Exploring Structural Encoding for Data-to-Text Generation

Joy Mahapatra
Indian Statistical Institute, Kolkata
joymahapatra90@gmail.com

Utpal Garain
Indian Statistical Institute, Kolkata
utpal@isical.ac.in

Abstract

Due to efficient end-to-end training and fluency in generated texts, several encoder-decoder framework-based models are recently proposed for data-to-text generations. Appropriate encoding of input data is a crucial part of such encoder-decoder models. However, only a few research works have concentrated on proper encoding methods. This paper presents a novel encoder-decoder based data-to-text generation model where the proposed encoder carefully encodes input data according to underlying structure of the data. The effectiveness of the proposed encoder is evaluated both extrinsically and intrinsically by shuffling input data without changing meaning of that data. For selecting appropriate content information in encoded data from encoder, the proposed model incorporates *attention gates* in the decoder. With extensive experiments on WikiBio and E2E dataset, we show that our model outperforms the state-of-the models and several standard baseline systems. Analysis of the model through component ablation tests and human evaluation endorse the proposed model as a well-grounded system.

1 Introduction

Data-to-text generation (Gatt and Krahmer, 2018; Reiter and Dale, 2000) aims to produce human-understandable text from semi-structured data such as tables, concepts, etc. The input data consists of multiple records (or events), where each record represents a particular field (or attribute) of the data. Table 1 shows an example for data-to-text generations, where text y is used to describe restaurant data x . The example consists of three records, and each record describes a field with a name (underlined part) and corresponding values (*italic* part).

data (x)

record 1	name: *The Punter*
record 2	food: *English*
record 3	price range: *high*

text (y)

The Punter is a restaurant with high prices.

Table 1: An example of data-to-text generation.

Due to efficient end-to-end training and fluency in generated texts (Novikova et al., 2017b; Sutskever et al., 2014), numerous data-to-text generation systems have adopted encoder-decoder based model (Gong et al., 2019; Liu et al., 2018). In such encoder-decoder based models, a proper meaningful encoding of input data is a real concern. As in most of the data-to-text generation, input data poses record-field structure (like in table 1), it is natural to realise the need for appropriate *structural* encoding for record-field structure. Most of the existing encoder-decoder models for data-to-text generation (Liu et al., 2018; Nema et al., 2018) primarily focus more on attention mechanisms than structural encoding. However, a recent interesting study by Gong et al. (2019) shows some effective encoding strategies based on functional dimensions of data.

Selecting appropriate content from input data (also known as content selection) is an important task. For example, according to table 1, the 'name' and 'price range' fields of the data is considered as contents w.r.t. text y. Detecting such contents from encoded data is a hard task.

We propose an encoder-decoder model for data-to-text generation where the encoder encodes data based on record-field structures. We introduce structure-wise attention gates to capture appropriate content from the encoded data. As records in an input data don't pose any ordering among them

Proceedings of the 14th International Conference on Natural Language Generation (INLG), pages 404–415,
Aberdeen, Scotland, UK, 20-24 September 2021. ©2021 Association for Computational Linguistics

(for example, in table 1, there is no order among the three records in x), the efficiency of the proposed encoder is estimated through shuffling those records. The comparisons of our system with existing high-scoring systems on WikiBio and E2E dataset bring out the distinction of our model. Additional human evaluation signifies that the proposed model performs well in terms of both readability and adequacy of generated text.

2 Notations

We follow the notations which are commonly used in popular data-to-text generation models (Angeli et al., 2010; Nie et al., 2019; Dhingra et al., 2019). The goal of our task ($\mathcal{T}$) is to produce text representation $y = y_1y_2...y_w$ (where y_i is the i-th word in y) from a given an input data x.

$$\mathcal{T} \ : \ \hat{y} \leftarrow \underset{y}{\operatorname{argmax}}\, p(y|x)$$

An input (x) consists of multiple records $\{r_i\}_{i=1}^{n}$. A record (r_j) contains a field f^j, with its name n^j and the corresponding value $v^j = v_1^j v_2^j ... v_w^j$ (where, v_i^j is the i-th word in v^j).

3 Approach

We propose a neural network based encoder-decoder model for the task $\mathcal{T}$, where the encoder structurally encodes the input (x). The decoder is a recurrent neural network with two sub-modules—(i) attention gates for appropriate content selection and (ii) copy module to handle the appearances of rare words in generating text.

3.1 Structural Encoder

The notion of the proposed encoder comes from the underlying structure of input data. We consider each input data comprises of two structures—(i) fields as fine-grained structure; (ii) records as coarse-grained structure. For example, in figure 1, input data x contains two records ($\{r_i\}_{i=1}^{2}$) with each record consists of two field parts ($r_1 = (f_1^1, f_2^1)$ and $r_2 = (f_1^2, f_2^2)$).

