


ARTICLE

# Data-to-text generation using conditional generative adversarial with enhanced transformer

Elham Seifossadat and Hossein Sameti 

Department of Computer Engineering, Sharif University of Technology, Tehran, Iran

**Corresponding author:** Hossein Sameti; Email: [sameti@sharif.edu](mailto:sameti@sharif.edu)

(Received 12 January 2022; revised 12 August 2023; accepted 29 September 2023)

## Abstract

In this paper, we propose an enhanced version of the vanilla transformer for data-to-text generation and then use it as the generator of a conditional generative adversarial model to improve the semantic quality and diversity of output sentences. Specifically, by adding a diagonal mask matrix to the attention scores of the encoder and using the history of the attention weights in the decoder, this enhanced version of the vanilla transformer prevents semantic defects in the output text. Also, using this enhanced transformer along with a triplet network, respectively, as the generator and discriminator of conditional generative adversarial network, diversity and semantic quality of sentences are guaranteed. To prove the effectiveness of the proposed model, called conditional generative adversarial with enhanced transformer (CGA-ET), we performed experiments on three different datasets and observed that our proposed model is able to achieve better results than the baselines models in terms of BLEU, METEOR, NIST, ROUGE-L, CIDEr, BERTScore, and SER automatic evaluation metrics as well as human evaluation.

**Keywords:** Data-to-text generation; transformers; self-attention; generative adversarial network

## 1. Introduction

Data-to-text (D2T) generation, the task of automatically generating text from structured non-linguistic data or meaning representation (MR), is an important research problem for various NLP applications such as task-oriented dialog, question answering, machine translation, image captioning, selective generation systems, and search engines (Reiter 2007). Examples of MRs and their corresponding texts are shown in Fig. 1. In addition to being fluent and coherent, the generated text should cover all and only the information provided in the MRs (Rebuffel *et al.* 2022). Traditional approaches for solving this problem have used rule-based (Gkatzia, Lemon, and Rieser 2016) or grammar-based (Mille, Dasiopoulou, and Wanner 2019) methods. Recently, the use of deep learning methods has expanded. These methods have been able to produce more fluent and diverse sentences compared to traditional methods (Wang, Schwing, and Lazebnik 2017; Panagiaris, Hart, and Gkatzia 2020; Schuz, Han, and Zarriess 2021). One of the most common deep learning methods is to use the maximum likelihood estimation (MLE) loss function. D2T models trained with this method, which are generally based on RNN networks (Wen *et al.* 2015a, b, c, 2016; Dusek and Jurcicek 2016; Mei, Bansal, and Walter 2016; Nayak *et al.* 2017; Riou *et al.* 2017; Gehrmann, Dai, and Elder 2018; Juraska *et al.* 2018; Liu *et al.* 2018; Oraby *et al.* 2018; Sha *et al.* 2018) or transformers (Gehrmann *et al.* 2018; Gong 2018; Radford *et al.* 2019; Kasner and Dušek 2020a; Peng *et al.* 2020; Harkous, Groves, and Saffari 2020; Chen *et al.* 2020; Kale and Rastogi 2020a; Raffel *et al.* 2020; Chang *et al.* 2021; Lee 2021), auto-regressively generate each



(a)	
MR:	(Bananaman, broadcastedBy, BBC) && (Bananaman, creator, John_Geering) && (Bananaman, starring, Graeme_Garden) && (Bananaman, firstAired, 1983-10-03) && (Bananaman, lastAired, 1986-04-15)
Text:	Bananaman first aired on the BBC on October 3rd, 1983 and broadcast its last episode on April 15th, 1986. Bananaman last aired on 15th April 1986. Graeme Garden is one of the stars of the series which was created by John Geering.
	WebNLG
(b)	
MR:	Train-Inform(Leave=12:45)
Text:	I'm not picky on when I want to arrive I just want to leave after 12:45.
	MultiWoz
(c)	
MR:	name[The Eagle], eatType[coffee shop], food[Japanese], priceRange[less than 20], customer rating[low], area[riverside], familyFriendly[yes], near[Burger King]
Text:	The Eagle is a low rated coffee shop near Burger King and the riverside that is family friendly and is less than 20 for Japanese food.
	E2E

**Figure 1.** Sample meaning representation (MR) and text chosen from (a) WebNLG, (b) MultiWoz, and (c) E2E datasets.

output token based on the sequence of previously generated tokens. Although these methods have achieved good performance on the D2T task, they also have problems. One of the major drawbacks of these methods is exposure bias caused by the difference between the training and testing phases. In the training phase, the ground-truth tokens are used as the model input, but in the test phase, the token generated in the previous step is used as the next step input. In this case, if the previously generated token is incorrect, the tokens that are subsequently generated will also be incorrect. Another problem with these methods, which is related to the nature of MLE, is the sensitivity to rare examples in the training dataset, which leads to a cautious prediction of the real data distribution. As a result of these problems, the produced text could still be dull, unnatural, and also containing incoherent representations. Moreover, since these methods have little control over the semantic flow in the process of generating text, the output sentences have duplicate and redundant information or lack parts of the input information in the output text (Juraska and Walker 2021).

In contrast to MLE, there is another class of deep learning-based methods called generative adversarial network (GAN) (Goodfellow *et al.* 2014), which aims to generate texts by an unsupervised learning approach (Kusner and Hernández-Lobato 2016; Zhang *et al.* 2017; Yu *et al.* 2017; Che *et al.* 2017; Lin *et al.* 2017; Gou *et al.* 2017; Fedus, Goodfellow, and Dai 2018; Xu *et al.* 2018; Wang and Wan 2018; Chen *et al.* 2018; Nie, Narodytska, and Patel 2019; Jiao and Ren 2021). The GAN model includes the generator and discriminator networks. The generator network tries to generate a sample similar to the real sample by taking the noise signal, and the discriminator network moves the generator in the right direction and generates samples closer to real samples by distinguishing real samples from the fake ones. This results in solving the exposure bias problem. However, training of the GANs faces challenges such as the instability of the training process and the mode collapse problem (Ledig *et al.* 2017). Mode collapse occurs when the generator finds a small number of template that fool the discriminator. Therefore, generator is not able to generate any sentences other than this limited set of templates.

To improve the quality of the generated sentences in terms of semantics, verbal, and diversity, we propose a novel model for D2T generation named conditional generative adversarial with enhanced transformer (CGA-ET). The generator network in the proposed model is our enhanced version of the vanilla transformer encoder and decoder (Vaswani *et al.* 2017). Since input of D2T models is an MR containing a set of distinct slots, which must be expressed in the output text completely without redundancy or repetition, we modified self-attention of each encoder block with a diagonal mask and added history of each slot's attention weights to multi-head cross-attention of its decoder. Furthermore, for strengthening the generator in order to generate diverse and verbally correct sentences, we use the triplet network (Hoffer and Ailon 2015), including the vanilla transformer encoder blocks, as the discriminator network. The triplet network takes triplet sets of input MR, real and fake sentences as input, extracts sets of rich context vectors from them, and tries to update its parameters in a way that the semantic similarity between the input MR and the real sentence is greater than the semantic similarity between the input MR and the fake sentence. In this way, the discriminator networks guide the generator to generate sentences that are semantically and structurally close to the input MR as well as the real sentence. In addition, in order to stabilize the model training process, the WGAN-GP (Gulrajani *et al.* 2017) loss function is used. To evaluate the performance of the CGA-ET model, experiments are performed on the WebNLG (Castro Ferreira *et al.* 2020), MultiWoz (Budzianowski *et al.* 2018), and E2E (Dusek, Howcroft, and Rieser 2019) datasets. The results of comparing the CGA-ET with state-of-the-art models demonstrate the superiority and efficiency of our proposed model.

The major contributions of this paper are summarized below:

- We introduce an enhanced version of the vanilla transformer encoder and decoder for the D2T generation task. In this model, a diagonal mask matrix is added to the self-attention scores of each encoder block, and the history of multi-head cross-attention scores of previously generated tokens between encoder and decoder is used to compute attention weights of the current token in the decoder. This decreases semantic errors in the output sentences.
- We introduce a new model called CGA-ET, which uses our enhanced version of the vanilla transformer encoder and decoder as the generator and the triplet network containing transformer encoder blocks as the discriminator.
- We use a combination of the WGAN-GP and triplet loss functions to train the model, so while stabilizing the model training process, by controlling the semantic similarity of the sentences and then generating diverse sentences, the mode collapse problem is prevented.

The rest of our paper is structured as follows: Section 2 presents related works. Section 3 presents the proposed CGA-ET model. Datasets, experimental setups, and evaluation metrics are described in Section 4. The resulting analysis and case studies are presented in Section 5. The paper is concluded in Section 6.

## 2. Related work

The main focus of our proposed model is on improving the semantic quality of generated sentences in the field of D2T generation. Due to the importance of semantic fidelity to the given input in the output sentences, so far many efforts have been made to preserve the meaning in tasks such as abstractive summarization, dialog generation, and D2T generation, selected works of which are mentioned below.

To prevent the generation of repetitive and redundant phrases and words in abstractive summarization, See *et al.* (2017) propose a hybrid pointer-generator architecture that copies words from the source text to improve accuracy by handling OOV words along with learning to generate new words. Also, they propose a coverage mechanism (COV) that keeps track of words that have

been summarized and that makes a model aware of its attention history. Li *et al.* (2018) propose a model that incorporates entailment knowledge into the summarization in an encoder–decoder structure. Their proposed encoder jointly learns summarization generation and entailment recognition, so it can grasp both the gist of the source sentence and be aware of entailment relationships. Furthermore, they train the decoder by entailment reward augmented maximum likelihood training to encourage it to produce a summary entailed by the source. In the same direction, Perez-Beltrachini and Lapata (2021) propose an attention mechanism that encourages the decoder in each time step to pay greater attention to a subset of input representations that are both relevant and diverse.

In the dialog generation task, to reduce missing or misplacing slot values in the generated output, Li *et al.* (2020) propose a dialog state tracker with reinforcement learning (RL). Their model uses a slot consistency reward which is the cardinality of the difference between the generated template and the slot–value pairs extracted from the input dialog act. Rashkin *et al.* (2021) use DialoGPT (Zhang *et al.* 2017) model and fine-tune it on their dialog dataset. Also, they introduce a set of control codes, entailment, objective voice, and lexical precision and concatenate them with dialog inputs to encourage the model to generate responses that are faithful to the provided evidence.

To improve faithfulness to resources in the D2T generation task, Liu *et al.* (2021) propose a two-step generator with a separate text planner, which is augmented by auxiliary entity information. First, the text planner predicts the content plan based on the input data. Second, given the input data and the predicted content plan, the sequence generator generates the text. Tian *et al.* (2019) first define a confidence score to detect semantic errors in each time step  $t$  of the decoder by using an attention score to measure how much the model is attending to the source and a language model to judge if a word conveys source information. Then, they propose a variational Bayes training framework that encourages a model to generate output text with high confidence, while learning the confidence score parameters at the same time. Wang *et al.* (2020) enhance the vanilla transformer model with two content-matching constraints. By using a latent-representation-level matching constraint, they encourage model to generate text semantically consistent with input data. Also, by using an explicit entity-level matching scheme, they control that the entities of input and the corresponding text are identical. PlanGen, proposed by Su *et al.* (2021), consists of a content planner to predict the most probable content plan from the input data and a sequence generator that takes the input data and predicted content plan and generates the output text. Then by training this model using a structure-aware RL objective, the model is encouraged to be faithful to the predicted content plan.

