Towards Diversity and Relevance in Neural Natural Language Response Generation

Master's Thesis of

Daniel Handloser

at the Interactive Systems Lab
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT)

Reviewer:	Prof. Dr. Alexander Waibel
Second reviewer:	Prof. Dr. Tamim Asfour
Advisor:	Stefan Constantin
Second advisor:	Prof. Dr. Tetsunori Kobayashi
Third advisor:	Dr. Yoichi Matsuyama

12. August 2019 – 11. February 2020

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

Adam Adaptive Moment Estimation

AIM Adversarial Information Maximization

ASR Automatic Speech Recognition

BERT Bidirectional Encoder Representations from Transformers

BLEU Bilingual Evaluation Understudy

BPE Byte Pair Encoding

CMU DOG CMU Document-grounded conversation

CNN Convolutional Neural Network

DAAD German Academic Exchange Service

DPG Deterministic Policy Gradient

DSTC7 7th Dialog System Technology Challenges

GAN Generative Adversarial Network

GB Gigabyte

GPT-2 Generative Pretrained Transformer 2

GPU Graphics Processing Unit

GRU Gated Recurrent Unit

HMM Hidden Markov Model

LSTM Long Short-Term Memory

MASS Masked Sequence to Sequence Pre-training for Language Generation

METEOR Metric for Evaluation of Translation with Explicit ORdering

MI Mutual information

MNLI Multi-Genre Natural Language Inference

RNN Recurrent Neural Network

Rouge Recall-Oriented Understudy for Gisting Evaluation

S2S Sequence to Sequence

SGD Stochastic Gradient Descent

SQuAD Stanford Question Answering Dataset

VIM Variational Information Maximization

WMT16 2016 Conference on Machine Translation

I declare that I have developed and written the enclosed thesis completely by myself, and have not used sources or means without declaration in the text.

Karlsruhe, 11. February 2020

. . . . . .

(Daniel Handloser)

Abstract

In Neural Response Generation a system produces responses in a chit-chat dialogue covering a wide variety of topics. While the research community has long been focused on task-specific systems, these open-domain queries have become an active area of research thanks to advancements in deep learning and the availability of large amounts of conversational data.

Early models were able to generate responses that are semantically valid responses with respect to the query, however their outputs are often short, general, and provide little information [81].

This work compares existing architectures that tackle these problems and try to generate more diverse and informative dialogues. We train and evaluate existing models on our *Reddit* corpus.

Two notable works for diversity and informativeness by Zhang et al., 2017 [89] and Gao et al., 2019 [26] are based on an adversarial and a recurrent multi-tasking approach, respectively. The two works are evaluated and compared to a baseline. The latter showed the most promising results.

While Gao et al. outperform existing works in the diversity of their outputs recurrent neural networks in general have difficulties when dealing with longer sequences of text [90]. To further increase diversity and informativeness we propose two novel approaches. We add a hierarchical encoder structure to Gao et al., 2019 in order to capture more information in multi-turn dialogues. Secondly we build a *Transformer* with the same multi-task setting, and leverage pre-training. The model shall both improve diversity and informativeness by leveraging Gao et al.'s approach, the non-recurrent architecture, the additional amount of parameters, and the high-volume of pre-training data (compared to the size of our dataset).

Both solutions outperform the other evaluated models in Bilingual Evaluation Understudy (BLEU), Metric for Evaluation of Translation with Explicit ORdering (METEOR), and Recall-Oriented Understudy for Gisting Evaluation (Rouge), which are commonly used to evaluate response generation tasks.

Zusammenfassung

Die Masterarbeit nimmt sich dem Generieren von Antworten auf Dialoge mit neuronalen Netzen an. Traditionell hat sich die Forschung in diesem Bereich hauptsächlich mit Systemen beschäftigt, die eine bestimme Aufgabe (task-specific) erledigen – wie beispielsweise das Buchen eines Tisches in einem Restaurant. Dank immer größerer Datensätze und Fortschritten im Bereich neuroanle Netze befassen sich aktuell mehrere Arbeiten mit domänenübergreifenden Dialogen (open-domain), in denen die Teilnehmer nicht an ein bestimmtes Thema gebunden sind.

Die ersten Modelle in diesem Bereich erzeugen semantisch korrekte Antworten, haben jedoch oft das Problem, dass diese sehr generisch sind und wenig nützliche Informationen enthalten.

In dieser Arbeit werden verschiedene Architekturen behandelt, die dafür sorgen, dass Antworten eine größere Vielfalt und mehr Relevanz bieten. Dafür wird ein Datensatz mit Dialogen von der Internetplattform *Reddit* erstellt. Auf diesen Texten werden neuronale Netze trainiert und evaluiert.

Zwei wichtige Arbeiten für mehr Vielfalt und Relevanz in Dialogsystemen stammen von Zhang et al., 2017 [89] und Gao et al., 2019 [26]. Letztere erzielt die besten Ergebnisse auf dem vorgestellten Datensatz.

Weiter werden zwei neue Ansätze vorgestellt. Zum einen wird die Idee von strukturierten latenten Räumen von Gao et al. mit einem Ansatz kombiniert, der Zusammenhänge in Eingaben zuverlässiger erkennt. Letzteres kann bei längeren Texten in rekurrenten neuronalen Netzen zu Problemen führen [69].

Im zweiten Ansatz werden strukturierte latente Räume auf die *Transformer*-Architektur [76] angewandt.

Bei der Evaluation basierend auf den Metriken BLEU, METEOR und Rouge – welche häufig in ähnlichen Arbeiten verwendet werden [89, 26, 90] – erzielen beide neuen Ansätze bessere Ergebnisse als die bestehenden.

Acknowledgments

First and foremost I would like to thank Prof. Alex Waibel, Prof. Tetsunori Kobayashi, and Margit Rödder from the interACT program for giving me the opportunity to work on this thesis at Waseda University in Tokyo, Japan. Further I would like to thank my advisors Stefan Constantin and Yoichi Matsuyama for giving me the freedom of working on a topic of my choice. I am especially thankful for the funding that was provided by the German Academic Exchange Service (DAAD).

Additionally I am very grateful for the people that were around me during the process of writing this thesis. This includes all the students from the lab at Waseda University as well as my friends and family at home who have been incredibly supportive.

Contents

Abstract i

Zusammenfassung iii

1. Introduction 1
- 1.1. Motivation . . . 1
- 1.2. Outline . . . 1

2. Fundamentals 3
- 2.1. Deep Learning . . . 3
  - 2.1.1. Sequence to Sequence Learning . . . 3
  - 2.1.2. Transformer . . . 4
  - 2.1.3. Generative Adversarial Networks . . . 6
  - 2.1.4. Autoencoder . . . 7
- 2.2. Dialogue Systems . . . 9
  - 2.2.1. Speech Dialogue Systems . . . 10
  - 2.2.2. Open-Domain Dialogue Systems . . . 10
- 2.3. Metrics . . . 11
  - 2.3.1. BLEU . . . 11
  - 2.3.2. Distinct . . . 11
  - 2.3.3. Entropy . . . 12
  - 2.3.4. METEOR . . . 12
  - 2.3.5. Rouge . . . 13

3. Related Work 15
- 3.1. Models . . . 15
  - 3.1.1. Large-Scale Pre-Training . . . 16
- 3.2. Datasets . . . 18
- 3.3. Metrics . . . 20

4. Models for Diversity and Relevance 21
- 4.1. Datasets . . . 21
- 4.2. Baseline . . . 23
- 4.3. Adversarial Approach . . . 24
  - 4.3.1. Adversarial Information Maximization . . . 25
- 4.4. Geometrical Approach . . . 28
- 4.5. Recurrent Hierarchical Approach . . . 31
  - 4.5.1. Training . . . 31

4.6. Transformer Approach 32
4.6.1. Training 32
4.7. Geometrical Transformer Approach 33

5. Evaluation **35**
5.1. Metrics 35
5.2. Baseline 38
5.3. Adversarial Approach 38
5.3.1. Training 40
5.4. Geometrical Approach 40
5.5. Geometrical Hierarchical Approach 41
5.6. Transformer Approach 41
5.7. Geometrical Transformer Approach 43
5.8. Comparison 44
5.9. Metric Evaluation 46

6. Conclusion **49**
6.1. Discussion 49
6.2. Future Work 49

Bibliography **51**

A. Appendix **59**
A.1. Recurrent Hierarchical Architecture 59

1. Introduction

1.1. Motivation

The majority of human conversation is based on "socialization, personal interests and chit-chat" [22]. Naaman et al., 2010 found that just under 5 % of posts on the social-media platform Twitter are questions, but 80 % of tweets revolve around "emotional state, thoughts or activities" [55]. Thus the potential training data allows for a variety of deep learning tasks revolved around such chit-chat conversations. However, these dialogue systems often perform poorly and tend to be hard to evaluate [88]. Therefore current approaches are mainly based on task-specific communication, such as question answering or booking a table at a restaurant.