The proposed encoder encodes input data based on this record-field structures (Figure 1) in bottom-up approach. Each structure (field and record) encoding involves two types of connections (the arrows in figure 1 show these connections)—

- The horizontal dotted arrows denote **horizontal connections**—the objective of these connections is to help in making relationships among close components.
- The dashed arrows denote **hierarchical connections**—the purpose of these connections is to accumulate information from all similar components (either records or fields) and forward that information to next stage.

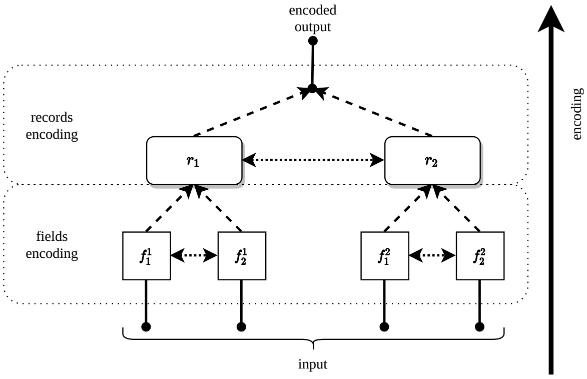

Figure 1: Structure of input data and bottom-up encoding.

So with the proposed structural encoder, knowledge about the record and field structures of input data get encoded.

3.1.1 Field Encoding

In field encoding of our structural encoder, all field words inside of a record are encoded together into a single vector representation, which eventually represents the record.

Embedding of each field's value (words) and field's name are obtained through learnable embedding matrix as follows.

$$Z_f[f_k^j] = [E_n[n^j]; E_v[v_k^j]]$$

where, $E_*[w]$ stands for embedding of w. $[;]$ denotes the concatenations. E_* are learnable parameters of size ($|v| \times d_{E_*}$), where $|v|$ is the size of vocabulary. Note that, here we use different embedding for both field's values and field' names. The $Z_f[f_k^j]$ denotes encoding for the k-th word in f^j field together with the field name. This Z_f is send to the horizontal connections of the field encoding.

Horizontal Connections in Field Encoding: Horizontal connections in field encoding are relating all fields words ($Z_f[f_*^j]$) inside of a record (r^j). Now, field words (f_*^j) can be either a sequence or bag-of-words (i.e. orderless/non-sequential). For example, in figure 1, the 'name' field contains two words *'The'* and *'Punter'* as a sequence. However, if the 'price range' field contains two words—*'high'* and *'medium'*, to denote a restaurant offers foods of both high and medium price range, then these two words behave as bag-of-words.

To appease both sequence and bag-of-words nature together we build horizontal connection of field encoding in a distinct way. For sequence data we use Bi-LSTM (Graves et al., 2013) networks; for orderless bag-of-words we skip (with help of skip-interconnections (He et al., 2016)) this Bi-LSTM network.

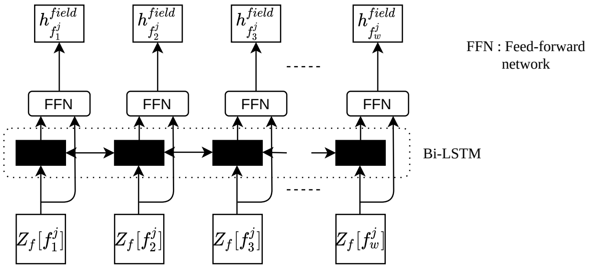

Figure 2: Horizontal connection in field encoding (for fields inside j-th records).

Eventually, a feed-forward network (equation 1) is used to merge skip-interconnections and Bi-LSTM. These skip-interconnections play an important role in our model while handling orderless/non-sequential data—we empirically show this in the experimental section 4.6. Figure 2 show such horizontal connections of field encoding.

The Bi-LSTM output for the field in j-th record is as follows,

$$h_{f^j} = \text{BiLSTM}(Z_f[f_1^j], ..., Z_f[f_w^j])$$

Finally, we make use of an affine transformation on $[h_{f_k^j}; f_k^j]$.

$$h^{field}_{f_k^j} = \tanh(\mathbf{W_{fa}}[h_{f_k^j}; f_k^j] + \mathbf{b_{fa}}) \qquad (1)$$

where, $\mathbf{W_{fa}}$ and $\mathbf{b_{fa}}$ are the learnable parameters. So, $h^{field}_{f^j}$ (and $h^{field}_{f_k^j}$ is k-th field) is output of horizontal connections of field encoding.

Hierarchical Connections in Field Encoding: The hierarchical connections in field encoding aim to accumulate all fields information ($h^{field}_{f_k^j}$) inside of a record (r_j) and gives a representation of the record from its fields point of view.

$$h_j^{mp} = \text{maxpool}(h^{field}_{f^j})$$

For hierarchical connections in field encoding, we use max-pooling and key-query based self-attention mechanism (Conneau et al., 2017). The max-pooling is used because of our intuition that max-pooling help in capturing the essence of a record from its fields ($h^{field}_{f_k^j}$). We draw this intuition from popular convolution neural networks (Krizhevsky et al., 2012).
We find that use of max-pooling is more effective than using the last state of underlying horizontal connection. Remember, max-poling considers all states of underlying horizontal connection—which is helpful for long sequences.