In the proposed model, we used a different approach than the works mentioned above. In the proposed model, to generate sentences with semantic fidelity to the input MRs, an adversarial learning method with an objective function, including constraints on the semantic distance between the generated sentences and their corresponding MRs, is used. The generator network in the proposed model is an enhanced version of the base transformer network. In this enhanced version, attention mask and history information are added to the encoder and decoder self-attention layers, respectively. In this way, this transformer network will be more compatible with D2T generation. The generator network in the proposed model is an enhanced version of the base transformer network. In this enhanced version, attention mask and history information are added to the encoder and decoder self-attention layers, respectively. In this way, this transformer network will be more compatible with D2T generation.

It is necessary to note that efforts have been made before to enhance the transformer model which is completely different from our proposed method. For example, Su *et al.* (2021) introduce a method called Rotary Position Embedding to effectively leverage the positional information into the training process of pretrained language models. In this method, absolute positions are encoded with a rotation matrix and incorporate explicit relative position dependency in self-attention layers formulation. The studies conducted on this method are shown that the objective function of

the pretraining process converges faster and dependency between words in very long sentences is better learned by this enhanced model compared to the base transformer model. In order to reduce the training time and floating-point operations of attention-based language models, a new network, named GroupBERT, is introduced by modifying the structure of the transformer network of Chelombiev *et al.* (2021). In this model, a dedicated grouped convolution module is added to each self-attention layer to better learn local and global interactions between tokens. Also, the density of fully connected layers is reduced by using grouped transformations. This model has achieved better results compared to the BERT model on language representation learning. Also, its efficiency improved both in terms of floating-point operations and time-to-train. To improve the quality of word alignment in the source text and target text in NMT, Song *et al.* (2020) added a dedicated head to the multi-head transformer, which is supervised during training. The evaluation results show that using this head to derive better word alignment leads to improvement in dictionary-guided decoding. In the field of D2T generation, Gong, Crego and Senellart (2019) apply a classification layer on the output of the encoder to define which slot and values of input MRs should be expressed in the output text and which ones not.

### 3. Approach

We use a combination of the transformers model and the adversarial learning method to generate text from nonlinguistic, structured input data. The general outline of our proposed model is shown in Fig. 2. The input of the proposed model is the MRs consisting of slot and value pairs and a noise vector drawn from a normal distribution  $N(0, 1)$ , and the output of the model is the descriptive text equivalent to the input MR. The proposed model consists of two parts: a generator and a discriminator. The generator network is the vanilla transformer encoder and decoder (Vaswani *et al.* 2017). This network gets the input MRs and the noise vector, then by using Gumbel-SoftMax (Kusner and Hernández-Lobato 2016) in the output layer, generates a conditional probabilistic distribution of the vocabulary tokens. Since the generator must be trained to produce sentences that are close to the real sentences, both verbally and semantically, a triplet network (Hoffer and Ailon 2015) consisting of the vanilla transformer encoder blocks is used as the discriminator network. The input to this network is a triplet including the real sentence, the fake sentence generated by the generator, and the input MR. The objective function of this network is defined in such a way that the semantic distances of the input MRs from the real sentences are less than the semantic distances of the input MRs from the fake sentences. In this way, the discriminator teaches the generator how to express the information of the input MRs to generate correct sentences that are semantically close to the real sentences. In addition, to avoid the mode collapse problem and make the model training process more stable, the WGAN-GP (Gulrajani *et al.* 2017) loss function is also used in the discriminator objective function. The details of our proposed model are given in the following subsections.

#### 3.1. Generator

As mentioned earlier, the generator network used in this work is the vanilla transformer with six encoder and decoder layers. Encoder layers are made up of blocks with similar structures, including multi-heads self-attention and feed-forward sublayers. For each word of the encoder input sequence, first, the encoding vector resulting from the sum of the word embedding vector and positional encoding vector is produced. The produced encoding vector then is converted into sets of three vectors  $q$ ,  $k$ , and  $v$  depending on the number of heads. Afterward, for each head, vectors  $q$ ,  $k$ , and  $v$  of all words are stored in matrices named  $Q$ ,  $K$ , and  $V$  with  $d$  dimension, and they are used to calculate the self-attention vector as follows:

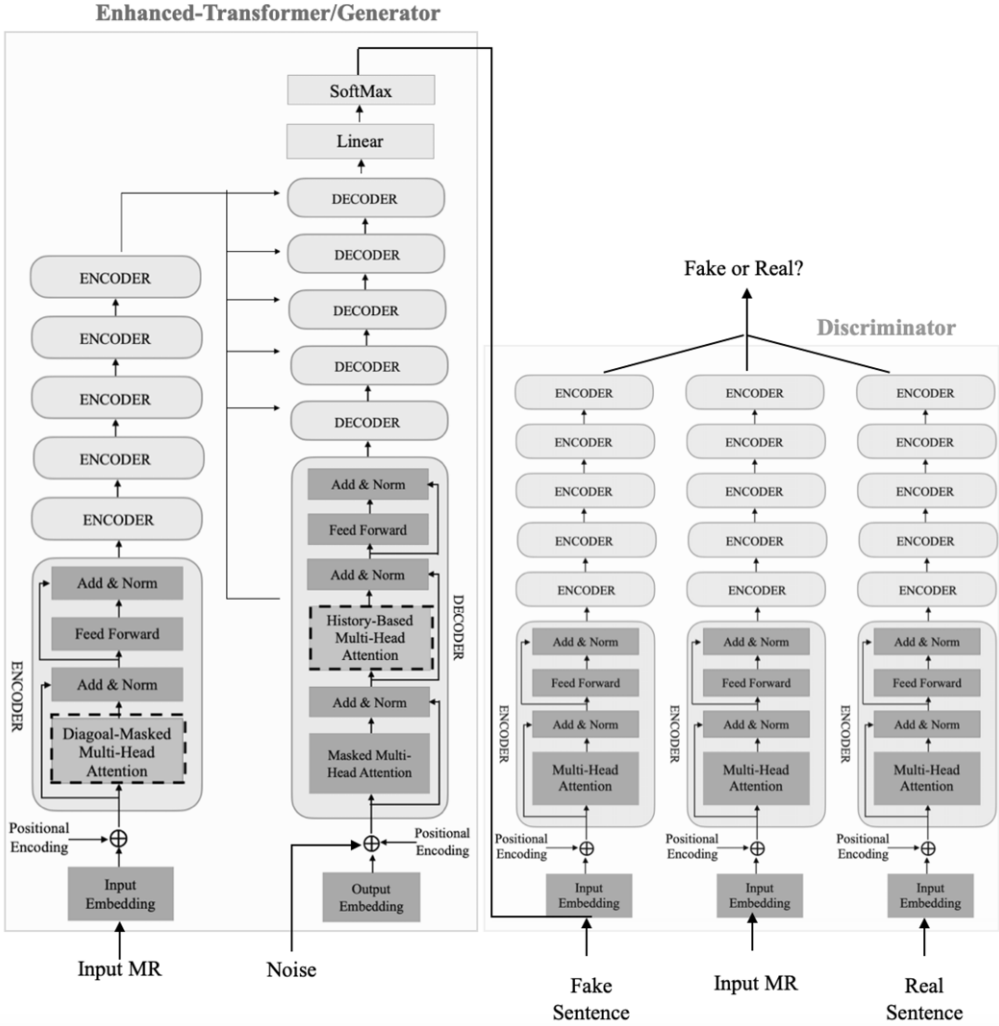


Figure 2. The block diagram of CGS-ET model. In the generator diagram, improvements made to the transformers-based generator network are shown as blocks with dashed borders. The discriminator diagram shows a triplet network that consists of three vanilla transformer encoders with sharing weights.

$$Scaled\ Dot-Product = \frac{Q \cdot K^T}{\sqrt{d}} \tag{1}$$

$$Attention(Q, K, V) = SoftMax(Scaled\ Dot-Product) \cdot V$$

The output of the SoftMax function shows the contribution of the other input sequence words in creating the final attention vector for each word. For example, when the input of the encoder is a sentence like “This morning I went to the bank that is near the river bank to open an account,” words like “open” and “account” contribute more to the attention vector of the first “bank,” and for the second “bank” more attention weight is given to “river.” After attention vectors for all tokens are generated in each layer, the attention vectors of all heads are concatenated and after passing through a feed-forward layer with the linear activation function, the final attention vector is made. This vector is then passed through the normalization layer with residual connection and is given

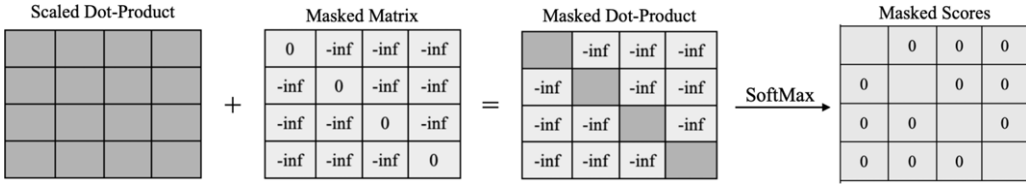


Figure 3. Generating *Masked Scores* for the self-attention sublayer of each encoder block.

as an input to the next encoder layer. For the D2T generation task, the input of encoder is not a sentence, but an MR includes a number of slots such as “*name[The Eagle], eatType[coffee shop], food[Japanese], priceRange[less than 20], customer rating[low], area[riverside], familyFriendly[yes], near[Burger King].*” Each slot has its own independent concept. Therefore, it is not right to generate the attention vector for each slot based on other slots in MR. For this reason, before applying the SoftMax function for each head, the output of the scaled dot product step is summed by a *Diagonal Mask* matrix. In this way, a MR vector is created for each input slot independent of the other slots (Fig. 3). Therefore, the attention vector for the encoder layers is calculated as follows:

$$\begin{aligned}
 \text{Scaled Dot-Product} &= \frac{Q \cdot K^T}{\sqrt{d}} \\
 \text{Masked Dot-Product} &= \text{Scaled Dot-Product} + \text{Mask} \\
 \text{Masked Scores} &= \text{SoftMax}(\text{Masked Dot-Product}) \\
 \text{Attention}(Q, K, V) &= \text{Masked Scores} \cdot V
 \end{aligned}
 \tag{2}$$

where *Mask* is a diagonal matrix whose diagonal elements are 0 and its other elements are *-inf*.

