attribute type. Numbers are encoded in binary representation. Record type is encoded as a one-hot vector. Mode attribute is encoded using specific ordinal encodings for example 'Lkly',

'SChc', 'Chc' are encoded as '00100000000000', '00010000000000' and '00001000000000' respectively. Similar works for directions, for example 'NW', 'NNE' and 'NE' are encoded as '00000100000000', '00000011000000' and '00000001000000' resp. Time interval were also encoded as ordinal encodings, for example '6-21' is encoded as '111100' and '6-13' is encoded as '110000', the six bits corresponding to six atomic time intervals available in the dataset. Other attributes and words were encoded as one-hot vectors.

3.1 Training and hyperparameter tuning

We used TensorFlow (Abadi et al., 2015) for our experiments. Encoder embeddings were initialized by generating the values from a uniform distribution in the range [-1, 1]. Other variables were initialized using Glorot uniform initialization (Glorot and Bengio, 2010). We tune each hyperparameter by choosing parameter from a ranges of values, and selected the model with best sBLEU score in validation set over 500 epochs. We did not use any regularization while training the model. For both the models, the hyperparameter tuning

was separately performed to give both models a fair chance of performance. For both the models, Adam optimizer (Kingma and Ba, 2014) was used with learning rate set to 0.0001. We found embedding size of 100, GRU size of 400, static record attention size P of 150 to work best for MHAM model. We also experimented using bi-directional GRU in the encoder but there was no significant boost observed in the BLEU scores.

Evaluation metrics: To evaluate our models we employed BLEU and Rouge-L scores. In addition to the standard BLEU (sBleu) (Papineni et al., 2002), a customized BLEU (cBleu) (Mei et al., 2016) has also been reported. cBleu does not penalize numbers which differ by at most five; hence 20 and 18 will be considered same.

4 Results and Analyses

Table 2 describes the results of our proposed models (MHAM and NHM) along with the aforementioned baseline models. We observe a significant performance improvement of 16.6 cBleu score (24%) and 18.3 sBleu score (30%) compared to the current state-of-the-art model of MBW. MHAM also shows an improvement over NHM in all metrics demonstrating the importance of hierarchical attention.

Model	sBleu	cBleu	Rouge-L
KL	36.54	-	-
ALK	38.40	51.50	-
MBW	61.0	70.4	-
NHM	76.2	85.0	86.4
MHAM	**79.3**	**87.0**	**88.5**

Table 2: Overall results

Attention analysis: Analysis of figure 3 reveals that the learnt attention weights are reasonable. For example, as shown in figure 3(a), for the phrase 'with a high near 52', the model had a high attention on *temperature* before and while generating the number '52'. Similarly while generating 'mostly cloudy', the model had a high attention on *precipitation potential*. Attribute attentions are also learned as expected (in figure 3(b)). The *temperature*, *wind speed* and *gust* records have high weights on min/max/mean values which describe these records.

Qualitative analysis: Table 1 contains example table-summary pairs, with summary generated by the proposed hierarchical and non-hierarchical
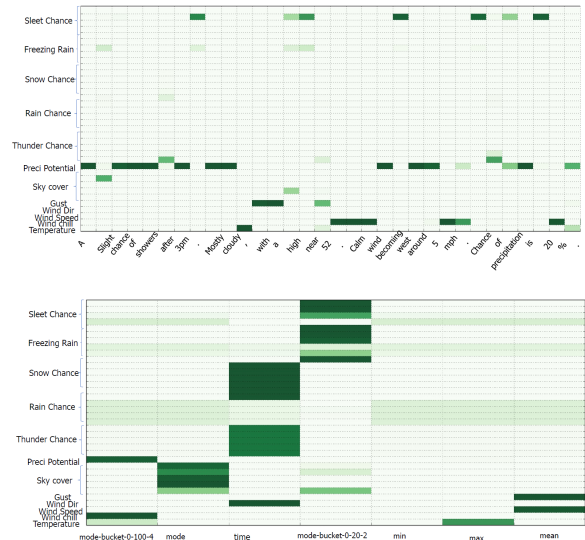

Figure 3: Heatmaps: (a) record level attention (top) and (b) attribute level attention (bottom).

versions. We observe that our model is able to generate numbers more accurately by enabling hierarchical attention. Our model is also able to capture weak signals like *snow accumulation*. Further, our proposed model MHAM is able to avoid repetition as compared to NHM.

5 Conclusion and Future Work

In this work, we have formulated the problem of *standard table summarization* where all the tables come from a predefined schema. Towards this, we proposed a novel mixed hierarchical attention based encoder-decoder approach. Our experiments on the publicly available WEATHERGOV benchmark dataset have shown significant improvements over the current state-of-the-art work. Moreover, this proposed method is theoretically more efficient compared to the current fully dynamic hierarchical attention model. As future work, we propose to tackle general tabular summarization where the schema can vary across tables in the whole dataset.

6 Acknowledgements

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin,

Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Ion Androutsopoulos, Gerasimos Lampouras, and Dimitrios Galanis. 2014. Generating natural language descriptions from OWL ontologies: the naturalowl system. *CoRR*, abs/1405.6164.

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512. Association for Computational Linguistics.

KyungHyun Cho, Aaron C. Courville, and Yoshua Bengio. 2015. Describing multimedia content using attention-based encoder-decoder networks. *CoRR*, abs/1507.01053.

Emilie Colin, Claire Gardent, Yassine Mrabet, Shashi Narayan, and Laura Perez-Beltrachini. 2016. The webnlg challenge: Generating text from dbpedia data. In *INLG 2016 - Proceedings of the Ninth International Natural Language Generation Conference, September 5-8, 2016, Edinburgh, UK*, pages 163–167.

Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M. Rush. 2017. Image-to-markup generation with coarse-to-fine attention. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 980–989.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS10). Society for Artificial Intelligence and Statistics*.

Srinivasan Iyer, Ioannis Konstas, Alvin Cheung, and Luke Zettlemoyer. 2016. Summarizing source code using a neural attention model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2073–2083, Berlin, Germany. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ioannis Konstas and Mirella Lapata. 2013. Inducing document plans for concept-to-text generation. In *EMNLP*.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. Association for Computational Linguistics.

Percy Liang, Michael I. Jordan, and Dan Klein. 2009. 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*, ACL '09, pages 91–99, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jeffrey Ling and Alexander M. Rush. 2017. Coarse-to-fine attention models for document summarization. In *Proceedings of the Workshop on New Frontiers in Summarization, NFiS@EMNLP 2017, Copenhagen, Denmark, September 7, 2017*, pages 33–42.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2016. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 720–730, San Diego, California. Association for Computational Linguistics.

Ramesh Nallapati, Bing Xiang, and Bowen Zhou. 2016. Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2253–2263.

627