For the key-query based self-attention on field encoding, we choose field values ($h^{field}_{f_*^j}$) as keys and max-pooled record value (h_j^{mp}) as query. Based on our previous intuition behind h_j^{mp}, the query of the self attentions holds essence of record (r_j).

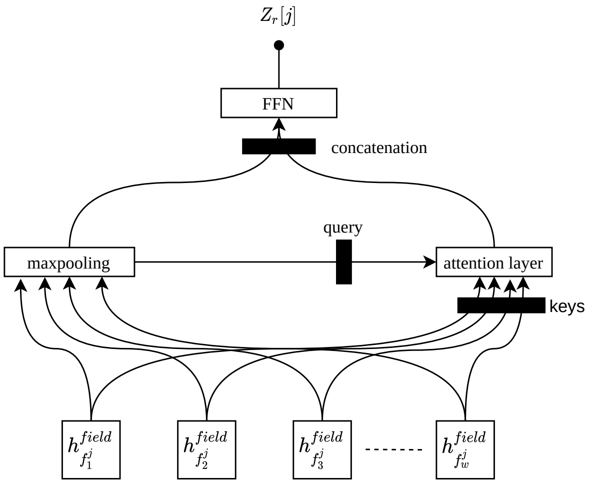

Figure 3: Hierarchical connections in field encoding (for fields inside j-th records).

For attention scoring, we use popular concatenative attention method (Luong et al., 2015).

$$score^f_{ji} = v_f^T \tanh(\mathbf{W_{fb}}[h_j^{mp}; h^{field}_{f_i^j}])$$
$$\alpha^f_{ji} = \text{softmax}(score^f_{ji})$$
$$c^f_j = \sum_{i=1}^{m} \alpha^f_{ji}.h^{field}_{f_i^j}$$

here, $score^f \in \mathcal{R}$ denotes attention score, α^f denotes the attention weight, and c^f_j is the attention/context vector for r_j record.

At the end of hierarchical connections in field encoding, we represent each record (r_j) through an affine transformation over concatenation of corresponding max-pooled value (h_j^{mp}) and context vector value (c_j^f).

$$Z_r[j] = \mathbf{W_{fc}}[h_j^{mp}; c_j^f] + \mathbf{b_{fc}}$$

Hence, $Z_r[j]$ is the output of field encoding for j-th records. Figure 3 shows the hierarchical connections of field encoding. It is worth to note here that as self-attention and max-pool operations don't rely on order of a data, hence there is no question of order-sensitiveness of hierarchical connections.

3.1.2 Record Encoding

The objective of record encoding is to find a vector representation for input data x in terms of its underlying record structures. The record encoding is quite similar to the field encoding. It contains both horizontal and hierarchical connections, which are same as field encoding. The horizontal connection output of record encoding is h_j^{record} for the j-th record. The final output of the record encoding is encoded representation $Z_d[x]$ of input data x.

3.2 Decoder

The decoder module consists of two key parts—structure-wise (i.e. record and field) attention gates and a copy module. Figure 4 shows a schematic view of our decoder.

3.2.1 State Update

The initial state of the decoder (s_0) is initialized through the output of record encoding, $Z_d[x]$. For updating the t-th step state (s_t) in the decoder, we use dual attention mechanisms (Liu et al., 2018; Gong et al., 2019) to merge attentions over record and field structures. We define attention over both record and field encoding on their horizontal connections outputs.

$$\beta_{tj}^r = \text{attention}(s_t, r_j)$$
$$\phi_t^r = \sum_{i=1}^{n} \beta_{ti}^r . h_{r_i}^{record}$$

where, β^r stands for attention weight (for j-th record, r_j, t-timestep of decoding) and ϕ_t^r is the attention/context vector of record encoding. Similarly, for field we get attention weight β^f.

Now, due to dual attention mechanism we modify effective attention weight of field encoding though $\gamma^f = \beta^f \times \beta^r$. Hence, the context vector (ϕ_t^f) of effective field attention is defined through γ^f.