Decoder layers in the vanilla transformer are also made of similar blocks including multi-head cross-attention, feed-forward, and multi-heads encoder–decoder attention sublayers where the last one is responsible for calculating the attention scores between the decoder *Q* and encoder *K* and *V* vectors. Both decoder attention layers of the vanilla transformer have the same computation as in Equation (1), without the use of a diagonal mask. The output of the last layer of the decoder passing through a feed-forward layer with the RELU activation function and then a feed-forward layer with the linear function generates a token based on the previously generated ones. Since in the D2T generation task, all the encoder input slots, not the redundant ones and also without repetition, must be expressed in the decoder output, it is necessary for the decoder to consider the weights previously given to each slot to calculate the current attention weight scores; as a result, duplicate or lack of expression of the slots in the output text will be prevented. For this purpose, attention vectors generated for previous tokens, which we called *History*, and the attention vector for the current token are given to a *sigmoid* gate. So, by using this gate, the attention vector between encoder and decoder is calculated based on a weighted combination of history and current information (Fig. 4):

$$\begin{aligned}
 \text{Attention} &= \text{SoftMax}\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V \\
 \text{History} &= \text{Concat}([0, 0, \dots, 0] \text{ Attention}[: - 1]) \\
 \text{History-based Attention}(Q, K, V) &= W_a \text{ sigmoid}(W_b \text{ Attention} + W_c \text{ History})
 \end{aligned}
 \tag{3}$$

where  $W_a$ ,  $W_b$ , and  $W_c$  are weight matrices that are learned during the model training process. Moreover, to solve the gradient back-propagation problem in adversarial training, in this work,

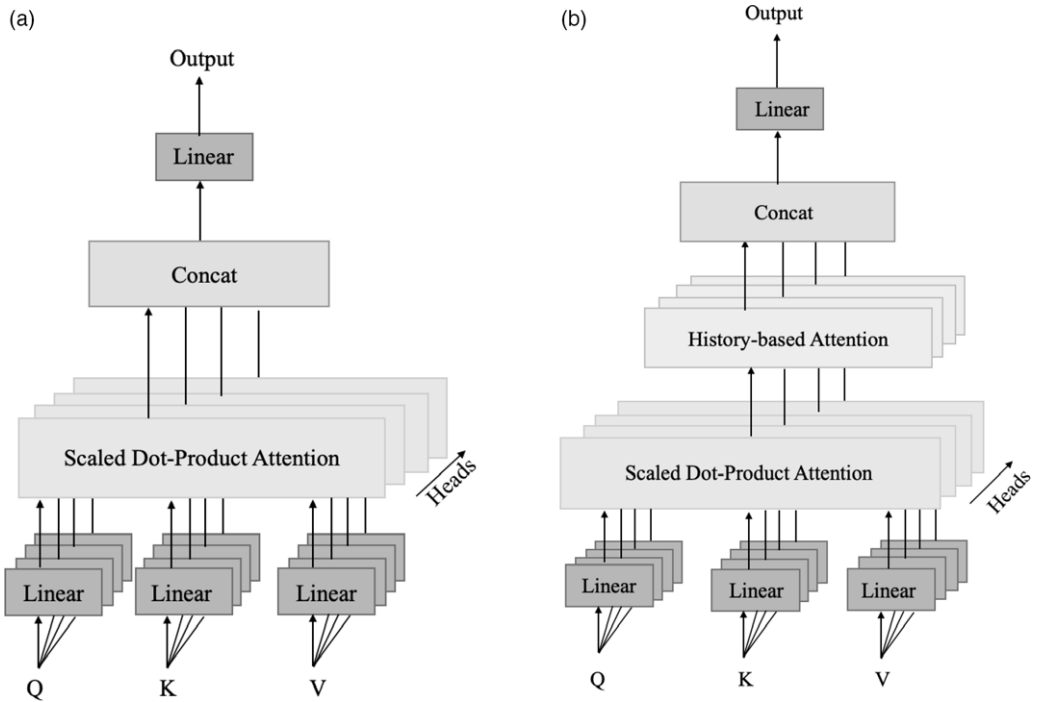


Figure 4. Generating the encoder-decoder self-attention vector in (a) the vanilla transformer and (b) the proposed model.

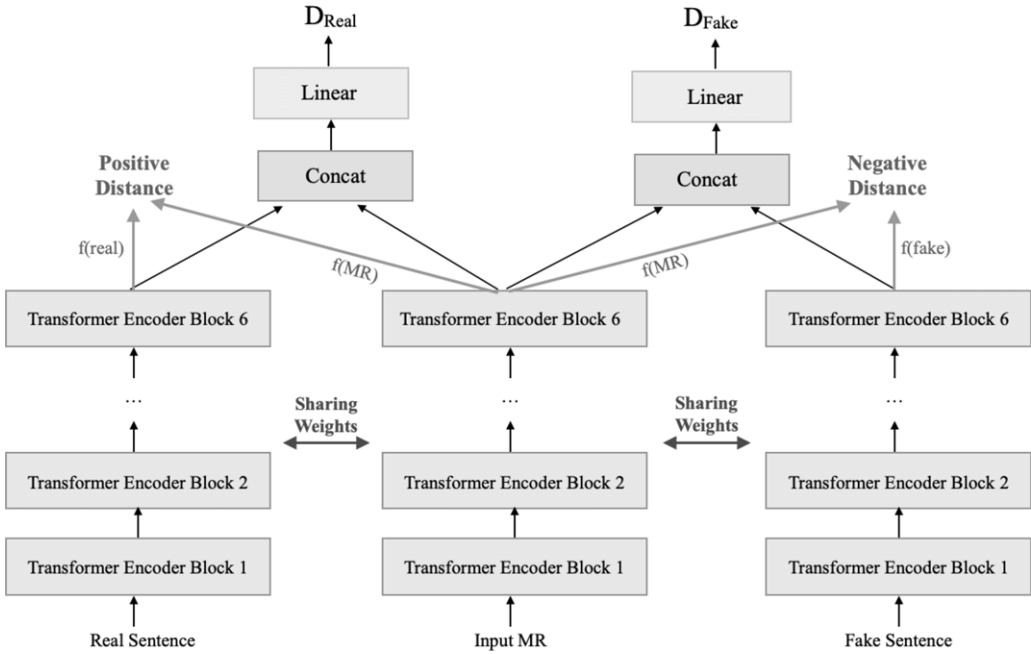
the Gumbel-SoftMax function is used instead of the SoftMax function in the last layer of the decoder.

### 3.2. Discriminator

The generator network needs a discriminative signal from the discriminator network to be trained. The more information the received signal contains, the more successful the generator training is and the closer the generated sentences are to real sentences. The closeness of real and fake sentences can be defined at two levels: word level and semantic level. Using the MLE objective function, the generator can be trained in a teacher-forcing manner (Bengio *et al.* 2015) to generate a sentence close to the real sentence at the level of the words. However, due to the MLE drawbacks such as exposure bias and sensitivity to rare examples resulting in conservative learning of real data distribution, this objective function cannot guarantee that the generated sentence is close enough to the real sentence in terms of maintaining the semantic structure and expressing the input MRs information. For this reason, in this work, the task of extracting the semantic features of sentences and estimating the distance between them is entrusted to the discriminator network.

The proposed discriminator is a triplet network composed of the transformer encoder blocks (Fig. 5). This network takes triplets including the input MR, the real sentence, and the generated sentence as an anchor, positive, and negative samples, respectively. Then it extracts the semantic features of each triplet in such a way that in the created semantic space the sum of the distance between the input MR and the real sentence (positive distance) with a margin value is still less than the distance between the input MR and the fake sentence (negative distance) (Hermans, Beyer, and Leibe 2017). In this work, to calculate the positive and negative distances between the semantic features of sentences, the element-wise cosine distance is used as follows:





**Figure 5.** Discriminator network. The hidden state vectors obtained from the encoders contain the semantic information of the input sentences. The discriminator network is trained in such a way that the distance between the semantic vectors of the real sentence and MR (positive distance) is less than the distance between the semantic vectors of the fake sentence and MR (negative distance). The discriminator network also tries to increase the distance between the real and predicted distributions of data ( $D_{Real}, D_{Fake}$ ).

$$\begin{aligned}
 \text{Positive Distance} &= 1 - \left( \frac{f_{Real}}{\|f_{Real}\|_2} \right) \left( \frac{f_{MR}}{\|f_{MR}\|_2} \right)^T \\
 \text{Negative Distance} &= 1 - \left( \frac{f_{Fake}}{\|f_{Fake}\|_2} \right) \left( \frac{f_{MR}}{\|f_{MR}\|_2} \right)^T
 \end{aligned}
 \tag{4}$$

where  $f_{Real}$ ,  $f_{Fake}$ , and  $f_{MR}$  indicate the semantic feature vector extracted by the encoder blocks from the real sentence, fake sentence, and input MR, respectively. In addition, in order to make the training process of the model more stable, the extracted semantic feature vectors are concatenated in pairs of real and anchor sentences as well as fake and anchor sentences, then these concatenated vectors are passed through a feed-forward layer with one output neuron and the linear activation function. The value of the output neuron is considered as the output discriminator for real and fake sentences, and their distances show how close their distributions are. In this way, in addition to discovering which sentence is better, the discriminator teaches the generator how to generate sentences that are semantically close to the real sentences.

### 3.3. Objective functions and training process

To train the discriminator network with the aim of increasing the semantic distance between real and fake sentences, the discriminator objective function is defined as follows:

$$L_D = L_{Triplet} + L_{WGAN-GP}
 \tag{5}$$

where discriminator tries to minimize this objective function by calculating  $L_{Triplet}$  and  $L_{WGAN-GP}$  as follows:

$$L_{Triplet} = \mathbb{E}[Positive\ Distance - Negative\ Distance + Margin] \quad (6)$$

$$L_{WGAN-GP} = \mathbb{E}(D_{Fake}) - \mathbb{E}(D_{Real}) + \lambda \mathbb{E}(\|\nabla D_{Interpolate}\|_2 - 1)^2$$

where  $\nabla$  denotes gradient and  $D_{Interpolate}$  is a linear mixture of the real and generated sentences. Accordingly, the generative objective function is also defined as follows:

$$L_G = \|f_{Real} - f_{Fake}\|_2^2 - \mathbb{E}(D_{Fake}) + L_{MLE} \quad (7)$$

where  $\|f_{Real} - f_{Fake}\|_2^2$  is the Euclidean distance between real and fake sentences semantic features. Also,  $L_{MLE}$  is a regularization term that calculates by using categorical cross-entropy between the real sentence and the sentence generated by generator network in a teacher-forcing manner. So by minimizing this loss, the value of the discriminator objective function is increasing.