The existing solutions for open-domain response generation often produce short and generic responses, such as "I agree", or "That is not true", which are bland and uninformative. To tackle this issue this work explicitly focuses on controlling informativeness and diversity for general conversations, which is called an open-domain dialogue.

1.2. Outline

This thesis explores different methods to generate diverse and informative responses to social dialogues. Chapter *2 Fundamentals* contains an introduction to deep learning, which is the technical basis of this work. This is followed by an introduction to *Dialogue Systems (2.2)* and an overview of *Metrics (2.3)* used to evaluate them.

Chapter *3 Related Work* presents different works in dialogue generation. The aim of this chapter is to give an overview of recent works in dialogue system that focus on the open-domain task.

In *4 Models for Diversity and Relevance* the technologies relevant for our work are discussed in detail. Based on these technologies we create two novel approaches that combine different architectures. This part also gives an overview of the training setup and data that we use to solve the problem of diversity and relevance in dialogue systems.

The above solutions are then compared in *5 Evaluation*. Strengths and weaknesses of the models are examined. We further asses the metrics themselves and compare their results to a human evaluation.

The final chapter *6 Conclusion* discusses the finding of this work and gives an outlook on potential future work.

2. Fundamentals

This chapter gives a theoretical introduction to the topics relevant for this thesis. *2.1 Deep Learning* covers technologies that are relevant for this work. In *2.2 Dialogue Systems* the different paradigms of human-machine conversation are compared and the final section *2.3* gives an overview of metrics that are often used to judge a dialogue system.

2.1. Deep Learning

Deep learning is the field of study concerned with neural networks that "learn [...] multiple levels of representation in order to model complex relationships among data" based on (artificial) neural networks [19]. While the idea of an artificial neural network has been introduced in the middle of the 20th century [52, 66], very deep networks have only become feasible in terms of computation with advancements in Graphics Processing Unit (GPU)s in recent years [78].

2.1.1. Sequence to Sequence Learning

Sequence to Sequence (S2S) learning converts sequences from one domain to sequences in another domain. Besides response generation, common objectives are machine translation or summarization tasks [71].

Given a source $x = (x_1, x_2, \ldots, x_i)$ and a target sentence $y = (y_1, y_2, \ldots, y_j)$, let $(x, y) \in (\mathbb{X} \times \mathbb{Y})$ be corresponding sequence pairs, where $\mathbb{X}$ und $\mathbb{Y}$ are the source and target domains. Formally a S2S model learns parameters $\theta \in \mathbb{R}^n$ to estimate $P(y|x;\theta)$. The objective is often formulated with log-likelihood:

$$L(\theta; (\mathbb{X} \times \mathbb{Y})) = \sum_{(x,y)\in(\mathbb{X}\times\mathbb{Y})} \log P(y|x;\theta).$$

The encoder-decoder is a framework for S2S learning, where the encoder creates a hidden representation of the source sequence and the decoder generates the target sequence based on the representation [71]. The most common architectures are based on Long Short-Term Memory (LSTM) [36] or Convolutional Neural Network (CNN) [73, 27].

The following subsection introduces the *Attention* mechanism, that is relevant for the models introduced later in this work.

2.1.1.1. Attention

The attention model is a modification to S2S models.
Until 2017, deep learning for S2S problems such as machine translation, speech recognition,

or response generation, was often based on Recurrent Neural Network (RNN)s [33, 5, 65]. The performance of these networks has been pushed by the introduction of the attention mechanism. This algorithm has been proposed by Bahdanau et al., 2014 for machine translation [5] and has later been applied to other language tasks [48, 56].
This section examines some problems of RNNs, and explains how attention can tackle these. The next section then introduces a feed-forward architecture based on attention called the *Transformer*.

When transforming one sequence into another sequence, recognizing dependencies between tokens in a sequence is crucial, because in recurrent networks a sequence of words is created word by word. It is a problem for a RNN to detect dependencies if there is a long range of tokens inbetween two relevant ones.

The LSTM attempts to solve the shortcomings of the standard RNN, but still the chance of the required information being stored in the LSTM's memory decreases exponentially with the absolute distance between sequence token indexes [11]. Hence large dependencies remain a problem when only using the last hidden state as context vector.

With the attention mechanism a context vector is created for every token in the input sequence. Hence we create n context vectors for a source sequence $s_1, s_2 \ldots s_n$ with length n. In order to calculate the context vector we compute a attention score α:

$$\alpha_{ij} = \frac{e^{e_{ij}}}{\sum_{k=1}^{T_x} e^{e_{ik}}},$$

$$e_{ij} = a(s_{i-1}, h_j),$$

where the score e_{ij} determines a weight for position j in the input with regards to position i in the output.

The context vector c_i is then calculated as follows:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j,$$

where h_j is the hidden representation of the j-th token in the source sequence.

A RNN using attention can leverage all hidden token representations, but can only process inputs sequentially. To allow for parallel inputs, the *Transformer* combines CNNs with attention, since a convolutional layer can process the entire input sequence in parallel.

2.1.2. Transformer

Vaswani et al., 2017 introduced the idea of moving away from RNNs with their non-recurrent *Transformer* model [76].

The *Transformer* they propose consists of 6 encoder and decoder layers respectively. In the encoder each layer consists of a self-attention and a feed-forward part. Every encoder features a residual connection followed by normalization [76], see *Figure 2.1*.
The decoder has a similar setup to the encoder, consisting of 6 layers that all contain a

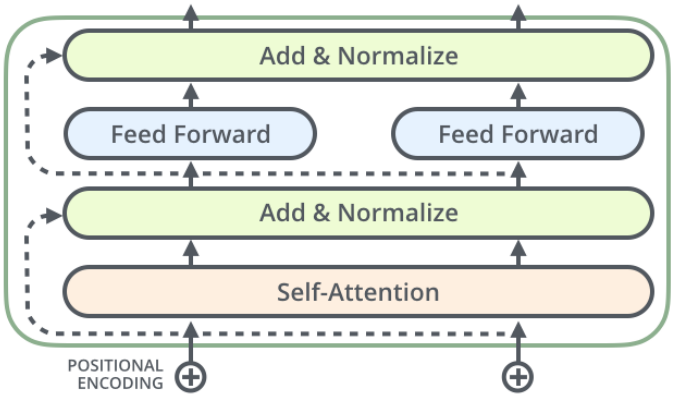

Figure 2.1.: Transformer Encoder layer [1]

self-attention and a feed-forward layer. Between these two the decoder also contains a "encoder-decoder attention" layer.

After the encoder has processed the input sequence, the output is transformed into attention vectors K and V. These vectors are then passed to all six decoder layers, more specifically to the "encoder-decoder attention" layers. Based on the previous output token the decoder generates one token at a time.

The following sections provide an insight into the self-attention mechanism used by the Transformer.

2.1.2.1. Self-Attention

The concept of self-attention allows the representation of a token to not only be based on itself, but rather incorporate further context from other tokens in the same sequence.

For each token the attention algorithm takes the input of the encoder and produces three vectors for every token: query q, keys k with dimension d and value v. These vectors are calculated by multiplying the embedding of the input token with three matrices. The weights of each matrix are learned during training.
For the entire sequence these vectors are stored in matrices Q, K, and V. The outputs are calculated as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V,$$

where *softmax* leads to a probability distribution determining how much attention should be given to each token [76].

2.1.2.2. Multi-Headed Attention

Vaswani et al. found that the Transformer best detects dependencies in the input sequence when employing multiple attention functions instead of just one [76].