Attention Gates: We introduce two attention gates for both field and record structures which help us to read context vectors ϕ_t^r and ϕ_t^f. We define these gates through current decoder state (s_t) and encoded data ($Z_d[x]$) as follows,

$$g_t^r = \sigma(\mathbf{W_{rg}} s_t + \mathbf{U_{rg}} Z_d[x] + \mathbf{b_{rg}})$$
$$g_t^f = \sigma(\mathbf{W_{fg}} s_t + \mathbf{U_{fg}} Z_d[x] + \mathbf{b_{fg}})$$

where, g_t^f and g_t^r are the attention gates for field and record context. Those two gates perform crucial function in our model as they handle content selection from the context vectors (which is nothing but encoded input) to decoder. The values of these gates change time to time to decide whether to inhibit (by value '0') and exhibit (by value '1') the content of context vectors. The attention context information is defined as below.

$$\hat{\phi}_t^r = g_t^r \odot \phi_t^r$$
$$\hat{\phi}_t^f = g_t^f \odot \phi_t^f$$

Finally, we update the decoder state with $\hat{c}_t^{record}$ and $\hat{c}_t^{field}$ as given below.

$$\tilde{s}_t = \tanh(\mathbf{W_d}[s_t; \hat{\phi}_t^{record}; \hat{\phi}_t^{field}])$$

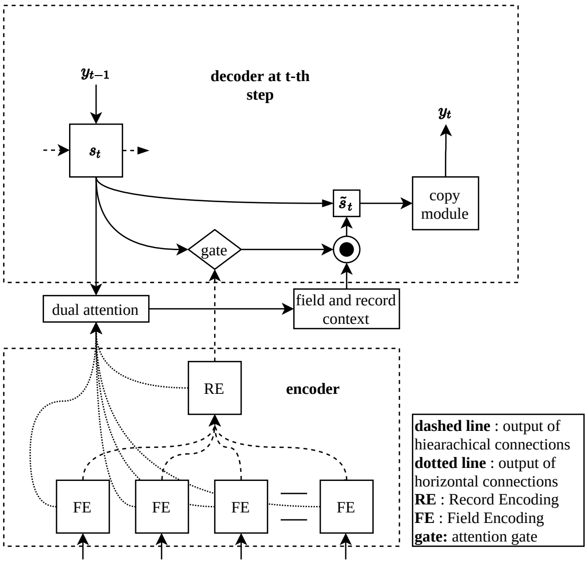

Figure 4: Proposed encoder-decoder model.

3.3 Copy Module

To handle rare words, we follows Gulcehre et al. (2016)'s conditional-copy techniques for our model, where the binary copy-indicator variable

(cp_t) in conditional copy-module defined as follows,

$$p(cp_t = 0|\tilde{s}_t, Z_d[x]) = \sigma(\mathbf{W_{cp}}[\tilde{s}_t; Z_d[x]])$$

We use input encoding $Z_d[x]$ with current decoder attention state $\tilde{s}_t$, to make the cp_t input-sensitive.

3.4 Loss Function

We use the negative log-likelihood function as loss function of our model:

$$loss = -\sum_{i=1}^{m} log(p(y_t^{(i)}|x^{(i)}, y_{<t}^{(i)})).$$

Two things are important to note here, (i) we use a $\langle$unk$\rangle$ whenever an out-of-vocabulary word appears in field; (ii) we never share embeddings between field's name and field's value.

4 Experiments

The experiment considers two popular benchmark datasets for data-to-text generations—WikiBio dataset and E2E dataset.

4.1 Baselines and Metrics

The following three baseline systems are considered in our experiment.

1. **Baseline 1:** It is a vanilla seq2seq model, based on the popular seq2seq (Sutskever et al., 2014) architecture and concatenative attention mechanism (Luong et al., 2015).

2. **Baseline 2:** To investigate the role of attentions in our model, this baseline model is considered where all attentions (at the decoder part) are removed from our proposed system.

3. **Baseline 3:** This is standard transformer (Vaswani et al., 2017) architecture based encoder-decoder model. We train this baseline model in our experiments from scratch.

Beside those three baselines, we consider two recent proposed data-to-text generator systems in our experiments. According to our knowledge, those systems have achieved high-scored performance on both WikiBio dataset and E2E dataset, in terms of automatic evaluations.

1. **Liu et al. (2018):** Liu et al. (2018) considers an encoder-decoder architecture, consists of a field-gating encoder (which enables structure-aware properties) and a dual attention-based decoder. Unlike our proposed encoder, Liu et al. (2018) encodes both records and fields information with parallel gating structures.

2. **Nie et al. (2019):** Nie et al. (2019) proposed an encoder-decoder model for data-to-text generation with self-attention mechanisms (Vaswani et al., 2017), for handling both sequential and non-sequential data. However, unlike our proposed model's where we incorporate both hierarchical and horizontal connections, Nie et al. (2019) mainly considers hierarchical self-attentions.

To evaluate the quality of generated text, we use three popular metrics—BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and METEOR (Banerjee and Lavie, 2005)[1]. We also use a recent data-to-text generation evaluation metric, PARENT (Dhingra et al., 2019), which considers both input data and reference texts in evaluation unlike above all three metrics which consider only reference. Dhingra et al. (2019) have extensively shown that PARENT metric correlates with human judgements more accurately than other automatic evaluations metrics.