---

**Algorithm 1:** Training process of our proposed model.

---

- 1: **Require:** Conditional-Generator  $G_\theta$ , Discriminator  $D_\phi$ , Training data distribution  $p_{data}$ , Max length sentence  $T$
  - 2: **Initialization:** Initialize  $G_\theta$  and  $D_\phi$  with random weights
  - 3: **Begin**
  - 4: **while** not done **do**
  - 5:     Sample a batch of (MR, Real sentence) pairs from training dataset
  - 6:     Sample  $Noise \sim N(0, 1)$
  - 7:     **for**  $i = 0$  **to**  $i = T$  **do**
  - 8:         Representation $_{i+1} = G_\theta(\text{Token}_{0:i} | \text{MR}, \text{Noise})$
  - 9:         Token $_{i+1} = \text{Gumbel-Softmax}(\text{Representation}_{i+1})$
  - 10:     **end for**
  - 11:     Fake Sentence = Token $_{0:T}$
  - 12:      $F_{Fake}, F_{Real}, F_{MR}, D_{Real}, D_{Fake} = D_\phi(\text{Fake Sentence}, \text{Real Sentence}, \text{MR})$
  - 13:     Compute distances with Equation (4)
  - 14:     Compute  $L_D$  with Equation (5)
  - 15:     Update  $\phi$
  - 16:     Compute  $L_G$  with Equation (7)
  - 17:     Update  $\theta$
  - 18: **end while**
- 

The whole training process of the proposed method is illustrated in Algorithm 1. As observed, to calculate the  $L_{MLE}$ , the generator network generates the output sentence in a teacher-forcing manner for the given MR, while to generate a fake sentence the tokens generated at each step are used as the input of the next step. Moreover, the generator and discriminator networks in the proposed model do not require any pretraining and both networks are trained adversarially from the beginning.

**Table 1.** Datasets statistics. *Attributes* shows the total number of slot types and *Avg-Len* indicates average length of sentences in each dataset

Dataset	#Train	#Validation	#Test	Attributes	Avg-Len
WebNLG	18,081	2260	4928	373	29.69
MultiWoz	8759	1001	768	27	14.72
E2E	33,525	4299	4693	8	20.10

## 4. Experiments

Experiments to evaluate the performance of the proposed CGA-ET model and compare it with other state-of-the-art models are described in this section. The used datasets, experimental setups, evaluation metrics, experiments results, and their analysis are described below.

### 4.1. Datasets

To qualitatively evaluate the proposed CGA-ET model, we used three datasets: (1) the enriched version of WebNLG + 2020 (Castro-Ferreira *et al.* 2018; Castro Ferreira *et al.* 2020), which is the modified version of the original WebNLG dataset (Gardent *et al.* 2017), containing sets of RDF triples (Subject, Predicate, and Object) extracted from DBpedia and their corresponding texts of 10 seen domains in the training data and 5 unseen domains of the test data; (2) Multiwoz 2.1 (Budzianowski *et al.* 2018) consisting of conversation sequences and their equivalent semantic representations in seven domains; and (3) cleaned version of E2E dataset (Dusek *et al.* 2019) containing descriptions of restaurants with their equivalent MR in form slots and values. Details of these datasets are given in Table 1.

### 4.2. Experimental setups

The proposed CGA-ET model is implemented using the TensorFlow library and trained on Google Colaboartory with one Tesla P100-PCIE-16 GB GPU for 20k steps. Encoder and decoder in both generator and discriminator networks include six multi-head attention layers with eight heads. The model dimensions and fully connected layers size are set to 256, the batch size is set to 10, and the dropout rate is set to 10%. The CGA-ET model at first is initialized with random weights and then optimized using an Adam optimizer with a learning rate of  $1e-4$ . This process is terminated by early stopping based on the validation loss. Moreover, for every 50 steps, L2-regularization with  $\lambda = 1e - 5$  is added to loss values. In the inference phase, beam search with width 10 is used, and for each MR, then the top sentence is selected based on its negative log-likelihood value.<sup>a</sup>

### 4.3. Automatic evaluation metrics

To automatically evaluate the quality of sentences produced by the proposed CGA-ET model and compare them with baseline models on the E2E dataset, as in the E2E challenge, BLEU (Papineni *et al.* 2002), METEOR (Lavie, Sagae, and Jayaraman 2004), NIST (Martin and Przybocki 2000), ROUGE-L (Lin 2004), and CIDEr (Vedantam, Zitnick, and Parikh 2015) metrics are used. The BLEU, METEOR, and BERTScore (Zhang *et al.* 2020) metrics used in the WebNLG + 2020 challenge are also used for the WebNLG dataset. For the MultiWoz dataset, BLEU and slot error rate

<sup>a</sup>Code and data will be published in <https://github.com/seifossadat/CGA-ET>

**Table 2.** Results on WebNLG seen and unseen test data. The best and second-best models are highlighted in **bold** and underline face, respectively

Model	Seen			Unseen		
	BLEU	METEOR	BERTScore F1	BLEU	METEOR	BERTScore-F1
mBART	59.13	42.2	0.960	42.24	37.5	0.943
PlanEnc	64.42	45.0	0.962	38.23	37.0	<b>0.954</b>
T5-small	62.60	45.0	–	38.80	37.0	–
T5-base	64.70	46.0	–	49.40	41.0	–
T5-large	63.90	46.0	–	<b>52.80</b>	41.0	–
T5-3B	62.80	45.0	–	52.00	<b>42.0</b>	–
OSU Neural NLG	61.20	43.4	0.962	<u>52.40</u>	<u>41.6</u>	<u>0.951</u>
FBConvAI	61.30	41.6	0.962	50.30	41.4	0.950
BT5	61.10	43.3	<u>0.963</u>	50.80	41.5	0.947
CGA-ET	<b>65.10</b>	<b>46.5</b>	<b>0.970</b>	50.60	41.0	0.940

(SER) metrics are used. SER is the fraction of times where at least one slot value from the input MR is not expressed or incorrectly expressed in the output sentence (Wen *et al.* 2015c).

#### 4.4. Human evaluation

Human evaluation is performed to evaluate the semantic quality of the generated sentences by our proposed model in terms of *faithfulness* (How many of the semantic units in the given sentence can be found/recognized in the given MR/RDF), *coverage* (How many of the given MRs slots values/RDF triples can be found/recognized in the given sentence), and *fluency* (whether the given sentence is clear, natural, grammatically correct, and understandable). For fluency, we asked the judges to evaluate the given sentence and then give it a score: 1 (with many errors and hardly understandable), 2 (with a few errors, but mostly understandable), and 3 (clear, natural, grammatically correct, and completely understandable). As judges, 20 English speakers are chosen (Fleiss's  $\kappa = 0.65$  Landis and Koch 1977 and Krippendorff's  $\alpha = 0.58$  Krippendorff 2011) for evaluating 50 tests MRs that were randomly selected from each dataset. To avoid any bias, judges are shown a randomly selected MR at a time, together with its gold sentence and the generated models. The judges were not aware of the model each output sentence is generated by and the sentences are presented each time in a different order.

#### 4.5. Results and analysis

##### 4.5.1. Results on WebNLG

The results of the BLEU and METEOR automatic metrics for the output of the CGA-ET and baseline models on the seen, unseen, and full test data are given in Tables 2 and 3. The baseline models used for comparison include the following:

**Table 3.** Results on WebNLG full test data. The best and second-best models are highlighted in **bold** and underline face, respectively

Model	BLEU	METEOR	BERTScore-F1
DATATUNER	52.40	42.4	-
mBART	50.34	39.8	0.951
PlanEnc	52.78	41.0	0.958
T5-small	52.00	41.0	-
T5-base	55.20	43.0	-
T5-large	<u>57.10</u>	<u>44.0</u>	-
T5-3B	54.00	43.0	-
OSU Neural NLG	53.50	41.4	0.956
FBConvAI	52.70	41.3	0.956
BT5	51.70	41.1	0.954
ReGen	56.30	42.5	-
CGA-ET	<b>58.01</b>	<b>44.5</b>	0.960

- DATATUNER (Harkous *et al.* 2020), which first fine-tuned the GPT-2 pretrained language model on the WebNLG dataset and then by using the RoBERTa (Liu *et al.* 2019) model as a semantic fidelity classifier it detects and avoids generation errors.
- mBART (Kasner and Dušek 2020b), which is fine-tuned on the WebNLG dataset.
- T5-small, T5-large, T5-base, and T5-3B (Kale and Rastogi 2020b), which are different versions of T5 pretrained model that are fine-tuned on the WebNLG dataset.
- ReGen (Dognin *et al.* 2021), which is the T5 model that fine-tuned using REINFORCE method (Williams 1992)

The top models participating in the *WebNLG + 2020* challenge (Castro Ferreira *et al.* 2020) are also selected for comparison:

- PlanEnc (Zhao, Walker, and Chaturvedi 2020), which consists of two graph convolution network-based encoders to extract both structural information and sequential content of the input graph. The decoder in this model is an LSTM conditioned on both encoder outputs, with attention and copy mechanism.
- OSU Neural NLG (Li *et al.* 2020), which fine-tuned the T5 pretrained language model on the WebNLG dataset.
- FBConvAI (Yang *et al.* 2020), which fine-tuned the BART (Lewis *et al.* 2020) model on the WebNLG dataset and proposed two different methods for giving a graph as input to the BART model.
- BT5 (Agarwal *et al.* 2020), which is the T5 model that is trained and fine-tuned bilingually on the WebNLG dataset.

As shown, the proposed model has outperformed all the baselines on seen and full test data and got the highest BLEU, METEOR, and BERTScore values. This advantage is due to the fact that in the process of training our CGA-ET model's generator and discriminator, by forcing the reduction

**Table 4.** Results of human evaluations on WebNLG dataset in terms of *Faithfulness*, *Coverage*, and *Fluency* (rating out of 3). The symbols \* and † indicate statistically significant improvement with  $p < 0.05$  and  $p < 0.01$ , based on the paired t-test and the ANOVA test, respectively

Model	Faithfulness (%)	Coverage (%)	Fluency
PlanEnc (Zhao <i>et al.</i> 2020)	97.05	83.11	2.73
OSU Neural NLG (Li <i>et al.</i> 2020)	94.40	81.52	2.65
FBConvAI (Yang <i>et al.</i> 2020)	95.73	76.01	2.78
BT5 (Agarwal <i>et al.</i> 2020)	93.12	74.34	2.61
CGA-ET	98.47*	85.20*	2.80†

of the semantic distance of the generated sentences with the input RDF triples, the generator has been successful in learning how to express all semantic elements of the given RDF triples in the output sentences. But for the unseen test set, our model performed moderately, because the generator has not trained in generating tokens for expressing unseen concepts. Moreover, as subjective evaluations, we compared the outputs generated by our CGA-ET model, with the published outputs of baseline models in terms of *faithfulness* (precision), *coverage* (recall), and *fluency* and reported its results in Table 4. Since the proposed model has a better ability to express the predicates in the input RDF than baseline models, it has achieved higher faithfulness and coverage values. Also, controlling the output of the generator using similarity in both levels of words and semantic has resulted in more fluent outputs. An example of the input RDF and output sentences generated by the CGA-ET and baseline models is given in Table 5.