In their initial work they compute 8 *Attention* matrices based on different Q, K, and V, which are then concatenated and multiplied with a weight matrix W:

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_8)W$$

$$head_i = Atttention(QW_i^Q, KW_i^K, VW_i^V),$$

where weight matrices $W_i^Q, W_i^K \in \mathbb{R}^{d_{model} \times d_k}$ and $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$, $W^0 \in \mathbb{R}^{8d_v \times d_{model}}$, and $d_{model}/8 = 64$.

The self-attention layer of the Transformer net features 8 attention heads.

2.1.2.3. Bidirectional Transformer

Bidirectional Encoder Representations from Transformers (BERT) is a pre-training technique introduced by Devlin et al., 2018 that archived state-of-the-art results in question answering on the Stanford Question Answering Dataset (SQuAD) v1.1, and in natural language inference on the Multi-Genre Natural Language Inference (MNLI) corpus [20]. Devlin et al. propose a bidirectional training process for the Transformer model, whereas previous works processed text sequences from left to right or employed a combination left-to-right and right-to-left training [20]. In training from left to right (forward) only the information of the previous tokens are accessible to predict the next token and when training from right to left (backward) only the following tokens in the sequence are available for the prediction of the current token. In bidirectional training however the information of the entire sentence can be used to predict each token.

2.1.3. Generative Adversarial Networks

The Generative Adversarial Network (GAN) has been introduces by Goodfellow et al., 2014 [31]. While the adversarial training process has shown great success for image generation, convergence issues and difficulties dealing with discrete data make the application of GANs to the text domain challenging.

The following introduces GANs, as well as their applications to text and discusses the reasons behind the problems of GANs with discrete data.

2.1.3.1. Real-Valued Data

GANs are an assumption-free method to estimate distributions and are therefore generative models. The unsupervised machine learning technique is implemented by two neural networks competing in a zero-sum game. They were introduced by Goodfellow et al., 2014 [30].

The adversarial process trains two models: a generative model G and a discriminative model D. The goal of D is to estimate whether a given sample of data comes from the training set or was created by G. The goal of G is to create new samples that will eventually be categorized as training samples by D, i.e. maximize the errors produced by D. In other words, the two networks play a minmax game with the value function $V(D, G)$ against each other:

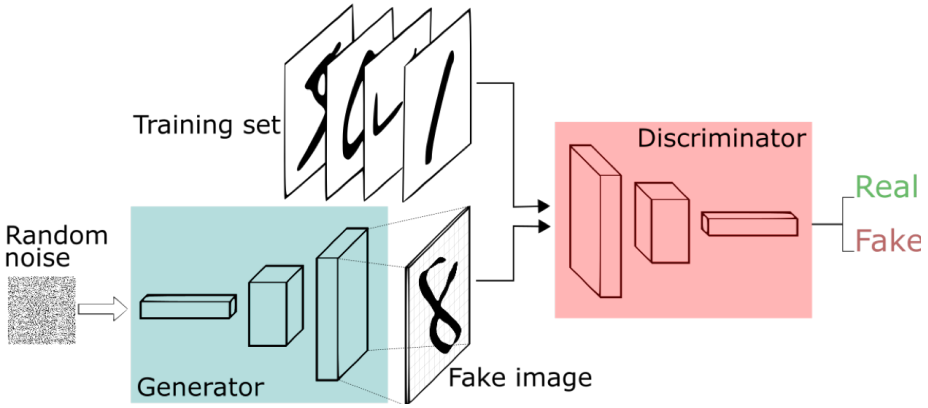

Figure 2.2.: Illustration of Generator G and Discriminator D in a GAN for images. [28]

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))],$$

where x is the input data, and $p_z(z)$ is defined as a prior on the input noise variables.

D and G are generally modelled as multi-layer perceptrons, and thus can be trained by backpropagation [30].

The training process of GANs is sensitive and total collapse of the model in training is a frequently encountered problem [67]. Therefore stabilizing the training process is a major part of current research in generative image modelling [87, 67].

2.1.3.2. Discrete Data

In image generation a generator G outputs a matrix of real values that represents an image. This matrix is fed to discriminator D and classified as real or fake. This behaviour can not be directly applied for language. In a world-level RNN language model for every time step t the input consists of two parts: the previous hidden state and the previous output. Based on those, the new hidden state is being generated. The next word is chosen by argmax. Thus for each time step a new word is generated. In back-propagation training cross-entropy loss is used to compare the output of the softmax layer to the actual one-hot encoding of the training sample [41].

Using this RNN language model for G in a GAN for response generation we no longer minimize the cross-entropy loss function from the RNN, but rather the training objective is to make D classify the sample as correct. However choosing the next word for every time step t is not differentiable [89], therefore it is not possible to backpropagate gradients through the model's outputs. There are multiple approaches to work around this problem. One is described in *4.3 Adversarial Approach*.

2.1.4. Autoencoder

The last part of the *Deep Learning* section focuses on the *Autoencoder*, which is part of two models in the main chapter of this work.

An Autoencoder is an unsupervised neural network that consists of an encoder and a decoder. The encoder encodes the input data in a lower dimensional space in order to reduce noise. The features generated from the input are then passed to the decoder, which aims to reconstruct the input data [29].

Parameters are trained using back-propagation to minimize reconstruction loss, which measures the difference between input and output.

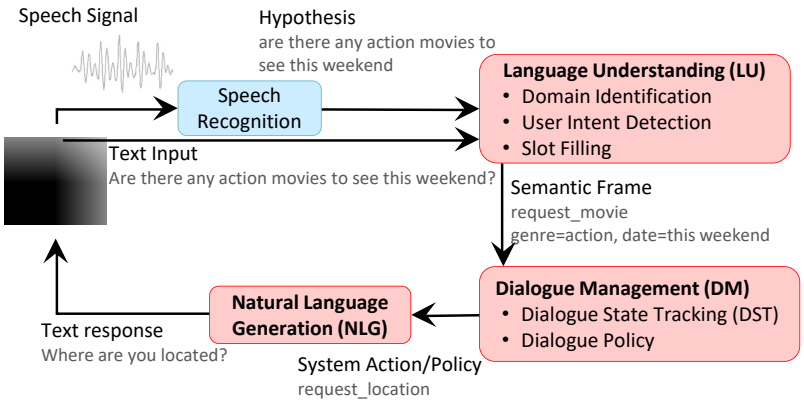

Figure 2.3.: Components of a speech dialogue system. [51]

2.2. Dialogue Systems

Human-machine conversation is one of the core problems in Natural Language Processing. Systems that solve this problem are called dialogue systems or conversational AI. The input and output of a dialogue system can be text, speech, graphics, haptics, gestures [79], or a combination of these.

The technology's popularity increased with the rise of virtual personal assistants, such as Siri, Google Now, and Amazon Echo throughout the 2010s [57]. According to *statista* 1.8 billion people will use virtual assistants by 2021 worldwide [57]. Therefore research has not only been carried out at universities but to a large extend at technology companies, such as Google, Amazon, or Apple [23].

Dialogue systems can be categorized into two paradigms: open-domain dialogue systems and task-based dialogue systems.
Traditionally research focused on task-oriented dialogue systems [83]. These solve Human-machine conversation in a specific scenario, for example ordering movie tickets or reserving a table in a restaurant.

Systems that do not only solve a single user scenario are called open domain dialogue systems, and have made some advances in the last few years, see Shum et al., 2018 [70] and Radford et al., 2019 [62].

Both systems are typically limited in the number of dialogue turns, meaning that there is a fixed amount of possible back-and-fourth turns between the dialogue system and the user.

The area of research that uses neural networks to generate textual responses is sometimes referred to as neural response generation [82, 81].

The technical foundation of dialogue systems has transformed from statistical methods being used until the beginning of this century from where on deep learning methods have started to make end-to-end systems possible.

2.2.1. Speech Dialogue Systems

Speech dialogue systems, sometimes referred to as conversational agents or spoken language systems, have several applications such as call routing and virtual personal assistants like Apple's Siri. This section aims to give a more complete overview of dialogue systems, and therefore covers more than the neural response generation systems that will be discussed later in this work.

