Text-to-Text Pre-Training for Data-to-Text Tasks

Mihir Kale
Google Research
mihirkale@google.com

Abhinav Rastogi
Google Research
abhirast@google.com

Abstract

We study the pre-train + fine-tune strategy for data-to-text tasks. Our experiments indicate that text-to-text pre-training in the form of T5 (Raffel et al., 2019), enables simple, end-to-end transformer based models to outperform pipelined neural architectures tailored for data-to-text generation, as well as alternative language model based pre-training techniques such as BERT and GPT-2. Importantly, T5 pre-training leads to better generalization, as evidenced by large improvements on out-of-domain test sets. We hope our work serves as a useful baseline for future research, as transfer learning becomes ever more prevalent for data-to-text tasks.

1 Introduction

Natural language generation from structured data, or data-to-text (Kukich, 1983; McKeown, 1985), is the task of generating natural language text conditioned on source content provided in the form of structured data such as a table, graph etc. Some example applications include task oriented dialog (Wen et al., 2015), summarizing weather forecasts (Sripada et al.; Goldberg et al., 1994), etc.

In this work we study the applicability of large scale text-to-text transfer learning learning for this task. In particular, we focus on pre-training in the form of the "Text-to-Text Transfer Transformer" (T5) models released by Raffel et al. (2019). Fine-tuning T5 achieves state-of-the-art results on diverse benchmarks spanning task oriented dialogue (MultiWoz), tables-to-text (ToTTo) and graph-to-text (WebNLG). Empirical results further demonstrate the following:

- Pre-training greatly improves robustness of models to out-of-domain inputs.
- By leveraging pre-training, a simple end-to-end transformer model can outperform sophisticated, multi-stage pipelined approaches and other exotic architectures like graph neural networks.
- T5 outperforms alternatives like BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2019).

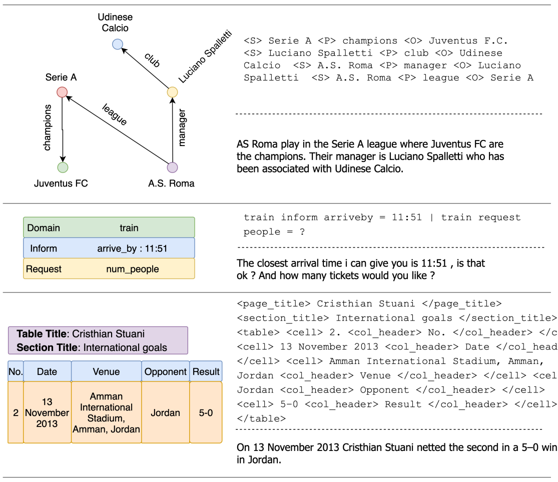
Our approach is simple, only scratching the surface of what is possible. There is much to be explored in the space of leveraging unlabelled data, developing unsupervised objectives etc. that are more tailored for generating text from structured data. We hope our work serves as a useful baseline for future research, as pre-training becomes ever more prevalent for this task.

2 Related Work

Data-to-Text Early research on data-to-text focused on rule-based methods (Reiter and Dale, 2000), while recent works have favored neural approaches (Wen et al., 2015). Liu et al. (2018) generate text by conditioning language models on tables, Puduppully et al. (2019) explicitly model entities and Marcheggiani and Perez-Beltrachini (2018) encode structured data using graph convolutional networks. Ferreira et al. (2019) and Moryossef et al. (2019) find that neural pipelined approaches perform better than end-to-end models.

Transfer Learning Devlin et al. (2018) showed that unsupervised pre-training can greatly benefit tasks like, question answering, summarization etc. In particular, Raffel et al. (2019) perform a large scale study of different training objectives, model capacity and size of data. Peng et al. (2020) and Chen et al. (2019b) show that pre-training in the form of GPT-2 can indeed improve performance on the data-to-text task as well.