#### 4.5.2. Results on MultiWoz

The experiments performed on the MultiWoz dataset are aimed at generating sentences from the input MRs. The results of the BLEU and SER automatic metrics for the output of the CGA-ET and transformer-based baseline models are given in Table 6. The baseline models used for comparison include the following:

- Hierarchically disentangled self-attention or HDSA (Chen *et al.* 2019), a transformer-based model that encodes the input MR into a multilayer hierarchical graph and assigns separate head attentions to each node in the generated graph.
- SC-GPT2 (Peng *et al.* 2020) is the GPT-2 model pretrained on a large D2T corpus and are fine-tuned on the MultiWoz dataset.
- Schema and T2G2 (Kale and Rastogi 2020a) are two methods of generating text using a common model. In this model, the semantically correct but possibly incoherent and ungrammatical utterances are first generated for an input MR by using schema-guided or template-guided approaches. These utterances are then rewritten into natural sentences using T5.
- T5-small, T5-large, T5-base, and T5-3B (Kale and Rastogi 2020b) that are different versions of the T5 pretrained language model fine-tuned on the MultiWoz dataset.

As one can observe from Table 7, the proposed model is able to obtain the lowest value of SER and the highest value of BLEU compared to the baseline models. A lower value of SER, which means fewer semantic defects in the generated sentences, leads to an improvement in the value of BLEU. Moreover, we evaluated the output sentences of our model using BERTScore for which a

**Table 5.** Comparison of the generated sentences from the WebNLG dataset for our proposed model and baselines. The missed meaning labels for each generated sentence are shown in red

<b>MR</b>	Nord_(Year_of_No_Light_album)  followedBy Live_at_Roadburn_2008_(Year_of_No_Light_album) ..... & Nord_(Year_of_No_Light_album)  producer Year_of_No_Light ..... & Nord_(Year_of_No_Light_album)  artist Year_of_No_Light ..... & Nord_(Year_of_No_Light_album)  recordLabel Crucial_Blast
<b>Reference</b>	Year of No Light is both the artist and the producer of the album Nord. The album was released on the label Crucial Blast. Year of No Light’s next album was Live at Roadburn 2008
PlanEnc (Zhao <i>et al.</i> 2020)	The Year of No Light album was produced by Year of No Light and released on the record label Crucial Blast. The album was followed by Live at Roadburn 2008. artist <i>(BLEU = 0.7123, BERTScore_F1 = 0.9246)</i>
OSU Neural NLG (Li <i>et al.</i> 2020)	The album Year of No Light was produced by Year of No Light and was followed by Live at Roadburn 2008. It was released on the record label Crucial Blast. artist <i>(BLEU = 0.6649, BERTScore_F1 = 0.9195)</i>
FBConvAI (Yang <i>et al.</i> 2020)	Nord was produced by Year of No Light and was signed to the record label Crucial Blast. It was followed by Live at Roadburn 2008. artist <i>(BLEU = 0.4793, BERTScore_F1 = 0.9221)</i>
BT5 (Agarwal <i>et al.</i> 2020)	The year of No Light is the producer of the album, Nord which was released on the record label Crucial Blast. The album was followed by Live at Roadburn 2008. artist <i>(BLEU = 0.7050, BERTScore_F1 = 0.9194)</i>
CGA-ET	The artist of Nord album is Year of No Light. It was produced by Year of No Light, released on the record label Crucial Blast and followed by Live at Roadburn 2008 <i>(BLEU = 0.7078, BERTScore_F1 = 0.9394)</i>
<b>MR</b>	Estádio_Municipal_Coaracy_da_Mata_Fonseca  location Arapiraca ..... & Agremiação_Sportiva_Arapiraquense  league Campeonato_Brasileiro_Série_C ..... & Campeonato_Brasileiro_Série_C  country Brazil ..... & Agremiação_Sportiva_Arapiraquense  nickname Alvinegro ..... & Agremiação_Sportiva_Arapiraquense  ground Estádio_Municipal_Coaracy_da_Mata_Fonseca
<b>Reference</b>	Estádio Municipal Coaracy da Mata Fonseca is the name of the ground of Agremiação Sportiva Arapiraquense in Arapiraca. Agremiação Sportiva Arapiraquense, nicknamed “Alvinegro”, lay in the Campeonato Brasileiro Série C league from Brazil
PlanEnc (Zhao <i>et al.</i> 2020)	Agremiação Sportiva Arapiraquense play in Brazil’s Campeonato Brasileiro Série C league and have the nickname Alvinegro. Their home ground is the Estádio Municipal Coaracy da Mata Fonseca in Arapiraca <i>(BLEU = 0.7311, BERTScore_F1 = 0.9243)</i>

Table 5. Continued.

OSU Neural NLG (Li <i>et al.</i> 2020)	Estádio Municipal Coaracy da Mata Fonseca is located in Arapiraca and is the home ground of Associação Sportiva Arapiraquense. They play in the Campeonato Brasileiro Série C league in Brazil and are nicknamed Alvinegro (BLEU = 0.7753, BERTScore_F1 = 0.9281)
FBConvAI (Yang <i>et al.</i> 2020)	Estádio Municipal Coaracy da Mata Fonseca is located in Arapiraca and is the ground of Associação Sportiva Arapiraquense who play in the Campeonato Brasileiro Série C league in Brazil. Their nickname is Alvinegro (BLEU = 0.7699, BERTScore_F1 = 0.9422)
BT5 (Agarwal <i>et al.</i> 2020)	Alvinegro is the nickname for Associação Sportiva Arapiraquense who play in the Campeonato Brasileiro Série C league in Brazil. Their ground is the Estádio Municipal Coaracy da Mata Fonseca which is located in Arapiraca (BLEU = 0.7667, BERTScore_F1 = 0.9224)
CGA-ET	Estádio Municipal Coaracy da Mata Fonseca is located in Arapiraca and is the ground of Associação Sportiva Arapiraquense. They play in the Campeonato Brasileiro Série C league in Brazil and thier nickname is Alvinegro (BLEU = 0.7665, BERTScore_F1 = 0.9465)

**Table 6.** Results on MultiWoz for generating sentence form input MR. The best and second-best models are highlighted in **bold** and underline face, respectively

Model	BLEU↑	SER↓
HDSA	26.5	12.14
SC-GPT2	30.8	<u>0.53</u>
Schema	33.3	1.89
T2G2	34.4	1.85
T5-small	34.6	1.27
T5-base	<u>35.1</u>	0.99
T5-large	34.7	0.92
T5-3B	34.8	0.86
CGA-ET	<b>37.5</b>	<b>0.41</b>

value of 0.915 was obtained indicating that the outputs of our proposed model are semantically close to the reference sentences. Since the outputs of the baseline models are not available, the reference sentences in the test set and the sentences generated by the CGA-ET model are used for subjective evaluation with results given in Table 7. In this dataset, some reference sentences in test data contain semantic errors and do not contain all concepts of the MR. For this reason, the proposed model has achieved higher coverage and faithfulness scores than the reference sentences. However, the score of fluency of the proposed model is lower compared to the reference sentences (about 0.07 units), which is an acceptable difference considering that the reference sentences are written by humans. Examples of input MRs and sentences generated by the CGA-ET model are given in Table 8.



**Table 7.** Results of human evaluations on MultiWoz dataset in terms of *Faithfulness*, *Coverage*, and *Fluency* (rating out of 3)

Model	Faithfulness (%)	Coverage (%)	Fluency
Reference	89.74	86.34	2.95
CGA-ET	93.17	90.21	2.89

**Table 8.** Comparison of the generated sentences from the MultiWoz dataset for our proposed model

<b>MR</b>	Booking-Book(Ref = 3FR3O8HO)
<b>Reference</b>	i have made your reservations and your reference number is 3fr3o8ho
CGA-ET	booking was successful. the table will be reserved for 15 minutes. reference number is: 3fr3o8ho (BLEU = 0.4062, BERTScore_F1 = 0.8999)
<b>MR</b>	Hotel-Inform(Price=cheap;Type=guesthouse)
<b>Reference</b>	i would like one of the cheap guesthouses please?
CGA-ET	i would like a guesthouses in the cheap price range (BLEU = 0.6839, BERTScore_F1 = 0.9160)
<b>MR</b>	Train-Inform(Depart=cambridge;Dest=ely;Leave = 18:15)
<b>Reference</b>	i am leaving from cambridge, and going to ely. i want to leave at 18:15
CGA-ET	i need a train from cambridge to ely, leaving after 18:15, please (BLEU = 0.5135, BERTScore_F1 = 0.9218)
<b>MR</b>	Attraction-Inform(Fee = 2.50 pounds;Addr=saint john's street)
<b>Reference</b>	the entrance fee is 2.50 pounds and the address is saint john's street
CGA-ET	it is located in saint john's street and its entrance fee is 2.50 pounds (BLEU = 0.7349, BERTScore_F1 = 0.9525)
<b>MR</b>	Restaurant-Inform(Food=european)
<b>Reference</b>	i think it would be great to try European food. Do you think they would accept a reservation for a party of ??
CGA-ET	i am looking for a restaurant serving european food (BLEU = 0.1757, BERTScore_F1 = 0.8784)

#### 4.5.3. Results on E2E

The results of the automatic metrics for the output of the CGA-ET and baseline models on the E2E dataset are given in Table 9. The baseline models used for comparison include the following:

- DATATUNER (Harkous *et al.* 2020), which first fine-tuned the GPT-2 pretrained language model on the E2E dataset and then by using the RoBERTa (Liu *et al.* 2019) model as a semantic fidelity classifier it detects and avoids generation errors.
- TGEN+ (Dusek *et al.* 2019), which is an LSTM-based sequence-to-sequence model with attention that uses slot matching script directly.

**Table 9.** Results on E2E. The best and second-best models are highlighted in **bold** and underline face, respectively

Model	BLEU↑	METEOR↑	NIST↑	ROUGE-L ↑	CIDEr↑
DATATUNER	43.60	39.00	–	57.50	2.000
TGEN+	40.51	37.61	6.1226	55.98	1.828
Schema	43.10	38.70	<u>6.4000</u>	56.80	1.900
T2G2	42.50	38.70	<u>6.4000</u>	56.90	1.900
+R	45.60	39.00	–	<u>65.70</u>	<u>2.678</u>
+RM	<u>46.10</u>	<u>39.80</u>	–	65.40	2.639
SQL	41.70	–	–	–	–
CGA-ET	<b>49.03</b>	<b>41.10</b>	<b>6.5040</b>	<b>65.82</b>	<b>2.690</b>

**Table 10.** Results of human evaluations on E2E dataset in terms of *Faithfulness*, *Coverage*, and *Fluency* (rating out of 3)

Model	Faithfulness (%)	Coverage (%)	Fluency
Reference	100.0	100.0	2.93
CGA-ET	98.09	97.83	2.87

- Schema and T2G2 (Kale and Rastogi 2020a) are two methods of generating text using a common model. In this model, the semantically correct but possibly incoherent and ungrammatical utterances are first generated for an input MR by using schema-guided or template-guided approaches. These utterances are then rewritten into natural sentences using T5.
- +R and +RM (Shen *et al.* 2020), which automatically extract the segmental structures of texts and learn to align them with their data correspondences. For reducing hallucination, they used constraints to prevent repetition (+R) and further reduce information missing (+RM).
- SQL (Guo *et al.* 2021), which is GPT-2 pretrained language model fine-tuned using soft Q-learning (Schulman, Chen, and Abbeel 2017)

The results in Table 10 show that the proposed model has outperformed all the baseline models in terms of the n-grams-based evaluation metrics. Also, we evaluated the output sentences of our model using BERTScore for which a value of 0.930 was obtained indicating that the outputs of our proposed model are semantically close to the reference sentences. As subjective evaluations, we compared the outputs generated by our CGA-ET model, with their gold reference, since the baseline outputs are not published. Table 10 shows the scores of each human evaluation factor for the E2E dataset. In the cleaned version of the E2E dataset, the semantic deficiencies that existed in previous versions have been fixed so that the concept of all input slots is expressed in the reference sentences. For this reason, the faithfulness and coverage scores for reference sentences are both 100%. The difference between the fluency scores of the CGA-ET model and the references is due to the fact that the reference sentences are written by humans. Examples of input MRs and sentences generated by the CGA-ET model are given in Table 11.