The traditional structure of such a system can be seen in *Figure 2.3*. The central component is the dialogue manager, that receives the processed user input and keeps track of the state of the dialogue. Additionally it communicates with the task manager and sends outputs to the components that generate the synthesized response to the input. This component can be modelled as a Markov Decision Process, a finite state, or a neural network [51]. The component for natural language understanding transforms the input to a structured representation of the meaning of a sentence. This was traditionally often done by a Hidden Markov Model (HMM) [51]. The speech recognition model architecture is based on HMM or a neural network, and processes the speech input. A content planner in the language generation block is responsible for the decision on what content is returned to the user. Based on that, the language generation block generates synthesized speech [51].

Dialogue systems today can often be end-to-end systems that do not follow the architecture of *Figure 2.3*, see *3 Related Work*.

2.2.2. Open-Domain Dialogue Systems

Open-domain dialogue system, as is covered in this work, do not only solve one specific task, therefore they are sometimes referred to as non-goal-driven.

Open-domain publications often train on posts of social media platforms, thanks to the large amount of data accessible online [69, 63]. Ritter et al., 2011 were the first to introduce a generative probabilistic model for conversations on social-media posts [69]. They formulated the problem as statistical machine translation, where one post is translated into its response. The work notes that this would be "more difficult than translating between languages, due to the wider range of possible responses, the larger fraction of unaligned words/phrases, and the presence of large phrase pairs whose alignment cannot be further decomposed." [63]

2.3. Metrics

The following introduces metrics to evaluate neural response models. Most of them are derived from other natural language processing tasks.

2.3.1. BLEU

BLEU [58] is a metric proposed for evaluating the results of machine translations. The generated translation is referred to as the candidate, and is being compared to one or more translations, called references.

Apart from translation, BLEU is used as metric for a variety of natural language processing tasks, including response generation [47]. In response generation the reference is one or more target sentences that respond to the same source sentence. The synthetically generated response is called the candidate or hypothesis.

BLEU evaluates based on word-level n-grams. An n-gram in this case is every combination of n adjacent words. The sentence "I like Tokyo" for example contains the following bigrams (2-grams): "# I", "I like", "like Tokyo", and "Tokyo #", where "#" is a delimiter for the start and end of a sentence. For each n-gram in a candidate sentence BLEU generates the maximum occurrence in any of the references. There are several versions of BLEU. They are typically named BLEU-n, where n is the length of the n-gram [14].

Formally BLEU is calculated by computing the percentage of tuples from the hypothesis that appear in the reference as well:

$$P(n) = \frac{Matched(n)}{H(n)},$$

with $H(n)$ being the amount of n-gram tuple in the hypothesis.

$$Matched(n) = \sum_{t_n} \min\left(C_h(t_n), \max_j C_{hj}(t_n)\right),$$

where t_n is a n-gram tuple from hypothesis h and $C_h(t_n)$ is the amount of occurrences of t_n in hypothesis, $C_{hj}(t_n)$ the number of occurrences in reference j of the hypothesis. The final score is calculated as:

$$BLEU = \left(\prod_{i=1}^{N} P(i)\right)^{\frac{1}{N}},$$

where N is the n-gram order, which most commonly is 4 [17].

2.3.2. Distinct

Li et al., 2015 proposed the *distinct-n* metric for measuring diversity in hypotheses. The metric does not take into account the source or reference of a response, but is purely based on n-grams in the hypotheses.

The metric counts the distinct n-grams divided by the number to total words [42]. The division by total words avoids favoring long sequences.

With C, H and t_n from the previous section, $Distinct$ is calculated as follows:

$$Distinct = \frac{C_n(t_n)}{H(1)}$$

2.3.3. Entropy

To account for the different frequency of n-grams that are not accounted for in the *Distinct* score, Zhang et al., 2018 [88] proposed the *Entropy* metric, which takes into account how the n-grams are distributed. For all n-grams N:

$$Entropy = -\frac{1}{\sum_{w \in N} \nu(w)} \sum_{w \in N} \nu(w) \log \frac{\nu(w)}{\sum_{w \in N} \nu(w)},$$

where ν is the frequency of an n-gram.

2.3.4. METEOR

The METEOR has been proposed for machine translation by Banerjee et al., 2005 [8]. Based on unigram matches (or word matches), the score is calculated based on precision, recall, and a fragmentation measurement. The latter aims to judge the quality of the word order. For machine translation METEOR has shown stronger correlation to human evaluation than BLEU.

With w_t as the number of unigrams in the hypothesis and m being the number of unigrams in the hypothesis:

$$Precision = \frac{m}{w_t},$$

$$Recall = \frac{m}{w_r},$$

with w_r being the number of unigrams in the references.

Based on $Precision$ and $Recall$ the weighted harmonic mean is calculated as follows:

$$F_{mean} = \frac{10PR}{R + 9P}$$

The penalty p is calculated based on longer n-grams than just unigrams (compared to $Precision$ and $Recall$). A chunk consists of a set of unigrams that appear in both hypothesis and reference. We calculate the minimal amount of chunks c:

$$p = 0.5(\frac{c}{u_m})^3,$$

where u_m is the amount of mapped unigrams. Based on this term METEOR is calculated as follows:

$$METEOR = F_{mean}(1 - p)$$

2.3.5. Rouge

The Rouge scores has been used as a metric for informativeness in previous response generation works [89, 81].

There are five different variations of this score – two of which are relevant for this work:

- *Rouge-n:* the n-gram overlap in reference and hypothesis
- *Rouge-l:* based on the longest sequence occurring in both reference and hypothesis

With the the number of unigrams in the hypothesis m_h, the number of unigrams in the references m_r and the number of n-gram overlaps o_n:

$$Precision_{Rouge-n} = \frac{o_n}{m_h},$$

$$Recall_{Rouge-n} = \frac{o_n}{m_r}.$$

3. Related Work

This chapter aims to give an overview of open-domain dialogue system research. More specifically, this chapter focuses on works about generation of conversational responses with deep learning.

There are several challenges that arise when creating an open-domain dialogue system, such as semantics, consistency, and interactiveness [37]. In this work however we focus on diversity and informativeness. Increasing such characteristics has become a part of dialogue research, since ordinary S2S models produce responses that are often bland [89] (sometimes called *dull response problem* [6]). This behaviour can be observed in human dialogue as well, where the response will often be "safe but bland" when there is a lack of understanding [37]. We try to mitigate this behaviour in this work.

When creating less bland, and thus more diverse responses, a pitfall is to create very diverse responses that do not carry meaningful information with regard to the source sentence. Therefore this work aims to create diverse and informative responses.

The first section below introduces different models for diverse and informative response generation. The second part gives an overview of publications about response generation metrics.

3.1. Models

One of the first works with an objective function that promotes diversity has been published by Li et al., 2015 [42]. Compared to the standard maximum-likelihood objective the results that they present outperform the previous state of the art in response generation measured with BLEU. Their work measures diversity as number of distinct unigrams (*distinct-1*)

Message	**S2S**	**Li et al., 2015**
I did not get the report from an mi-6 agent.	I don't think that's a good idea.	You did the right thing, did you?
You haven't been given an assignment in this case.	I don't know what you are talking about.	I've been looking all over for you.
I'm losing my grip.	I don't know what you are talking about.	I'm the only one in the world.
I am ready to help.	Come on, come on	I have something we need to talk about.

Table 3.1.: Results of Li et al., 2015 [42] compared to a S2S baseline.

and bigrams (*distinct-2*) and exceeds previous works in both metrics [42]. *Table 3.1* shows responses of this work compared to a a 4-layer S2S baseline.

Following Li et al., several works have tried to improve diversity and informativeness with S2S [69], adversarial [44], and reinforcement learning [45] approaches.

Li et al., 2017 were among the first to leverage adversarial training for response generation. The generative model produces responses, while the discriminator decides whether a response is generated by a human or a machine. Their evaluation shows benefits of the adversarial training process [44], however does not specifically evaluate the diversity of the responses.

Zhang et al., 2018 explicitly leveraged the adversarial training process for informativeness and diversity. Their results outperform existing models on *Reddit* and *Twitter* data they collected but did not publish [91]. A more detailed description of their approach can be found in *4.3 Adversarial Approach*.

While diversity can be improved with adversarial networks or reinforcement learning, other works have published adjustments to S2S models that lead to more diverse and informative responses.
Serban et al., 2016 apply the hierarchical recurrent encoder-decoder network to the response generation task [69]. In their evaluation they outperform existing works, however they note that the responses are "somewhat generic" [69]. The hierarchical approach is included in this work as well, see *4.4 Hierarchical Geometrical Approach*.