Proceedings of The 13th International Conference on Natural Language Generation, pages 97–102,
Dublin, Ireland, 15-18 December, 2020. ©2020 Association for Computational Linguistics



```
<S> Serie A <P> champions <O> Juventus F.C.
<S> Luciano Spalletti <P> club <O> Udinese
Calcio  <S> A.S. Roma <P> manager <O> Luciano
Spalletti  <S> A.S. Roma <P> league <O> Serie A
```

AS Roma play in the Serie A league where Juventus FC are the champions. Their manager is Luciano Spalletti who has been associated with Udinese Calcio.

Domain	train
Inform	arrive_by : 11:51
Request	num_people

```
train inform arriveby = 11:51 | train request
people = ?
```

The closest arrival time i can give you is 11:51 , is that ok ? And how many tickets would you like ?

Table Title: Cristhian Stuani
Section Title: International goals

No.	Date	Venue	Opponent	Result
2	13 November 2013	Amman International Stadium, Amman, Jordan	Jordan	5-0

```
<page_title> Cristhian Stuani </page_title>
<section_title> International goals </section_title>
<table> <cell> 2. <col_header> No. </col_header> </cell>
<cell> 13 November 2013 <col_header> Date </col_header>
</cell> <cell> Amman International Stadium, Amman,
Jordan <col_header> Venue </col_header> </cell> <cell>
Jordan <col_header> Opponent </col_header> </cell>
<cell> 5-0 <col_header> Result </col_header> </cell>
</table>
```

On 13 November 2013 Cristhian Stuani netted the second in a 5–0 win in Jordan.

Figure 1: Examples from each dataset - The first row is WebNLG, second is Multiwoz and third is ToTTo. Each row illustrates the structured data (left), its linearized representation (top) and the target text(bottom)

3 Pre-training

We rely on the T5 pre-trained models released by Raffel et al. (2019). They consist of a transformer based encoder-decoder architecture. These models were pre-trained in a multitask fashion with an unsupervised "span masking" objective on Common Crawl data as well as supervised translation, summarization, classification, and question answering tasks. Note that none of the supervised tasks include language generation from structured data. [1]

To study the impact of model capacity, we experiment with different T5 variants - Small (60 million parameters), Base (220 million), Large (770 million) and 3B (3 billion).

4 Fine-tuning

Our modeling approach is simple. The data-to-text task is cast in the text-to-text framework by representing the structured data as a flat string (linearization). Figure 1 shows examples of the input representation for each dataset. We then fine-tune T5 on the data-to-text corpus for a small number of steps.

Following (Raffel et al., 2019), models are fine-tuned with a constant learning rate of 0.001. We use a batch size of 131,072 tokens, and a maximum input length of 512 tokens. The maximum training steps is set to 5K for WebNLG, while the larger ToTTo dataset is trained for 10K steps. The T5 vocabulary consists of 32,000 sentencepieces. All the model parameters are updated in the fine-tuning process.

The best checkpoint is chosen based on the BLEU (Papineni et al., 2002) score on the development set. Decoding is done via greedy search. In the final evaluation, for each dataset we rely on metrics used by prior work.

5 Datasets

We conduct experiments on 3 English datasets spanning a variety of domains.

- **ToTTo** (Parikh et al., 2020) consists of Wikipedia tables paired with natural language descriptions. The input is a set of cells from a table, along with metadata such as the title of the table.
- **MultiWoz** (Budzianowski et al., 2018) is a corpus of 10K human-human dialogs for developing task oriented dialogue systems. For the NLG task, a meaning representation encapsulating system actions must be verbalized into natural language response.
- **WebNLG** (Gardent et al., 2017), where the task is to convert a graph of subject-object-predicate triples into a textual description.

Each dataset uses a different kind of structured data (tables, meaning representations and graph/triples). Table 1 lists the sizes of the three datasets and Figure 1 shows examples for each.

Dataset	Train	Dev	Test
WebNLG	18.1K	2.2k	4.9k
ToTTo	120K	7.7k	7.7k
Multiwoz	56.8K	7.3k	7.3k

Table 1: Dataset sizes.