The loss curves of our CGA-ET model during the adversarial training process are shown in Fig. 6. As illustrated, the adversarial process between generator (Gen) and discriminator (Dis)

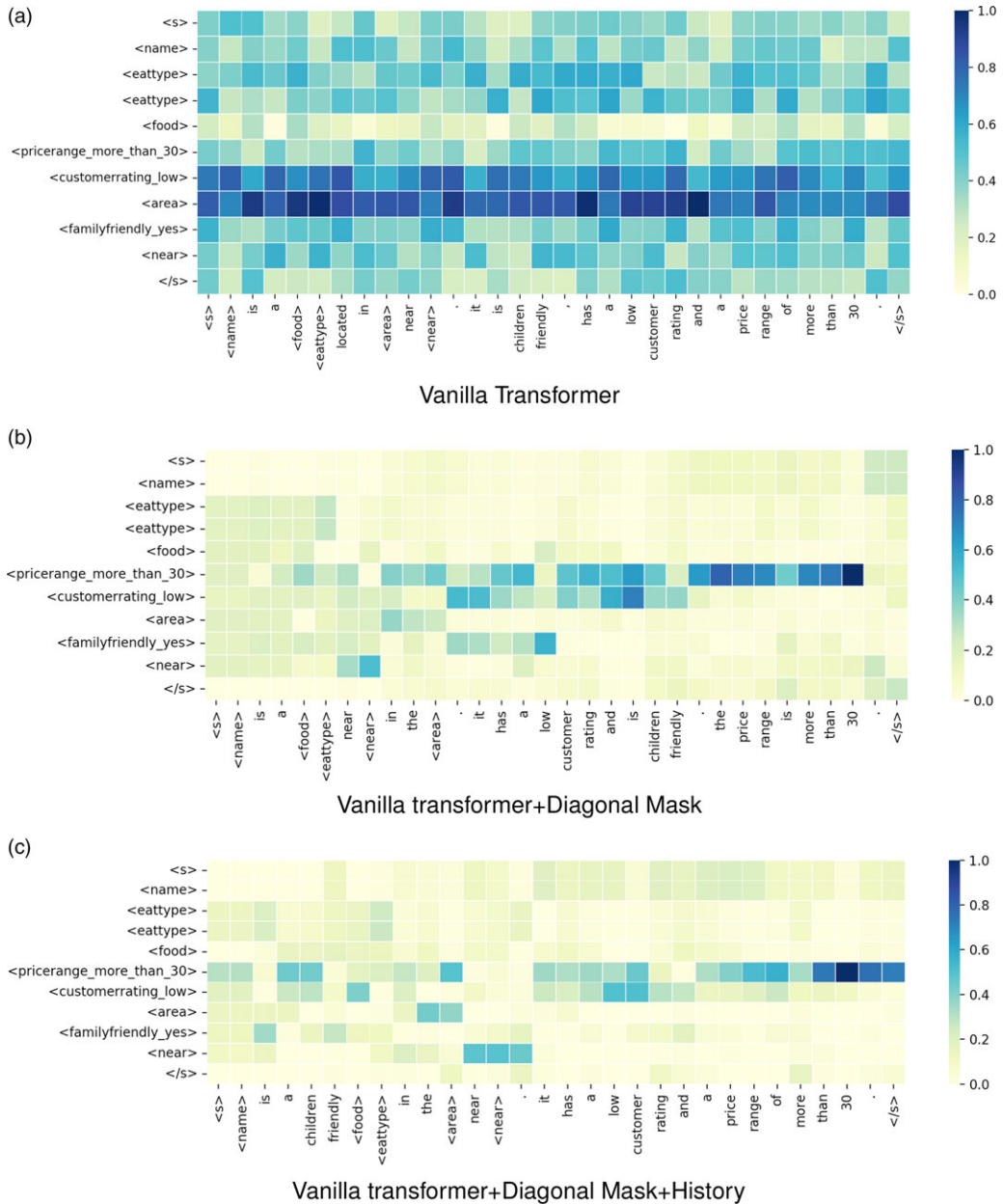
Table 11. Comparison of the generated sentences from the E2E dataset for our proposed model

<b>MR</b>	name[The Wrestlers], eatType[pub], food[Italian], priceRange[moderate], area[city centre], familyFriendly[no], near[Raja Indian Cuisine]
<b>Reference</b>	Located in the centre of the city, close to Raja Indian Cuisine, The Wrestlers is a pub offering Italian food
CGA-ET	The Wrestlers is a pub providing Italian food, it is located in the centre of the city near Raja Indian Cuisine (BLEU = 0.7878, BERTScore_F1 = 0.9547)
<b>MR</b>	name[The Wrestlers], eatType[restaurant], food[Japanese], priceRange[more than pounds 30], area[city centre], familyFriendly[yes], near[Raja Indian Cuisine]
<b>Reference</b>	Near Raja Indian Cuisine is a Japanese restaurant and restaurant named The Wrestlers. It has a price range of more than pounds 30 and is kid-friendly
CGA-ET	The Wrestlers is a Japanese restaurant in city centre near Raja Indian Cuisine. it is children friendly and has a price range of more than pounds 30 (BLEU = 0.7602, BERTScore_F1 = 0.9420)
<b>MR</b>	name[Zizzi], eatType[coffee shop], customer rating[high], near[Burger King]
<b>Reference</b>	Near Burger King is a highly rated coffee shop called Zizzi
CGA-ET	Zizzi is a coffee shop near Burger King with a high customer rating (BLEU = 0.6408, BERTScore_F1 = 0.9181)
<b>MR</b>	name[The Vaults], eatType[pub], food[Italian], priceRange[moderate], customer rating[1 out of 5], area[city centre], familyFriendly[no], near[Rainbow Vegetarian Café]
<b>Reference</b>	The Vaults, a moderately priced Italian pub in the centre of the city near Rainbow Vegetarian Café, has a low customer rating and does not welcome children
CGA-ET	There is an adult only pub called The Vaults which provides Italian food in the moderate prices. It is situated in the city centre near to the Rainbow Vegetarian Café with 1 out of 5 customer rating (BLEU = 0.5584, BERTScore_F1 = 0.9111)
<b>MR</b>	name[Clowns], eatType[coffee shop], customer rating[3 out of 5], near[All Bar One]
<b>Reference</b>	Near All Bar One there is a coffee shop named Clowns and it is rated 3 out of 5
CGA-ET	There is a coffee shop called Clowns near All Bar One with 3 out of 5 customer ratings (BLEU = 0.6859, BERTScore_F1 = 0.9382)

is quite stable. Initially, the value for the WGAN-GP loss function is high and prevents the error gradient for the fake sentence from being zero. During the training process, the amount of WGAN-GP is reduced, while triplet loss controls the semantic similarity of real and fake sentences with the input MR. Fluctuations in generator loss value indicate a productive attempt to mislead the discriminator under the control of MLE (similarity at the word level) and Distance (semantic similarity of sentences).

#### 4.6. Ablation study

As mentioned earlier, we made modifications to the vanilla transformer encoder and decoder to create an enhanced transformer model that is more compatible with the task D2T, and we used this enhanced model for the generator network of our CGA-ET model. To evaluate the effect of the modifications and compare them with the vanilla transformer, experiments are performed. In these experiments, the *vanilla transformer*, *vanilla transformer + Diagonal Mask*, and



**Figure 6.** Encoder–decoder attention weights for an E2E MR and the equivalent sentence generated by (a) Vanilla transformer, (b) Vanilla transformer+Diagonal Mask, and (c) Vanilla transformer+Diagonal Mask+History.

*vanilla transformer + Diagonal Mask + History* are trained separately on the training sets of E2E, WebNLG, and MultiWoz datasets using MLE. We also fine-tuned the GPT2 model on all three training datasets to compare it with the vanilla and enhanced transformer. For this purpose, we gave the sequence of pairs slots and values in the form `< slot_value >` to the GPT-2 and adjusted the weights of all 12 layers of it so that the equivalent sentence would be generated at the output. We also used beam search with a width of 5 and temperature 0.7 to generate the sentences in

**Table 12.** Comparison of the effect of modifications on the vanilla transformer in data-to-text generation

Dataset	Model	BERTScore	BERTScore	BERTScore
		Recall	Precision	F1
WebNLG	Vanilla Transformer	0.74	0.85	0.79
	Vanilla Transformer + Diagonal Mask	0.77	0.89	0.82
	Vanilla Transformer + Diagonal Mask + History	0.80	0.91	0.85
	Fine-Tuned GPT-2	0.70	0.84	0.76
MultiWoz	Vanilla Transformer	0.73	0.81	0.77
	Vanilla Transformer + Diagonal Mask	0.78	0.83	0.80
	Vanilla Transformer + Diagonal Mask + History	0.80	0.84	0.82
	Fine-Tuned GPT-2	0.72	0.79	0.75
E2E	Vanilla Transformer	0.75	0.81	0.78
	Vanilla Transformer + Diagonal Mask	0.79	0.82	0.80
	Vanilla Transformer + Diagonal Mask + History	0.81	0.84	0.82
	Fine-Tuned GPT-2	0.72	0.78	0.75

the inference phase. The results obtained on each test dataset are given in Table 12. As observed, when the vanilla transformer is trained and tested on a dataset, its generated output sentences got a higher value for F1 than when the pretrained models are used. Furthermore, as previously mentioned, the use of the diagonal mask in calculating the weights of attention in the encoder results in separate generation of input slots without considering the order of slots in the input MR. This makes the model less restricted in expressing slots in a particular order or in relation to other slots in equivalent sentences; so as a result, its F1 score is increased. Moreover, the use of history in calculating encoder–decoder attention weights causes the decoder to always pay attention to the sequence of slots previously expressed in the output when generating output tokens. Hence, semantic similarity with gold test sentences is increased. To illustrate the effect of using diagonal mask and history in output sentences, the encoder–decoder attention weights for an MR input from the E2E dataset and the generated sentences by enhanced and vanilla transformers are shown in Fig. 7.