Another improvement to S2S dialogue models is the geometrical approach by Gao et al., 2019. The framework increases diversity with an additional Autoencoder. Their work is incorporated into this work and is described in *4.4 Hierarchical Geometrical Approach* as well.

While there have been improvements in diversity and informativeness in recent years [89, 26], the recurrent architecture is still prone to perform poorly in leveraging long-range dependencies in sequences. In *4.5 Section Recurrent Hierarchical Approach* we introduce a novel approach that brings a hierarchical structure to the encoder-part of encoder-decoder model to address this issue.

3.1.1. Large-Scale Pre-Training

Following Vaswani et al.'s introduction of the *Transformer* architecture [76], pre-training has become a prominent topic for language tasks.

The models discussed above employ a supervised learning scheme, where the network is trained on a task-specific dataset. Devlin et al., 2018 (*BERT*) [20] and Radford et al., 2019 (*Generative Pretrained Transformer 2 (GPT-2)*) [62] pre-train their models in an unsupervised fashion. The datasets used for pre-training in those works consist of text from websites, Wikipedia, books, or a combination of them. These models can then be fine-tuned on task-specific datasets for response generation or other tasks such as question answering. Even without fine-tuning (*zero-shot*) these models have outperformed some non-*Transformer* architectures that have been fine-tuned on a specific task [62].

In this work we build a novel *Transformer* model that addresses diversity and informativeness in *Section 4.7 Geometrical Masked Transformer* by combining the idea of structured latent spaces [26] with the *Transformer*.

Name	Utterances	Turns	References
CMU DOG	193	31.6	1
DailyDialog	13,118	7.9	1
DSTC7	2,800,000	n/a	≥ 1 (train, validation), ≥ 6 (test)
Our Reddit dataset	7,300,000	≥ 2	≥ 10
PersonaChat	164,356	>7	1
Topical-Chat	248,014	22	1
Ubuntu Dialog Corpus	7,100,000	7.7	1
Wizard of Wikipedia	201,999	9	1

Table 3.2.: Datasets with open-domain dialogues compared

3.2. Datasets

The works introduced above mostly rely on data from *Reddit* or *Twitter* [89, 26, 90]. Due to copyright issues these datasets are often not published [89, 26], however sometimes instructions on how to recreate them are provided [90]. In the following section we will give an overview of publicly available datasets.

Table 3.2 compares multiple datasets that we encountered in the research for this work. Next to the number of utterances we compare the amount of dialogue turns and the number of references. The latter value denotes how many distinct references are responding to the same source sequence.

While there are various task-specific datasets available [68], we will only cover open-domain datasets. The following part gives some details about the datasets in *Table 3.2*.

Daily Dialog contains 13,000 samples about topics surrounding daily life. The corpus is manually annotated with an intent and an emotion [46].

The *Ubuntu Dialog Corpus* contains conversations surrounding the Linux distribution *Ubuntu*. The corpus contains almost 1 million dialogues with over 7 million dialogue turns in total. While the dataset's domain is limited to the *Ubuntu* distribution, the dialogues are not limited to a specific task [49].

For the creation of the *Persona-Chat* dataset crowd workers are used. For each of the 10,000 dialogues the workers have a persona description assigned and are then asked to have a conversation as that person in which they get to know the persona of the conversation partner [88].

In *CMU Document-grounded conversation (CMU DOG)* participants talk about 30 movies. The grounded information is provided with a *Wikipedia* article [93].

Texts from *Wikipedia* are also used for the *Wizard of Wikipedia* dataset. For over 1,000 different articles each conversation consists of a expert and a learner, where the latter does not have access to the article and wants to learn about the topic [21].

For the 7th Dialog System Technology Challenges (DSTC7) a guide was published to create a dataset based on *Reddit*. The 2.8 million conversations each contain an URL to a

website that contains the grounded information. The validation part of the dataset contains 6 reference responses to the same source sequence [60].

The *Reddit* dataset created for this work is similar to the one from the DSTC7, however it provides multiple references for train, test, and validation split – which is crucial for both training and evaluation of our models.

Topical Chat is designed for conversations around various topics. The reading material for workers included *Wikipedia* articles, *Washington Post* articles, and fun facts about specific topics [32]. The workers were asked to have conversations with both: partners with the same reading material as well as partners with different reading material [32].

3.3. Metrics

In section *2.3 Metrics* of the *Fundamentals* chapter we introduced commonly used metrics in response generation. With increased research in diversity and informativeness several works started to criticize existing metrics. In this section we aim to give an overview of that criticism. Later in section *5.1* we introduce our evaluation scheme to potentially tackle the shortcoming of existing metrics, by evaluation on a multi-reference dataset.

Serban et al., 2016 wrote that previously used metrics such as "cosine similarity, BLEU, Levenshtein distance (...) will primarily favor models that output the same number of punctuation marks and pronouns as are in the test utterances, as opposed to similar semantic content (e.g. nouns and verbs)". They further state that these metrics are "known to lack diversity" [69].

Liu et al., 2016 published a study of unsupervised evaluation metrics for response generation. The results show that commonly used metrics "do not correlate strongly with human judgement" [47]. They did not propose a metric with better positive correlation to human judgement, [47] leaving an open research question.

Bowman et al., 2016 introduced adversarial evaluation, where a discriminant function is trained to distinguish between machine- and human-generated responses [12]. Other works have proposed metrics such as *Rouge*, *Entropy* [89], or precision and recall based on *BLEU* [26]. However none of these metrics have shown significant positive correlation to human evaluation of diversity and informativeness, which is why they are generally accompanied by an additional manual evaluation [37, 90].

4. Models for Diversity and Relevance

Diversity and relevance in dialogue systems has been discussed in previous works. Two recent deep learning publications that published state-of-the-art results are based on an adversarial [89] and a recurrent S2S approach [26].
This chapter starts by describing the underlying training data and then introduces six models:

- **Baseline:** the baseline is a encoder-decoder model based on Gated Recurrent Unit (GRU) [15]
- **Adversarial:** informativeness and diversity based on a GAN
- **Recurrent**
  - **Geometrical:** an approach to increase diversity based on GRU
  - **Geometrical Hierarchical (ours):** advances the geometrical model in order to leverage the hierarchy of the multi-turn dialogue input
- **Transformer**
  - **Masked Transformer:** model that incorporates attention.
  - **Geometrical Transformer (ours):** combines the geometrical approach with the *Masked Transformer*

4.1. Datasets

The research community has gathered a wide variety of datasets for training dialogue systems [68]. For the automated evaluation of this work it is crucial to have multiple responses, also called references, to the same dialogue source (see chapter *5 Evaluation*). Therefore single-reference datasets such as *OpenSubtitles* [75] and *AmazonQA* [34] are not suitable to evaluate the models discussed in the following. In this work we train and evaluate on data from *Reddit*.

The dataset we created contains comments posted on the platform between 2011 and 2013. Each training sample consists of at least two dialogue turns. After the last dialogue turn there are at least 10 comments, or targets, referencing the same previous comments, which is the case in about 5 % of all *Reddit* comments gathered in that period of time. Conversations with more than two dialogue turns were also incorporated by concatenating multiple comments before the first dialogue turn. The data is shuffled and split into 70 % training, 15 % test, and 15 % validation data.