As several past studies (Belz and Reiter, 2006; Reiter, 2018; Chaganty et al., 2018; Novikova et al., 2017a) have found that automatic evaluation is not a reliable way to evaluate data-to-text generation models, we also perform human evaluation on generated texts from our system.

4.2 Parameters Settings

The dimension of both field word and field name embedding are set to 200. We use a separate embeddings for a field name as well as for field values (i.e. vocabulary words). Adam optimization techniques (Kingma and Ba, 2015) (with initial learning rate=0.0001, β_1=0.9, β_2=0.999) are used to train the model. The depth of all BiLSTM models set as 2. In all cases, dropout value is fixed to 0.5. Most of the hyperparameters are tuned on predefined validation sets. In generating text from decoder the beam-search technique with beam size 4 is used. For baseline 3, we use standard transformer with stack size of 6 in both decoder and

[1] For BLEU, ROUGE and Meteor, https://github.com/tuetschek/e2e-metrics

dataset	instances	total words	tokens/sentence	sentences/instance	references/instance
WikiBio	728K	400K	26.1	1	1
E2E	50.5K	5.2K	14.3	1.5	8.1

Table 2: Dataset statistics.

encoder. In WikiBio dataset, while choosing sizes for vocabulary(or word types) and field types we closely follow Lebret et al. (2016). In most of the cases, we fix batch size to 32/64 and we train our model at most 120000 steps. We use NVIDIA GeForce RTX 2080 Ti graphics card for our experiments.

4.3 WikiBio Dataset and Results

Lebret et al. (2016) introduced WikiBio dataset from biographical articles on Wikipedia. Table 2 shows statistics of WikiBio dataset. From Table 3, we observe that our model achieves better outcomes in terms all automatic evaluation metrics than baselines and those two recent reported best results. Some examples of generated texts of our proposed system are given in Appendix A (Table 10).

model	BLEU	ROUGE-L	METEOR	PARENT
baseline 1	0.338	0.418	0.271	0.463
baseline 2	0.348	0.445	0.262	0.455
baseline 3	0.381	0.486	0.340	0.407
Liu et al. (2018)	0.447	0.528	0.363	0.538
Nie et al. (2019)	0.450	0.522	0.371	0.527
proposed method	**0.465**	**0.566**	**0.397**	**0.540**

Table 3: Results from WikiBio dataset.

4.4 E2E Dataset and Results

Novikova et al. (2017b) introduced E2E dataset 2 on restaurant text domain. From the comparison results presented in table 4, it is quite clear that our model outperforms the baselines and other reported systems, almost in every cases except Liu et al. (2018) model performs a bit better ($\sim$ 1% compared to our model) in terms of PARENT metric for E2E Dataset. Some samples of generated text are provided in Appendix A (Table 11).

model	BLEU	ROUGE-L	METEOR	PARENT
baseline 1	0.517	0.520	0.344	0.569
baseline 2	0.534	0.572	0.350	0.572
baseline 3	0.568	0.594	0.425	0.642
Liu et al. (2018)	0.653	0.614	0.428	**0.726**
Nie et al. (2019)	0.662	0.612	0.399	0.663
proposed method	**0.675**	**0.683**	**0.442**	0.716

Table 4: Results on E2E dataset.

4.5 Analysis from Ablation Tests

To understand effectiveness of copy module, attention gates and encoder of our model, we do component ablation study based on BLEU and PARENT scores.

4.5.1 Ablation of Copy Module

Table 5 shows the copy module ablation results. From the reported BLEU and PARENT scores, it is quite clear that the copy module plays an important role in generating better text for both datasets.

	WikiBio		E2E	
	BLEU	PARENT	BLEU	PARENT
with copy module	0.465	0.540	0.675	0.716
without copy module	0.424	0.527	0.619	0.682

Table 5: Ablation of copy module (based on BLEU and PARENT score).

4.5.2 Ablation of Attention Gates

To observe the effectiveness of the attention gates we perform ablation tests on them. In terms of BLEU score, we find very small to no improvement. However, attention gates show a clear improvement in terms of PARENT metrics scores. Moreover, while doing qualitative analysis, we observe that the quality of generated texts is improved through these attention gates. Table 6 shows such qualitative analysis results. It may be noted that our model makes a few mistakes irrespective of whether attention gates are used or not. However, in terms of quality of generated text attention gates play an affirmative role as the number of wrongly inserted words is less for the model with attention gates compared to the model without gates.

	WikiBio		E2E	
	BLEU	PARENT	BLEU	PARENT
with attention gate	0.465	0.540	0.674	0.716
without attention gates	0.458	0.513	0.680	0.694

Table 7: Ablation of attention gates (based on BLEU and PARENT score).