#### 4.7. Case study

Examples of generated texts for a certain MR from the WebNLG, MultiWoz, and E2E datasets by CGA-ET and baseline models are shown in Tables 8, 9, and 10, respectively. The MR chosen from the WebNLG dataset has two similar objects and subjects but with two different predicates *artist* and *producer*, which is a challenge for the D2T models to express both predicates in their output sentences. As shown in Table 5, for the first example, unlike our proposed model, neither of the baseline models could express both predicates in the output. In the second example, all the input concepts are expressed in the sentences generated by baseline and our models. But the structures of the produced sentences are different, and this has caused the sentences to have different levels of fluency. Since the sentences generated by the baseline models are not published for the MultiWoz and E2E datasets, the outputs of the CGA-ET model for five MRs from these datasets are shown

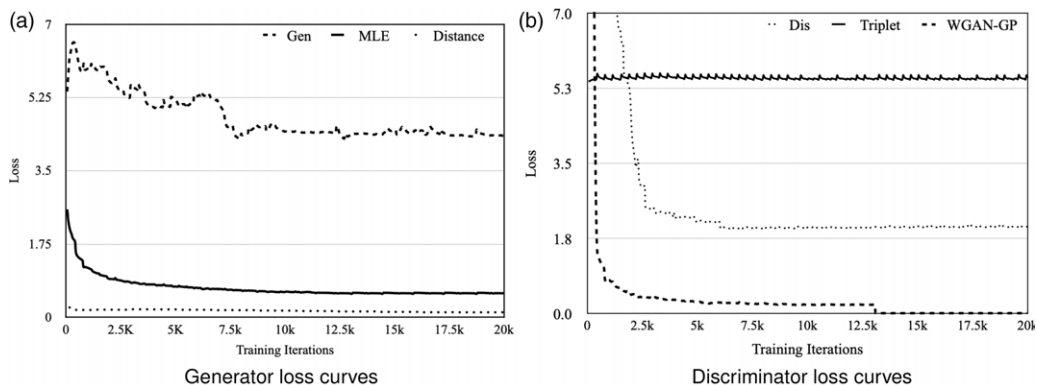


Figure 7. Different loss curves of CGA-ET model during adversarial training process.

in Tables 8 and 9. These sentences show that the proposed model is able to express all the required concepts in the input MR.

Apart from the faithfulness and coverage, the other differences between the sentences generated by our models, references, and baseline outputs are in their structure, syntax, and length. This diversity in the selection of phrases and words to express the input concepts is the effect of using an adversarial method for training the proposed model. Because, unlike the MLE-based baseline models, the generator in the proposed model does not generate sentences based on frequent patterns in the training data. As can be seen from the BLEU scores given in the tables obtained by comparing each sentence with the reference, this difference in the structure of the sentences generated by our proposed model led to getting a lower score when evaluated by the n-gram-based evaluation metrics, despite the semantic and structural correctness. But the BERTScore metric clearly shows the semantic similarity of the generated sentences to the reference sentence.

## 5. Conclusion

In this paper, we present a novel conditional generative adversarial method for D2T generation. The generator network in this model is our enhanced version of the vanilla transformer encoder and decoder. In this enhanced version, which is created to reduce semantic errors in the output sentences, a diagonal mask matrix is added to the self-attention computation step of the vanilla transformer encoder. Moreover, in the encoder–decoder multi-head cross-attention computation step of the vanilla transformer decoder, the history of attention scores for the previously generated tokens is considered. The discriminator network in the proposed model is a triplet network that consists of three vanilla transformer encoders with sharing weights. The loss function of the discriminator network includes WGAN-GP loss and triplet loss function between the real and generated output sentences and the input MR; as a result, while stabilizing the model training process, the semantic similarity between the generated sentence and the real sentence is maintained. Experimental results demonstrate that our proposed model can generate higher-quality sentences than transformer-based baseline models, in terms of BLEU, METEOR, NIST, ROUGE-L, CIDEr, BERTScore, and SER evaluation metrics.

## References

- Agarwal O., Kale M., Ge H., Shakeri S. and Al-Rfou R. (2020). *Machine translation aided bilingual data-to-text generation and semantic parsing*. In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web*, Dublin, Ireland, pp. 125–130.

- Bengio S., Vinyals O., Jaitly N. and Shazeer N. (2015). *Scheduled sampling for sequence prediction with recurrent neural networks*. In *Advances in Neural Information Processing Systems*, Quebec, Canada, pp. 1171–1179.
- Betti F., Ramponi G. and Piccardi M. (2020). *Controlled text generation with adversarial learning*. In *Proceedings of the 13th International Conference on Natural Language Generation*, Dublin, Ireland, pp. 29–34.
- Budzianowski P., Wen T.H., Tseng B., Casanueva I., Ultes S., Ramadan O. and Gasic M. (2018). *MultiWOZ: A large scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modeling*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, pp. 5016–5026.
- Castro-Ferreira T., Moussallem D., Krahmer E. and Wubben S. (2018). *Enriching the WebNLG corpus*. In *Proceedings of the 11th International Conference on Natural Language Generation*, Tilburg University, The Netherlands, pp. 171–176.
- Castro Ferreira T., Gardent C., Ilinykh N., Lee C., Mille S., Moussallem D. and Shimorina A. (2020). *The 2020 Bilingual, Bi-Directional WebNLG+ Shared Task: Overview and Evaluation Results (WebNLG+ 2020)*. arXiv: [2102.03556](https://arxiv.org/abs/2102.03556).
- Chang E., Shen X., Zhu D., Demberg V. and Su H. (2021). *Neural Data-to-Text Generation with LM-based Text Augmentation*. arXiv: [2102.03556](https://arxiv.org/abs/2102.03556).
- Che T., Li Y., Zhang R., Hjelm R.D., Li W., Song Y. and Bengio Y. (2017). *Maximum-likelihood augmented discrete generative adversarial networks*. arXiv: [1702.07983](https://arxiv.org/abs/1702.07983).
- Chelombiev I., Justus D., Orr D., Dietrich A., Gressmann F., Koliouisis A. and Luschi C. (2021). *Groupbert: Enhanced transformer architecture with efficient grouped structures*. arXiv: [2106.05822](https://arxiv.org/abs/2106.05822).
- Chen L., Dai S., Tao C., Shen D., Gan Z., Zhang H., Zhang Y. and Carin L. (2018). *Adversarial text generation via feature-mover's distance*. In *Advances in Neural Information Processing Systems*, Montréal, Canada, pp. 4666–4677.
- Chen W., Chen J., Qin P., Yan X. and Wang W.Y. (2019). *Semantically conditioned dialog response generation via hierarchical disentangled self-attention*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 3696–3709.
- Chen Z., Eavani H., Liu Y. and Wang W.Y. (2020). *Few-shot nlg with pre-trained language model*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 183–190, Online.
- Devlin J., Chang M., Lee K. and Toutanova K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, pp. 4171–4186.
- Dognin P., Padhi I., Melnyk I. and Das P. (2021). *ReGen: Reinforcement learning for text and knowledge base generation using pretrained language models*. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 1084–1099.
- Dusek O. and Jurcicek F. (2016). *Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings*. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, pp. 45–51.
- Dusek O., Howcroft D.M. and Rieser V. (2019). *Semantic noise matters for neural natural language generation*. In *Proceedings of the 12th International Conference on Natural Language Generation*, Tokyo, Japan, pp. 421–426.
- Fedus W., Goodfellow I. and Dai A.M. (2018). *MaskGAN: Better text generation via filling in the\_*. arXiv: [1801.07736](https://arxiv.org/abs/1801.07736).
- Gardent C., Shimorina A., Narayan S. and Perez-Beltrachini L. (2017). *Creating training corpora for NLG micro-planners*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, pp. 179–188.
- Gehrmann S., Dai F.Z. and Elder H. (2018). *End-to-End content and plan selection for data-to-text generation*. In *Proceedings of the 11th International Conference on Natural Language Generation*, Tilburg University, The Netherlands, pp. 45–56.
- Gkatzia D., Lemon O. and Rieser V. (2016). *Natural language generation enhances human decision-making with uncertain information*. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, pp. 264–268.
- Gong H. (2018). *Technical report for E2E NLG challenge*. In *E2E NLG Challenge System Descriptions*. Available at [http://www.macs.hw.ac.uk/InteractionLab/E2E/final\\_papers/E2E-Gong.pdf](http://www.macs.hw.ac.uk/InteractionLab/E2E/final_papers/E2E-Gong.pdf)
- Gong L., Crego J. and Snellart J. (2019). *Enhanced transformer model for data-to-text generation*. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, Hong Kong, pp. 184–156.
- Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A. and Bengio Y. (2014). *Generative adversarial nets*. In *Advances in Neural Information Processing Systems*. Quebec, Canada, pp. 2672–2680.
- Gou J., Lu H., Zhang W., Yu Y. and Wang J. (2017). *Long text generation via adversarial training with leaked information*. arXiv: [1709.08624](https://arxiv.org/abs/1709.08624).
- Guo H., Tan B., Liu Z., Xing E. and Hu Z. (2021). *Text generation with efficient (soft) q-learning*. arXiv: [2106.07704](https://arxiv.org/abs/2106.07704).
- Gulrajani I., Ahmed F., Arjovsky M., Dumoulin V. and Courville A.C. (2017). *Improved training of Wasserstein GANs*. In *Proceedings of the Advances in Neural Information Processing Systems*, Long Beach, CA, USA, pp. 5767–5777.
- Harkous H., Groves I. and Saffari A. (2020). *Have your text and use it too! End-to-end neural data-to-text generation with semantic fidelity*. In *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain, pp. 2410–2424, (Online).
- Hermans A.J., Beyer L. and Leibe B. (2017). *In defense of the triplet loss for person re-identification*. arXiv: [1703.07737](https://arxiv.org/abs/1703.07737).
- Hoffer E. and Ailon N. (2015). *Deep metric learning using triplet network*. In *International Workshop on Similarity-Based Pattern Recognition*, Copenhagen, Denmark, pp. 84–92.