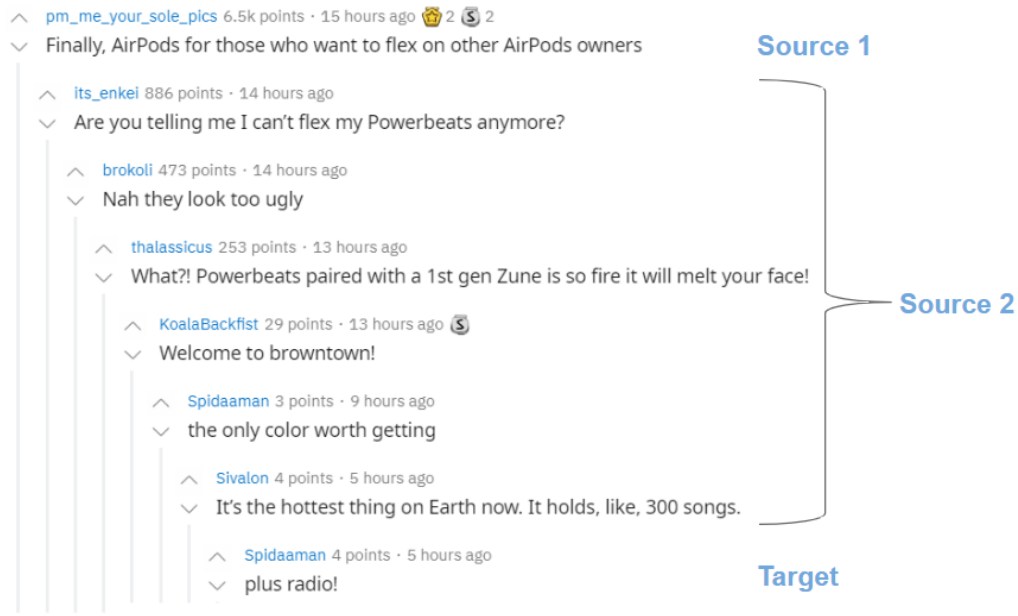

Figure 4.1.: Example of *Reddit* comments and how they appear as source and target in our dataset. Source: *reddit.com*

Our *Reddit* dataset contains 7.3 million comments with an average target length of 12.1 words. *Figure 4.1* shows an example of several comments that are concatenated to resemble a dialogue with two turns.

The data has been cleaned from noise by removing URLs and non-ASCII characters. All characters are made lower-case. The vocabulary contains the 10,000 most common words in the dataset for the adversarial and recurrent approaches.

For the *Transformer*-based approach the vocabulary was bound to the one used in pre-training, which contained 30,000 words from the *Wikipekia* and *BookCorpus* dataset [71].

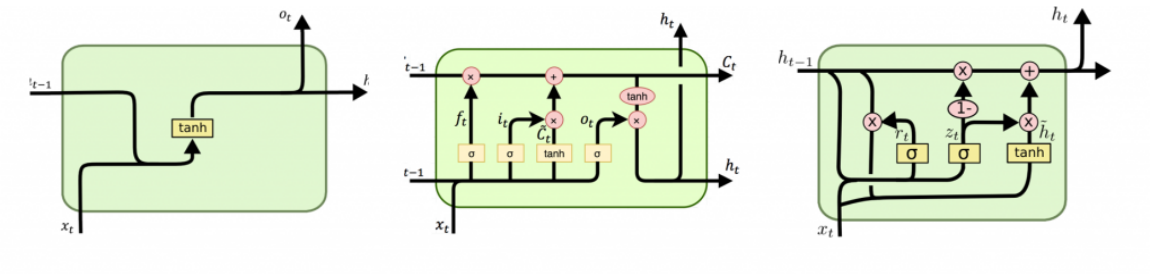

Figure 4.2.: From left to right: RNN, LSTM, and GRU with 0, 3, and 2 gates respectively. With hidden layer vector h_t, output vector x_t, and σ, tanh activation functions [64]

4.2. Baseline

The baseline for this work is a encoder-decoder model. Encoder and decoder both consist of three GRU [15] layers. The GRU is similar to a LSTM [36], however features one less memory gate. This leads to less parameters than the LSTM. Despite the less parameters GRU has shown to be on par with LSTM results for music and speech signal modeling tasks [16]. For faster computation we use the GRU over the LSTM.

Every GRU cell consists of 128 hidden units. Based on Zhao et al., 2017 [92] and Gao et al., 2019 [26] we use softmax sampling to generate multiple hypotheses. *Figure 4.2* illustrates RNN, LSTM, and GRU cells and shows the 3 input, output, and forget gates of the LSTM compared to the 2 input and forget gates of the GRU.

For word representation we train embeddings with vector size 128. This is done for all recurrent models.

4.3. Adversarial Approach

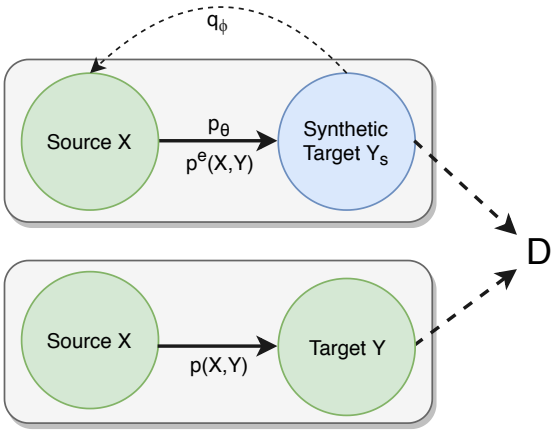

Figure 4.3.: The GAN model with the generator above creating synthetics targets Y_S by approximating $p(X, Y)$ from the training data (bewlow). Both synthetic and original source, target pairs are input for the discriminator D

Semisupervised and unsupervised learning with GANs has allowed for several advancements in computer vision in recent years [18]. While the application of GANs to language problems conveys some problems, they have been successfully applied to language tasks in some works [59, 88].

In adversarial training the generator G generates responses in a dialogue, while the discriminator D judges the quality of this response by either categorizing it as a real sample or rejecting it as a synthetic response that has been generated. This setup allows to employ a discriminator that explicitly promotes diversity and informativeness.

While the original GAN framework proposed by Goodfellow et al., 2014 set new standards on real-valued data such as images [31], generating sequences of discrete outputs comes with additional challenges [38, 84, 86].

For one, the gradient of D is calculated based on the sequences generated by G. This gradient is used to change G's parameters in order to generate potentially more realistic outputs [86]. This approach is flawed with discrete outputs. While a slight change in an image does not inherently change the image, with a confined number of words in a dictionary, this change will most likely not map to an actual word [29, 86].
Secondly, D can only judge a complete sequence of tokens. In a setting where a RNN is employed as generator one token is being generated per time step. Assessing the quality of partial sequences on the other hand is not trivial [86].

Yu et al., 2017 introduced the idea of treating the problem as a sequential decision making process [86] based on Bachman and Precup, 2015 [4]. In this setting G is an agent in a reinforcement learning problem, where the generated token acts as state and the action is the next token to be generated. Which action is executed is defined by the policy function $\pi(a|s, \theta)$, a probability distribution over actions a given state s.

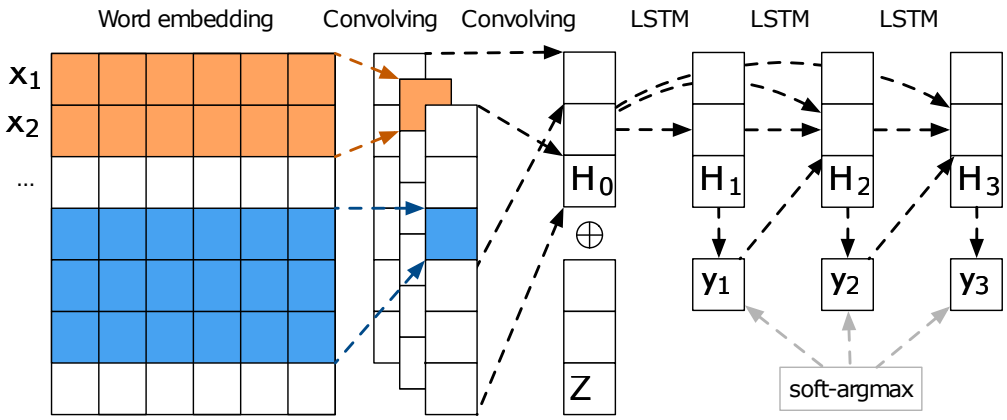

Figure 4.4.: Architecture of the GAN's Generator *G Source: [91]*

Sequences are then generated by the optimal policy π^* with the optimal parameter θ^*. To find the optimal parameter Yu et al. used policy gradients [86].

For word vector representation we employ *word2vec* [54] which is a pre-trained word embedding from Mikolov et al., 2015 at *Google*. It contains 3 million 300-dimension English word vectors and was trained on the *Google News* dataset with around 100 billion words [2].