WikiBio	
input	name[myles wilder], birth-date[january 28, 1933], birth-place[new york city, new york], death_date[april 20, 2010 -lrb- age 77-rrb-], death-place [temecula, california], occupation [television writer and producer], spouse [bobbe wilder -lrb- survives him -rrb-], article-title [myles wilder]
reference	myles wilder -lrb- january 28 , 1933 - april 20 , 2010 -rrb- was a television comedy writer and producer .
without attention gates	myles wilder -lrb- 28 , 1933 april , -rrb- held a television comedy writer and .
with attention gates	myles wilder -lrb- january 28 , 1933 – april 20 , 2010 -rrb- was an american television writer and producer .
E2E	
input	name[Blue Spice], eatType[restaurant], food[English], area[riverside], familyFriendly[yes], near[Rainbow Vegetarian Cafe]
one reference	there is a restaurant that provides food and is children friendly, near rainbow vegetarian cafe and the riverside and is called blue spice.
without attention gates	there is a soup restaurant that provides food and good spice. friendly, near rainbow vegetarian cafe and the riverside and blue.
with attention gates	near the rainbow vegetarian café in the riverside area is a restaurant called blue spice that serves english food and is children friendly.

Table 6: A qualitative analysis for the role of attention gates in our model (wrong word, word is not available in input) [E2E dataset contains several gold references for a single input data, but due to space constraint only one reference is given.].

	WikiBio		E2E	
	BLEU	PARENT	BLEU	PARENT
with all connections	0.465	0.540	0.675	0.716
without horizontal connections	0.369	0.483	0.532	0.580
without hierarchical connections	0.414	0.498	0.581	0.673

Table 8: Ablation of encoder connections (based on BLEU and PARENT score).

4.5.3 Ablation of Encoder Connections

Through ablation test, we analyze the effectiveness of our encoder connections—both horizontal connections and hierarchical connections. Table 8 reports results of ablation test on encoder's connections. It is observed that the proposed model performed better when both connections present.

model	WikiBio		E2E	
	BLEU	PARENT	BLEU	PARENT
with skip-interconnection	0.421 ± 0.011	0.493±0.017	0.637 ± 0.009	0.682 ± 0.013
without skip-connection	0.413 ± 0.024	0.490±0.044	0.628 ± 0.021	0.652 ± 0.036

Table 9: Ablation test of skip-interconnections with shuffling records in input data.

4.6 Analysis of Model with Shuffled Input

Earlier, we have mentioned that in order to encode both sequential and non-sequential (orderless) data through our proposed encoder, we introduced skip-interconnection to effectively handle them. To be more precise horizontal connections are responsible for the sequential data encoding, whereas hierarchical connections play essential roles for the non-sequential data. Finally, the skip-interconnections use both outputs from horizontal and hierarchical connections to nullify model bias toward record/field orders. In this section, we will investigate the role of skip-interconnections with random shuffling of records of input data. The aim of this experiment is to show the effectiveness of the proposed encoder on shuffled data. This experiment is evaluated through both intrinsic and extrinsic way.

Extrinsic Evaluation: Here we conduct ablation test on skip-interconnections with shuffling records in input data on both of the datasets. On each dataset's test set, such record shuffling are performed for five times . Table 9 presents effective of proposed encoder's skip-interconnections in terms of low fluctuation (standard deviations) measures on both PARENT and BLEU metric.

Intrinsic Evaluation: To more closely observe the effect of skip-interconnections on our model in handling shuffled input data, we show t-SNE plots (Maaten and Hinton, 2008) for encoded representations of input data with our encoder. Two random data instances are sampled from each of the two datasets (WikiBio and E2E), and each data instance is shuffled close to 30 different arrangements. We show t-SNE plots of encoded representations of those shuffle data through our encoder. The well disentangled encoded representations of shuffled data (in figure 5) with skip-interconnections clearly prove effectiveness of skip-interconnections.

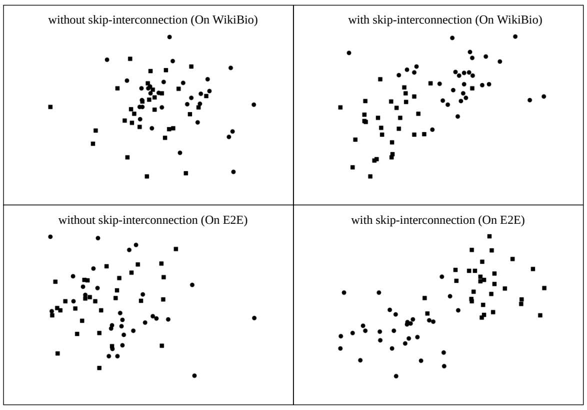

Figure 5: t-SNE plots for encoder representations on shuffled input data (circled and squared points represent two data different data instances).