- Jiao Z. and Ren F. (2021). WRGAN: Improvement of RelGAN with Wasserstein loss for text generation. *Electronics* 10(3), 275.
- Juraska J., Karagiannis P., Bowden K. and Walker M. (2018). A deep ensemble model with slot alignment for sequence-to-sequence natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, Louisiana, pp. 152–162.
- Juraska J. and Walker M. (2021). Attention Is indeed all you need: Semantically attention-guided decoding for data-to-text NLG. In *Proceedings of the 14th International Conference on Natural Language Generation*, Scotland, UK, pp. 416–431.
- Kale M. and Rastogi A. (2020a). Template guided text generation for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 6505–6520, Online.
- Kale M. and Rastogi A. (2020b). Text-to-text pre-training for data-to-text tasks. In *Proceedings of the 13th International Conference on Natural Language Generation*, Dublin, Ireland, pp. 97–102.
- Kasner Z. and Dušek O. (2020a). Data-to-text generation with iterative text editing. In *Proceedings of the 13th International Conference on Natural Language Generation*, Dublin, Ireland, pp. 60–67.
- Kasner Z. and Dušek O. (2020b). Train hard, finetune easy: Multilingual denoising for RDF-to-text generation. In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web*, Dublin, Ireland, pp. 171–176.
- Krippendorff K. (2011). Computing Krippendorff's alpha-reliability. Departmental Papers (ASC); University of Pennsylvania.
- Kusner M.J. and Hernández-Lobato J.M. (2016). GANS for sequences of discrete elements with the Gumbel-softmax distribution. arXiv: 1611.04051.
- Landis J.R. and Koch G.G. (1977). The measurement of observer agreement for categorical data. *Biometrics* 33(1), 159.
- Lavie A., Sagae K. and Jayaraman S. (2004). The significance of recall in automatic metrics for mt evaluation. In *Proceedings of the Machine Translation: From Real Users to Research*, Berlin, Germany, pp. 134–143.
- Ledig C., Theis L., Huszar F., Caballero J., Aitken A., Tejani A., Totz J., Wang Z. and Shi W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition*, Hawaii, USA, pp. 105–114.
- Lee J. (2021). Transforming Multi-Conditioned Generation from Meaning Representation. arXiv: 2101.04257.
- Lewis M., Liu Y., Goyal N., Ghazvininejad M., Mohamed A., Levy O., Stoyanov V. and Zettlemoyer L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880, Online.
- Li H., Zhu J., Zhang J. and Zong C. (2018). Ensure the correctness of the summary: In-corporate entailment knowledge into abstractive sentence summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, New Mexico, USA, pp. 1430–1441.
- Li X., Maskharashvili A., Stevens-Guille S.J. and White M. (2020). Leveraging large pretrained models for webnlg 2020. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web*, Dublin, Ireland, pp. 117–124.
- Li Y., Yao K., Qin L., Che W., Li X. and Liu T. (2020). Slot-consistent NLG for task-oriented dialogue systems with iterative rectification network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 97–106, Online.
- Lin C.W. (2004). Rouge: A package for automatic evaluation of summaries. In *Association for Computational Linguistics*, Barcelona, Spain, pp. 74–81.
- Lin K., Li D., He X., Zhang Z. and Sun M.T. (2017). Adversarial ranking for language generation. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, pp. 3155–3165.
- Liu T., Wang K., Sha L., Chang B. and Sui Z. (2018). Table-to-text generation by structure aware seq2seq learning. In *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, Hilton New Orleans Riverside, New Orleans, Louisiana, USA, pp. 4881–4888.
- Liu T., Zheng X., Chang B. and Sui Z. (2021). Towards faithfulness in open domain table-to-text generation from an entity-centric view. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 13415–13423.
- Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L. and Stoyanov V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv: 1907.11692.
- Martin A. and Przybocki M. (2000). The nist 1999 speaker recognition evaluation-an overview. *Digital Signal Processing* 10(1-3), 1–18.
- Mei H.Y., Bansal M. and Walter M. (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*, San Diego, California, pp. 720–730.
- Mille S., Dasiopoulou S. and Wanner L. (2019). A portable grammar-based NLG system for verbalization of structured data. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, New York, USA, pp. 1054–1056.
- Nayak N., Hakkani-Tur D., Walker M. and Heck L. (2017). To plan or not to plan? discourse planning in slot-value informed sequence to sequence models for language generation. In *Conference of the International Speech Communication Association*, Stockholm, Sweden, pp. 3339–3343.



- Nie W., Narodytska N. and Patel A. (2019). *Relgan: Relational generative adversarial networks for text generation*. In *Proceedings of the International Conference on Learning Representations*.
- Oraby O., Reed L., Tandon S., S. T.S., Lukin S. and Walker M. (2018). *Controlling personality-based stylistic variation with neural natural language generators*. In *Proceedings of the 19th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Melbourne, Australia, pp. 180–190.
- Panagiaris N., Hart E. and Gkatzia D. (2020). *Improving the naturalness and diversity of referring expression generation models using minimum risk training*. In *Proceedings of the 13th International Conference on Natural Language Generation*, Dublin, Ireland, pp. 41–51.
- Papineni K., Roukos S., Ward T. and Zhu W.J. (2002). *Bleu: A method for automatic evaluation of machine translation*. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, pp. 311–318.
- Peng B., Zhu C., Li C., Li X., Li J., Zeng M. and Gao J. (2020). Few-shot natural language generation for task-oriented dialog. arXiv: [2002.12328](https://arxiv.org/abs/2002.12328).
- Perez-Beltrachini L. and Lapata M. (2021). Multi-document summarization with determinantal point process attention. *Journal of Artificial Intelligence Research* 71, 371–399.
- Radford A., Wu J., Child R., Luan D., Amodei D. and Sutskever I. (2019). Language models are unsupervised multitask learners. *OpenAI* 1(8), 9, Technical report.
- Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W. and Liu P. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21, 1–67.
- Rashkin H., Reitter D., SinghTomar G. and Das D. (2021). *Increasing faithfulness in knowledge-grounded dialogue with controllable features*. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 704–718, Online.
- Rebuffel C., Roberti M., Soulier L., Scouthecten G., Cancelliere R. and Gallinari P. (2022). Controlling hallucinations at word level in data-to-text generation. *Data Mining and Knowledge Discovery* 36(1), 318–354.
- Reiter E. (2007). *An architecture for data-to-text systems*. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, Saarbrücken, Germany, pp. 97–104.
- Riou M., Jabaian B., Huet S. and Lefèvre F. (2017). *Online adaptation of an attention-based neural network for natural language generation*. In *Conference of the International Speech Communication Association*, Stockholm, Sweden, pp. 3344–3348.
- Schulman J., Chen X. and Abbeel P. (2017). Equivalence between policy gradients and soft Qlearning. arXiv: [1704.06440](https://arxiv.org/abs/1704.06440).
- Schuz S., Han T. and Zariess S. (2021). *Diversity as a by-product: Goal-oriented language generation leads to linguistic variation*. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Singapore, pp. 411–422.
- See A., Liu J.P. and Manning C.D. (2017). *Get to the point: Summarization with pointer-generator networks*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, pp. 1073–1083.
- Sellam T., Das D. and Parikh A. (2020). *BLEURT: Learning robust metrics for text generation*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7881–7892, Online.
- Sha L., Mou L., Liu T., Poupard P., Li S., Chang B. and Sui Z. (2018). *Order-planning neural text generation from structured data*. In *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, Hilton New Orleans Riverside, New Orleans, Louisiana, USA, pp. 5414–5421.
- Shi Z., Chen X., Qiu X. and Huang X. (2018). *Toward diverse text generation with inverse reinforcement learning*. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, pp. 4361–4367.
- Shen X., Chang E., Su H., Zhou J. and Klakow D. (2020). *Neural data-to-text generation via jointly learning the segmentation and correspondence*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7155–7165, Online.
- Song K., Wang K., Yu H., Zhang Y., Huang Z., Luo W., Duan X. and Zhang M. (2020). *Alignment-enhanced transformer for constraining NMT with pre-specified translations*. In *Proceedings of the The 44th AAAI Conference on Artificial Intelligence*, New York, USA, pp. 8886–8893.
- Su J., Lu Y., Pan S., Wen B. and Liu Y. (2021) *Roformer: Enhanced transformer with rotary position embedding*. arXiv: [2104.09864](https://arxiv.org/abs/2104.09864).
- Su Y., Vandyke D., Wang S., Fang Y. and Collier N. (2021). *Plan-then-generate: Controlled data-to-text generation via planning*. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, pp. 895–909.
- Tian R., Narayan S., Sellam T. and Parikh A.P. (2019). *Sticking to the Facts: Confident Decoding for Faithful Data-to-Text Generation*. arXiv: [1910.08684](https://arxiv.org/abs/1910.08684).
- Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L. and Polosukhin I. (2017). *Attention is all you need*. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, New York, USA, pp. 6000–6010.

- Vedantam R., Zitnick L. and Parikh D.** (2015). *CIDEr: Consensus-based image description evaluation*. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, pp. 4566–4575.
- Wang K. and Wan X.** (2018). *SentiGAN: Generating sentimental texts via mixture adversarial networks*. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, pp. 4446–4452.
- Wang L., Schwing A. and Lazebnik S.** (2017). *Diverse and accurate image description using a variational auto-encoder with an additive Gaussian encoding space*. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, New York, USA, pp. 5758–5768.
- Wang Z., Wang X., An B., Yu D. and Chen C.** (2020). *Towards faithful neural table-to-text generation with content-matching constraints*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1072–1086, Online.
- Wen T.H., Gasic M., Kim D., Mrksic N., Su P.H., Vandyke D. and Young S.** (2015a). *Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking*. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Prague, Czech Republic, pp. 275–284.
- Wen T.H., Gasic M., Mrksic N., Rojas Barahona L.M., Su P.H., Vandyke D. and Young S.** (2015b). *Toward multi-domain language generation using recurrent neural networks*. In *NIPS Workshop on Machine Learning for Spoken Language Understanding and Interaction*, Hanoi, Vietnam.
- Wen T.H., Gasic M., Mrksic N., Su P.H., Vandyke D. and Young S.** (2015c). *Semantically conditioned LSTM-based natural language generation for spoken dialogue systems*. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, pp. 1711–1721.
- Wen T.H., Gasic M., Mrksic N., Rojas Barahona L.M., Su P.H., Vandyke D. and Young S.** (2016). *Multi-domain neural network language generation for spoken dialogue systems*. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California, pp. 120–129.
- Williams R.** (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8(3-4), 229–256.
- Xu J., Ren X., Lin J. and Sun X.** (2018). *Diversity-Promoting GAN: A cross-entropy based generative adversarial network for diversified text generation*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, pp. 3940–3949.
- Yang Z., Einolghozati A., Inan H., Fan A., Donmez P. and Gupta S.** (2020). *Improving text-to-text pre-trained models for the graph-to-text task*. In *Proceedings of the 3rd WebNLG Workshop on Natural Language Generation from the Semantic Web*, Dublin, Ireland, pp. 107–116.
- Yu L., Zhang W., Wang J. and Yu Y.** (2017). *SeqGAN: Sequence generative adversarial nets with policy gradient*. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, California, USA, pp. 2852–2858.
- Zhang T., Kishore V., Wu F., Weinberger K.Q. and Artzi Y.** (2020). *Bertscore: Evaluating text generation with bert*. In *International Conference on Learning Representations*.
- Zhang Y., Gan Z., Fan K., Chen Z., Heno R., Shen D. and Carin L.** (2017). *Adversarial feature matching for text generation*. In *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, pp. 4006–4015.
- Zhang Y., Sun S., Galley M., Chen Y., Brockett C., Gao X., Gao J., Liu L. and Dolan B.** (2020). *DIALOGPT: Large scale generative pre-training for conversational response generation*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 270–278, Online.
- Zhao C., Walker M. and Chaturvedi S.** (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*, pp. 2481–2491, Online.