4.3.1. Adversarial Information Maximization

Yu et al. manage to generate sequences in an adversarial framework. The model evaluated in this work leverages this approach to explicitly improve diversity and relevance of those sequences. The following examines the learning method Adversarial Information Maximization (AIM) proposed by Zhang et al., 2018 that encourages informative and diverse response generation [91]. Let $p_\theta(Y|X)$ be a generative model for or an input sequence $X = \{x_1, x_2, ...x_m\}$ and a corresponding target sequence $Y = \{y_1, y_2, ...y_n\}$. Figure 4.3 gives an overview of the adversarial training process, where the discriminator D distinguishes between "real" (X, Y) pairs from the training data and "fake" pairs (X, Y_s), that contain synthetic Y_s from the generator model $p_\theta(Y|X)$.

The following section lays out the framework's generator and discriminator architecture followed by an introduction to the backward model, that aims to increase relevance of the generated answers.

4.3.1.1. Generator

For a source sentence X the generator G outputs a sequence Y. The architecture of G illustrated in Figure 4.4 consists of an encoder and a decoder. 3 convolutional layers encode X into a hidden vector H_0. A vector Z containing random noise is then added to H_0 in order to produce more diverse responses.

The decoder part consists of 3 LSTM layers, which generate each word y_i given the hidden states H_0, H_{i-1} and Z. The original LSTM introduced by Hochreiter and Schmidhu-

ber, 1997 samples words from a multinomial distribution [36]. Zhang et al. aimed to use the reparameterization trick by Kingma et al., 2013 [39], which is possible by using the *soft-argmax* operation, an approximation using the *Softmax* function.

4.3.1.2. Discriminator

Given a source sentence X the discriminator needs to differentiate between targets from the training data Y and generated targets Y_s. Using two embedding networks based on CNNs the sequences are projected onto an embedding space with fixed dimensions. The source embedding network E_s maps X into the embedding space, while the target embedding network E_t maps both Y and Y_s into fixed-sized space. These vectors are then compared by their cosine similarity $d_{\cos}$.

Discriminator D's objective is to maximize the difference between $d_{\cos}(Y, X)$ and $d_{\cos}(Y_s, X)$, while G tries to minimize the same. Formally the loss is described as:

$$\mathcal{L}_{GAN} = -\mathbb{E}_{T,T_s,S} = [2\tanh^{-1}(d_{\cos}(Y, X) - d_{\cos}(Y_s, X))],$$

where $2\tanh^{-1}$ is used to smooth the gradients. The idea of an embedding-based discriminator comes from *Wasserstein GAN* by Arjovsky et al., 2017 [3].

The gradients from D are propagated to $p_\theta(Y|S)$ with Deterministic Policy Gradient (DPG), as opposed to the policy gradient applied by Yu et al., 2017. For the generated response $T_s(S, Z)$ DPG estimates gradients with a Monte carlo approximation:

$$\nabla_\phi D(T_s, S) = \mathbb{E}_Z \nabla_{T_s} D(T_s, S) \nabla_\phi T_s(S, Z)$$

4.3.1.3. Mutual Information Objective

Mutual information (MI) has been applied to neural response generation by Li et al., 2015 [42], after being introduced to speech recognition by Bahl et al., 1986 [7]. Li et al. suggested that previously used objective functions, such as the likelihood of the target given the source, produce less diverse, less interesting, and less appropriate responses than the proposed MI objective.

While previous works used to re-rank responses based on MI [35], Zhang et al., 2018 leveraged MI in training as well. The informativeness of a response in this model is judged by the mutual information of the source-target pair (X, Y_s). Therefore the unknown joint distribution $p(X, Y)$ is approximated by

$$p_{est}(X, Y) = p_\theta(Y|X)p(X),$$

where the forward model $p_\theta(Y|X)$ is trained to make $p_{est}(X, Y)$ approximate $p(X, Y)$, but still maintain high MI in $p_{est}(X, Y)$.

To save computational cost [9] MI is calculated based on a variational approximation proposed by Barber et al., 2003 [9]. The MI between X and Y is defined as:

$$\mathbb{I}_{p^e}(X, Y) := \mathbb{E}_{p^e(X,Y)} \log \frac{p^e(X, Y)}{p(X)p^e(Y)}$$

$$= H(X) + \mathbb{E}_{p^e(Y)} d_{KL}(p^e(X|Y), q\phi(X|Y)) + \mathbb{E}_{p^e(X,Y)} \log q_\phi(X|Y)$$

$$\geq \mathbb{E}_{p(X)}\mathbb{E}_{p_\theta(Y|X)} \log q_\phi(X|Y) =: \mathcal{L}_{MI}(\theta, \phi),$$

where $q_\phi(X|Y)$ is a backward model approximating $p^e(S|T)$, H is the entropy of a random variable and d_{KL} is the Kullback–Leibler divergence measuring the difference between two probability distributions. $q_\phi(X|Y)$ is implemented with the same CNN and LSTM architecture as the generator, see *Figure 4.4*
With the Monte-Carlo variant of policy gradients called *REINFORCE* [80] and the empirical average as baseline b Zhang et al. approximate the gradient as follows:

$$\nabla_\theta \mathcal{L}_{MI}(\theta, \phi) = \mathbb{E}_{p_\theta(Y|X)}[\log q_\phi(X|Y) - b] \cdot \nabla_\theta \log p_\theta(Y|X),$$

$$\nabla_\phi \mathcal{L}_{MI}(\theta, \phi) = \mathbb{E}_{p_\theta(Y|X)} \nabla_\theta \log q_\phi(X|Y),$$

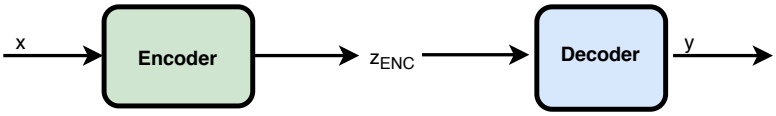

Figure 4.5.: The architecture of the standard encoder-decoder baseline.

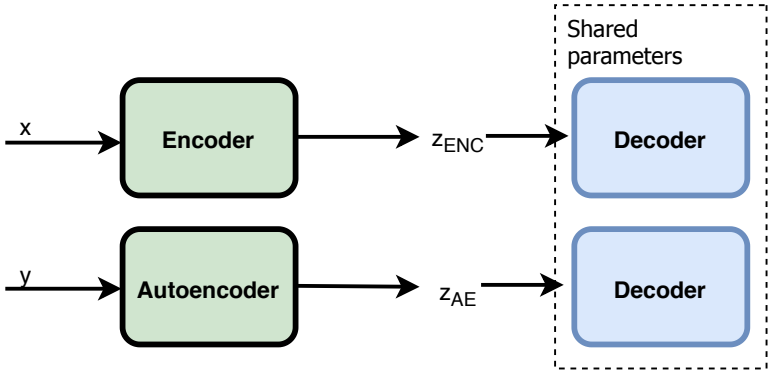

Figure 4.6.: Architecture proposed by Luan et al., 2017

4.4. Geometrical Approach

S2S models often produce relevant responses that tend to be bland [89, 26]. The following section presents the *Geometrical Approach* by Gao et al., 2019.

The idea of combining two models for response generation by sharing a decoder between both encoders in a multi-task setting has been introduced by Luan et al., 2017. Their work is not aimed at increasing diversity specifically and leverages additional persona context to generate responses. It does however propose an additional *Autoencoder* and shared parameters for the decoders [50], as used in the model proposed later on.