6 Results and Discussion

6.1 WebNLG

The evaluation is done using BLEU and METEOR (Lavie and Agarwal, 2007), similar to (Ferreira et al., 2019). The test set is split into two parts - seen and unseen. The examples in the unseen set are drawn from domains not present in the training set, along with roughly 100 new predicates.
Some of the baselines we compare with are:

[1] Initial experiments with T5 variants trained on a purely unsupervised objective did not show any difference in performance.

Model	BLEU			METEOR		
	O	S	U	O	S	U
Melbourne†	45.1	54.5	33.3	0.37	0.41	0.33
GTR-LSTM†	37.1	54.0	29.2	0.31	0.37	0.28
Pipe-Trans	51.7	56.4	38.9	0.32	0.41	0.21
Step†	47.4	53.3	34.4	0.39	0.44	0.34
DualEnc	51.4	63.4	36.7	0.41	0.45	0.37
T5-Small	52.0	62.6	38.8	0.41	0.45	0.37
T5-Base	55.2	**64.7**	49.4	0.43	**0.46**	0.41
T5-Large	**57.1**	63.9	**52.8**	**0.44**	**0.46**	**0.41**
T5-3B	54.0	62.8	52.0	0.43	0.45	0.42

Table 2: Results on WebNLG. O stands for Overall test set, S for Seen and U for Unseen. Pipe-Trans is Pipeline-Transformer.

- **Melbourne**, a neural encoder-decoder approach, which scored the highest in the automatic evaluation of the WebNLG challenge (Gardent et al., 2017). The model relies on delexicalization, where entities are replaced with placeholders.

- **GTR-LSTM** (Distiawan et al., 2018), which employs a graph based triple encoder.

- **Step-by-Step** (Moryossef et al., 2019) which splits the generation procedure into a planning stage followed by a neural generation stage.

- **Pipeline-Transformer** (Ferreira et al., 2019), a pipelined neural system consisting of discourse ordering, text structuring, lexicalization and referring expression generation.

- **DualEnc** (Zhao et al., 2020), the current state-of-the-art system. It consists of a graph convolution network based planning model which first predicts the order of the triples, followed by a separate LSTM with attention and copy mechanism model to generate the text. To train the planning model, the approach relies on extra annotations for the triple ordering. Such annotations are can be expensive and time consuming to obtain, especially for large, complex inputs.

Results are reported in Table 2, for the overall test set as well as the Seen and Unseen splits. T5-Large performs the best across BLEU as well as METEOR. It improves over DualEnc by 4.3 BLEU on the overall test set. It also displays excellent generalization to new domains and relations, with a 14 BLEU improvement on the unseen test set. The results indicate that with pre-training, end-to-end neural models can surpass sophisticated pipelined approaches while being much more robust to domain shift.

Model	Overall		Non-Overlap	
	BLEU	PAR	BLEU	PAR
PGen	41.6	51.6	32.2	45.2
BERT-to-BERT	44.0	52.6	34.8	46.7
T5-3B	**49.5**	**58.4**	**41.4**	**54.2**

Table 3: Results on the ToTTo test set. PAR is short for PARENT. PGen stands for Pointer Generetator (See et al., 2017a).

Model	Overall		Non-Overlap	
	BLEU	PAR	BLEU	PAR
BERT-to-BERT	44.0	52.6	34.8	46.7
T5-Small	45.7	55.9	37.7	51.6
T5-Base	47.7	57.1	39.6	52.6
T5-Large	48.1	57.3	39.8	52.8
T5-3B	48.4	57.8	40.4	53.3

Table 4: Results on the ToTTo development set for different variants of T5.

6.2 ToTTo

Following (Parikh et al., 2020), BLEU and PARENT are employed as evaluation metrics for this table-to-text generation task. PARENT is a reference less, word-overlap based metric that reflects the factual accuracy of generated text relative to the structured data. Dhingra et al. (2019) find that PARENT correlates better with human factual accuracy judgements in comparison to other generation metrics like ROGUE (Lin, 2004) and METEOR. The following baseline models are compared:

- **Pointer Generator** (See et al., 2017b) - An LSTM based seq2seq model with attention and pointer network based copy mechanism.