4.7 Human Evaluation

In human-based evaluation for annotations purpose, we select four university graduate students from various majors. A sample of 150 generated texts from the proposed model is chosen for each of the two datasets (E2E and WikiBio) for the annotation task. Along with the generated text, we also provide input data and reference text to annotators. Every instance is annotated by at least three human annotators. In human-based evaluation, we primarily look for two essential qualities in generated texts–*adequacy* (or correctness) and *readability* (or fluency) (Gatt and Krahmer, 2018). The *adequacy* indicates whether appropriate/correct content of input data is contained within the generated text or not. The term *readability* defines fluency, clarity, and linguistic quality of a generated text. For adequacy and readability, every annotator is asked to rate each text on a scale of 1 to 5, where 5 is the best. Human evaluation results are presented in Figure 6 along with the inter-annotators agreement in terms of Krippendorff's alpha coefficient[2]. Evaluation results show that experiment on WikiBio dataset resulted in better readability and informativeness compared to the results obtained for E2E dataset.

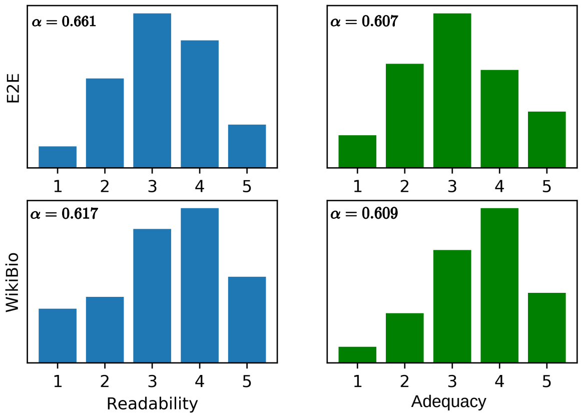

Figure 6: Average human rating of texts generated from the proposed model (**top left**:measures readability in E2E, **top right**: informativeness in E2E, **bottom left**: readability in WikiBio, **bottom right**: informativeness in WikiBio). α denotes the Krippendorff's alpha coefficient.

5 Related Works

The research presented in this paper is related to the recent data-driven data-to-text generation effort where text is generated from structured data (Angeli et al., 2010; Mei et al., 2016; Lebret et al., 2016; Liu et al., 2018). There are several types of data-driven text generation systems. Belz (2008) used probabilistic context-free grammar for text generation from structured data. Chen and Mooney (2008) introduced a strategic text generation technique for sportscasting of a simulated soccer game. Among data-driven text generations, Angeli et al. (2010) was probably the first to propose a domain-independent approach with an application on weather forecasting. With the advent of recurrent neural network language model (Mikolov et al., 2010), neural text generation models are proposed several in number and successfully applied to several different text generation tasks, from poem generation (Zhang and Lapata, 2014) to image captioning (Xu et al., 2015; Kiros et al., 2014; Karpathy and Fei-Fei, 2015). After the seq2seq model (Sutskever et al., 2014) and various attention mechanisms (Xu et al., 2015; Luong et al., 2015) are reported in the literature, the encoder-decoder model in neural text generation become quite ubiquitous. For selective generation task where readers focus on a certain selective part of the input, Mei et al. (2016) proposed an encoder-decoder model with an attention mechanism. In a concept-to-text generation where aim lies in generating text descriptions from complex concepts, the

[2] https://www.nltk.org/api/nltk.metrics.html

encoder-decoder based models also achieve high accuracy (Lebret et al., 2016; Sha et al., 2018; Liu et al., 2018; Nema et al., 2018). For the dialogue system too, this kind of data-driven approach finds some important results (Wen et al., 2015; Shang et al., 2015). The encoder-decoder model has also shown promising results on table-to-text generation task (Bao et al., 2018; Gong et al., 2019).

6 Conclusion

In this paper, we propose an effective structural encoder for encoder-decoder based data-to-text generation, which carefully encodes record-field structured input data. With extensive experiments, we show that the proposed model is capable of handling both sequential and order-less (non-sequential) data. For selecting appropriate contents from encoded data, we incorporated attention gates in the proposed model. Evaluation of the model on WikiBio and E2E dataset brings out the potential of the proposed system in generating quality text. The immediate extension of this study may consider analysis of the model's behavior at sub-task levels, i.e., its effect on content selection, text/discourse planning, or on surface realization. These experiments may unveil more interesting features of the proposed model. Moreover, further research is needed to improve the quality of output text.

Acknowledgement

This work is partially supported by Science and Engineering Research Board (SERB), Dept. of Science and Technology (DST), Govt. of India through Grant File No. SPR/2020/000495.

References

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.

Satanjeev Banerjee and Alon Lavie. 2005. 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*, pages 65–72.

Junwei Bao, Duyu Tang, Nan Duan, Zhao Yan, Yuanhua Lv, Ming Zhou, and Tiejun Zhao. 2018. Table-to-text: Describing table region with natural language. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Anja Belz. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4):431–455.

Anja Belz and Ehud Reiter. 2006. Comparing automatic and human evaluation of nlg systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Arun Chaganty, Stephen Mussmann, and Percy Liang. 2018. The price of debiasing automatic metrics in natural language evalaution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653.