- **BERT-to-BERT** (Rothe et al., 2019) - A transformer based encoder-decoder model, where both the encoder and decoder are initialized with BERT.

Since it deals with open domain tables, ToTTo is arguably the most challenging dataset. Notably, it features a hidden test set, which is split into two halves - Overlap and Non-Overlap. The Non-Overlap test set features examples that are out-of-domain from the training set.

Results are reported in Table 3. T5-3B[2] achieves state-of-the-art results [3], improving upon the BERT baseline by 5.5 BLEU and 5.8 PARENT. Moreover, the model is more robust to out-of-domain tables, with larger improvements of 6.6 BLEU and 7.5 PARENT on the Non-Overlap test set. Table 4 reports results on the development set for the different T5 model sizes. T5-Small outperforms BERT-to-BERT, even though it has 3x fewer parameters (220M vs 60M).

Model	BLEU	SER
HDSA†	26.5	12.14
SC-GPT2	30.8	**0.53**
T5-Small	34.6	1.27
T5-Base	**35.1**	0.99
T5-Large	34.7	0.92
T5-3B	34.8	0.86

Table 5: Results on Multiwoz. †(Chen et al., 2019a)

6.3 MultiWoz

Evaluation on MultiWoz is done using BLEU and SER (Slot Error Rate). SER is the fraction of examples where at least one slot value from the structured data is not expressed in the predicted response. [4]

Our baselines are

- **HDSA** (Chen et al., 2019a) is a transformer based architecture that encodes the dialog acts into a multi-layer hierarchical graph, with individual attention heads modeling specific nodes in graph.

- **SC-GPT2** (Peng et al., 2020) is a GPT-2 (345M parameters) model that is further pretrained on a large data-to-text dialog corpus consisting of 400,000 examples and finally fine-tuned on MultiWoz. This 2 stage pretraining approach is currently state-of-the-art for Multiwoz.

Results are reported in Table 5. All T5 based models (including T5-small which has 5x fewer parameters) outperform SC-GPT2 by 4-5 BLEU without any in-domain pre-training. We note that the SER score on MultiWOZ is slightly worse in comparison with SC-GPT. SC-GPT generates 5 predictions for each input and then ranks them based on the SER score itself, which naturally leads to better slot error rates. On the other hand, we generate a single output.

	Seen		Unseen	
	Nat	Acc	Nat	Acc
DualEnc	2.30	89.2	1.99	66
T5-Large	**2.39**	**92.0**	**2.33**	**90.0**

Table 6: Human evaluation on WebNLG. Nat is short for Naturalness and Acc is short for Accuracy.

6.4 Human Evaluation

We conduct a human evaluation study on WebNLG. Human raters are presented with predicted text, along with up to 3 ground truth references. They are asked to judge the prediction along two axes - (1) Accuracy - A binary rating to gauge whether the prediction conveys the same information as the gold references and (2) Naturalness - A five point scale between 1-3, with 3 indicating a perfectly fluent and grammatical response. Each prediction is rated by 3 raters. For accuracy, we take the majority vote and for naturalness we take the average. We evaluate 500 examples, equally split between the Seen and Unseen test sets.

The evaluation is performed for T5-Large and the current state-of-the-art DualEnc model. Results are reported in Table 6. On the Seen set, both models perform well, with T5 being rated better across both metrics. On the Unseen set, DualEnc shows a large drop of 24% in accuracy while the fluency degrades to just 1.99. Remarkably, T5 sees only a marginal drop, scoring 90% on accuracy and 2.33 on fluency. Table 7 shows some qualitative examples.

6.5 Impact of model capacity

Our experiments with different T5 variants of varying sizes shed some light on how model capacity impacts performance. The results suggest that it largely depends on the size and complexity of the dataset. For instance, MultiWoz exhibits the least variation in the structured data and is fairly large at 56k examples. Here, even the smallest model T5-Small, is on par with the larger models. WebNLG has only 18K examples and features roughly 200

[2] We used beam search with a width of 10 for the test set submission.