David L Chen and Raymond J Mooney. 2008. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th international conference on Machine learning*, pages 128–135. ACM.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

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.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170.

Heng Gong, Xiaocheng Feng, Bing Qin, and Ting Liu. 2019. Table-to-text generation with effective hierarchical encoder on three dimensions (row, column and time). 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 3143–3152, Hong Kong, China. Association for Computational Linguistics.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th*

Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 140–149, Berlin, Germany. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

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, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1203–1213. The 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*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1412–1421. The Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

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 *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 720–730. The Association for Computational Linguistics.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

Preksha Nema, Shreyas Shetty, Parag Jain, Anirban Laha, Karthik Sankaranarayanan, and Mitesh M Khapra. 2018. Generating descriptions from structured data using a bifocal attention mechanism and gated orthogonalization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1539–1550.

Feng Nie, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019. An encoder with non-sequential dependency for neural data-to-text generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 141–146.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017a. Why we need new evaluation metrics for nlg. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252.

Jekaterina Novikova, Ondrej Dusek, and Verena Rieser. 2017b. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, Saarbrücken, Germany, August 15-17, 2017*, pages 201–206. Association for Computational Linguistics.

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.

Ehud Reiter. 2018. A structured review of the validity of bleu. *Computational Linguistics*, 44(3):393–401.

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

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the*

Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers, pages 1577–1586. The Association for Computer Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Peihao Su, David Vandyke, and Steve J. Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1711–1721. The Association for Computational Linguistics.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680.

A Sample of Generated Texts from The Proposed Data-to-Text Generation Model on WikiBio and E2E Datasets

A Sample of generated texts on WikiBio dataset	
Instance 1	
input	name[pietro micheletti], birthdate[19 october 1900], deathdate[25 march 2005] birthplace[maciano di pennabilli, italy], deathplace[maciano di pennabilli, italy] allegiance[italy italy], branch[italian army], serviceyears[1917-1925], rank[major], battles[world war i fiume's war], laterwork[farmer manager], award[" military cross " " ordine di vittorio veneto" " knight of the order of merit of the italian republic "], articletitle[pietro micheletti]
reference	pietro micheletti -lrb- 19 october 1900 – 25 march 2005 -rrb- was an italian military commander .
generated text	pietro micheletti -lrb- october 19 , 1900 – march 25 , 2005 -rrb- was a italian army manager .
Instance 2	
input	name[jason buxton], nationality [canadian], occupation[film director, screenwriter] knownfor["blackbird"], articletitle[jason buxton]
reference	jason buxton is a canadian film director and screenwriter .
generated text	jason buxton is an canadian film director and screenwriter .
Instance 3	
input	name[bert crossthwaite], fullname[herbert crossthwaite], birthdate[4 april 1887], birthplace[preston, england], deathdate[20 may 1944], deathplace[birmingham, england], position[goalkeeper], years[–1906–1907 1907–1909 -1909 –1910 1910 – 1914 1914 –1915], clubs [blackpool fulham exeter city birmingham stoke], caps[0 1 2 49 0], goals[0 0 0 0 0], articletitle[bert crossthwaite]
reference	herbert crossthwaite -lrb- 4 april 1887 – 20 may 1944 -rrb- was an english footballer who played as a goalkeeper .
generated text	herbert bert crossthwaite -lrb- 4 april 1887 – 20 may 1944 -rrb- was an english footballer who played as a goalkeeper .

Table 10: A sample of generated texts from the proposed model on WikiBio dataset (wrong words generation, missing information)

A Sample of generated texts on E2E dataset	
Instance 1	
input	name[Giraffe], eatType[restaurant], food[French], area[city centre], familyFriendly[yes], near[Raja Indian Cuisine]
reference 1	the giraffe is a restaurant - family-friendly serving french food in the city centre, near the raja indian Cuisine
reference 2	giraffe, a family-friendly restaurant in the city centre serving french food is located near raja indian cuisine.
reference 3	giraffe is a restaurant located in the city centre near the raja indian cuisine. it offers french cuisine with a family-friendly atmosphere.
reference 4	city centre based restaurant near raja indian cuisine, giraffe is family-friendly and serves french cuisine.
reference 5	giraffe is a restaurant that offers french food with a family-friendly atmosphere. it is located in the city centre near the raja indian cuisine.
generated text	giraffe is a french restaurant in the riverside area near raja indian cuisine. it is not family-friendly.
Instance 2	
input	name[Cocum], eatType[pub], near[The Sorrento]
reference 1	cocum is a pub by the sorrento.
generated text	cocum is a pub near the ranch.
Instance 3	
input	name[Giraffe], eatType[pub], near[The Bakers]
reference 1	giraffe is a pub found near the bakers.
reference 2	a pub named giraffe is near the bakers.
generated text	giraffe is a pub near the bakers.

Table 11: A sample of generated texts from the proposed model on E2E dataset (wrong words generation, missing information)