[3] The leaderboard can be found at https://github.com/google-research-datasets/totto.

[4] The metric is noisy since the comparison is done via exact match, does not accoutn for paraphrases and does not cover all slots.

Input DualEnc T5	<aidastella, christening date, 2013-03-16> Aidastella was inaugurated on March 16 , 2013 . Aidastella was christened on March 16 , 2013 .
Input DualEnc T5	<Andra (singer). genre , rhythm and blues> Andra singer is rhythm and blues . Andra is a singer who plays rhythm and blues .
Input DualEnc T5	<Aaron deer, genre, indie rock><Aaron Deer, origin, Indiana><Aaron Deer, origin, United States> Aaron Deer , indie rock , has a origin of Indiana and is located in United States . Aaron Deer is an American from Indiana who is part of the genre of indie rock .
Input DualEnc T5	<Alvah Sabin, birth date, 1793-10-23><Alvah Sabin, office (worked at , worked as), secretary of state of Vermont> Alvah Sabin was born on October 23 , 1793 and is in secretary of state of Vermont . Alvah Sabin was born on 23 October 1793 and served as secretary of state of Vermont .

Table 7: Model predictions on the WebNLG Unseen set. DualEnc struggles to verbalize predicates and produces ungrammatical output. T5 output is accurate and more grammatical.

distinct relations. On the seen test set, all models perform comparably. However, on the unseen test set we notice that performance increases with model size. In particular, there is a stark jump of 10 BLEU when going from T5-Small to T5-Base, implying that model capacity is critical for out-of-domain generalization. A similar trend is observed for ToTTo (Table 4), with a noticeable improvement from Small to Base, followed by smaller improvements upto T5-3B.

7 Conclusion

In this study we evaluated pre-training in the form of T5 for the data-to-text task. We found that it leads to state-of-the-art results, while greatly improving robustness to out-of-domain inputs. In the future, we hope to design unsupervised pre-training objectives that are specifically tailored for the data-to-text task. We also hope to extend this work to multiple languages, especially low resource ones.

References

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026.

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019a. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3696–3709.

Zhiyu Chen, Harini Eavani, Yinyin Liu, and William Yang Wang. 2019b. Few-shot nlg with pre-trained language model. *arXiv preprint arXiv:1904.09521*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4884–4895.

Bayu Distiawan, Jianzhong Qi, Rui Zhang, and Wei Wang. 2018. Gtr-lstm: A triple encoder for sentence generation from rdf data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1627–1637.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 552–562.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The webnlg challenge: Generating text from rdf data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133.

Eli Goldberg, Norbert Driedger, and Richard I Kittredge. 1994. Using natural-language processing to produce weather forecasts. *IEEE Expert*, 9(2):45–53.

Karen Kukich. 1983. Design of a knowledge-based report generator. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics*, pages 145–150. Association for Computational Linguistics.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine*

Translation, pages 228–231. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2018. Table-to-text generation by structure-aware seq2seq learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Diego Marcheggiani and Laura Perez-Beltrachini. 2018. Deep graph convolutional encoders for structured data to text generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 1–9.

Kathleen R McKeown. 1985. Text generation: using discourse strategies and focus constraints to generate natural language text.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2267–2277.

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.

Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. *arXiv preprint arXiv:2002.12328*.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6908–6915.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Ehud Reiter and Robert Dale. 2000. *Building natural language generation systems*. Cambridge university press.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2019. Leveraging pre-trained checkpoints for sequence generation tasks. *arXiv preprint arXiv:1907.12461*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017a. 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)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Abigail See, Peter J Liu, and Christopher D Manning. 2017b. 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)*, pages 1073–1083.

Somayajulu Sripada, Ehud Reiter, and Ian Davy. Sumtime-mousam: Configurable marine weather forecast generator.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Chao Zhao, Marilyn Walker, and Snigdha Chaturvedi. 2020. Bridging the structural gap between encoding and decoding for data-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.