Luan et al. introduce a multi-task learning framework that is shown in Figure 4.6. It consists of an encoder-decoder model and an *Autoencoder*. For an input sequence $X = \{x_1, x_2, ...x_m\}$ and a corresponding target sequence $Y = \{y_1, y_2, ...y_n\}$ the S2S model

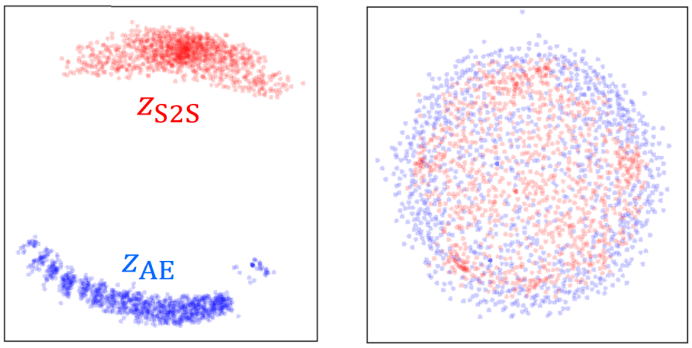

Figure 4.7.: Hidden representations of source (red) and target (blue) sequences from Luan et al., 2017 (left) and Gao et al., 2019 (right) [26]

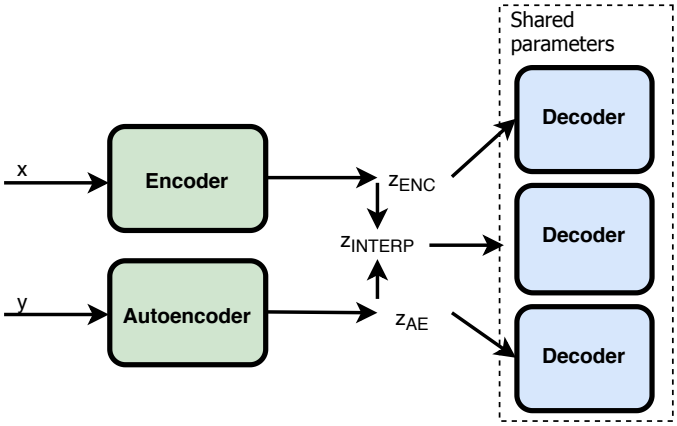

Figure 4.8.: Architecture of the geometrical approaches in section 4.4 and 4.7

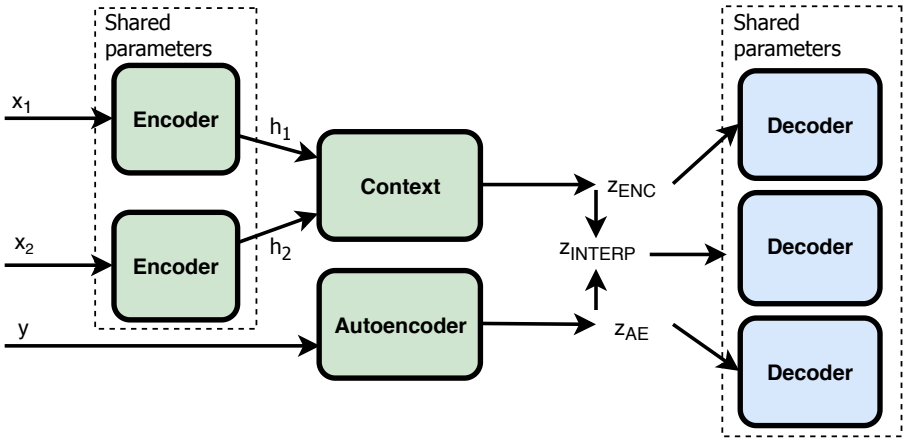

Figure 4.9.: Our architecture of the recurrent hierarchical approach, see section 4.5

uses a LSTM layer to encode X. The last hidden state of the LSTM h_m is then used to initialize the decoder LSTM, which predicts y_t using h_m and y_{t-1}.

Similarly the *Autoencoder* is made up of an encoder and a decoder based on *LSTM*. While the S2S encoder maps from source to target, the *Autoencoder* predicts its input sequence. It shall be noted that in Luan et al.'s work the input sequences are persona data, which is not the case for the model proposed in this work later on. Persona data is additional information about a participant in a dialogue. The personas are often described in a textual profile [88].

The parameters of both decoders are shared.
The training process computes the gradients of the S2S model and updates its weights based on the loss function before updating the weights of the *Autoencoder* based on another batch with the same target user and persona data [50].

Gao et al., 2019 adapted the architecture shown in Figure 4.6 for diverse response generation. In their work the target sequences Y is fed to the *Autoencoder*. While the encoder of the S2S model produces a hidden representation of the source sequence X, the Autoencoder produces such a representation given the target responses Y corresponding

to the source. This is in contrast to Luan et al. that used the *Autoencoder* to encode persona data. The targets Y in the training data potentially include diverse responses.

The hidden space in this setting forms two different clusters, as the left part of figure 4.7 shows. The cluster of red dots contains the hidden vectors representing the source sentence, while the blue cluster contains hidden representations of target sequences. Both representations are in clearly separate areas of the hidden space. However in order for the encoder to leverage the information learned from the *Autoencoder* a hidden space with fewer gaps between the two is desirable for Gao et al. [26]. Their goal is to create a hidden space where hidden representations of corresponding source target-pairs have a short distance between them. When that is the case, the geometrical relationship between source and target has a semantic meaning.

This is archived by structuring the hidden space based on source and target sentence. In a Gaussian distribution sampling further from the mean to increase diversity tends to make samples infrequent and less relevant [26]. However in this geometrical approach the distance from the predicted response shall match the relevance, and the direction must indicate the diversity [26]. The following section presents an approach to archive such a relationship between source and target representations.

Bringing the hidden spaces of both models closer together can be archived via regularization based on minimizing the distance between both hidden vectors:

$$\mathcal{L}_{interpolation} = \frac{1}{|y|} \log p(y|z_{interpolation}),$$

$$z_{interpolation} = uz_{S2S} + (1-u)z_{AE},$$

where $u \in U(0,1)$ is a uniform random variable and $|y|$ is the word count. $\mathcal{L}_{interpolation}$ enforces an interpolation between z_{S2S} and z_{AE}, in order for both of them to generate the same response and also prevents different responses from pointing into a similar direction.

The loss further incorporates a term to ensure that the hidden vectors are scattered over the entire space instead of forming clusters, while still keeping corresponding z_{S2S} and z_{AE} pairs close together:

$$\mathcal{L}_{fusion} = \sum_{i \in batch} \frac{d(z_{S2S}(x_i), z_{AE}(y_i))}{n} - \sum_{i,j \in batch, i \neq j} \frac{d(z_{S2S}(x_i), z_{S2S}(x_j))}{n^2 - n} - \sum_{i,j \in batch, i \neq j} \frac{d(z_{AE}(y_i), z_{AE}(y_j))}{n^2 - n},$$

$$d(a,b) = \sqrt{(a-b)^2},$$

where n is the batch size.

The final loss then combines $\mathcal{L}_{interpolation}$ and $\mathcal{L}_{fusion}$ with a standard multi-task loss:

$$\mathcal{L} = -\frac{1}{|y|} \log p(y|z_{S2S}) - \frac{1}{|y|} \log p(y|z_{AE}) + \alpha \mathcal{L}_{interpolation} + \beta \mathcal{L}_{fusion} \tag{4.1}$$

The regularization results in a mapping between semantic characteristics and geometry of the hidden space. More specifically the semantic diversity maps to the geometrical direction, while the semantic relevance corresponds to the geometrical distance. Both are archived by $\mathcal{L}_{interpolation}$ and $\mathcal{L}_{fusion}$ regularization respectively.

4.5. Recurrent Hierarchical Approach

This section starts by explaining shortcomings of the above *Geometrical Approach* and addresses them with a novel approach that adds a hierarchical structure to the *Geometrical Approach*. In order to avoid confusions with the *Transformer* model – that will be introduced later – we call this the *Recurrent Hierarchical Approach*.

While research in dialogue modeling often focuses on a single-turn conversation, in real-life scenarios conversations span over multiple dialogue turns. Creating relevant responses therefore requires incorporating long-term dependencies into response generation.

The underlying RNN in the previous section suffers from vanishing gradients when dealing with long sequences. This is partly mitigated by the memory gate of a the GRU, however detecting dependencies becomes harder with longer sequences. This work therefore proposes a hierarchical structure in order to make responses more informative based on long input sequences.

A conversation turn in a dialogue consists of a sequence of tokens $c = t_0, t_1, \ldots, t_m$. More than one of such sequences form a multi turn-conversation $m_c = c_0, c_1, \ldots, c_n$, where n is the number of conversation turns.

Based on Serban et al., 2016 this work models this hierarchy with GRUs:

- *Encoder GRU:* models the token-level sequences c.
- *Context GRU:* operates on the sequence-level m_c.

The *Encoder GRU* receives the source dialogue turns x_i as inputs and creates the hidden representation h_i. These are then fed to the *Context GRU*, which is one level higher in the hierarchy and stores past sequences. After processing all c, the hidden state of the context module contains information about all utterances.

The decoder part of the framework then receives the hidden representation of all past utterances, whereas in the basic *Geometrical Approach* it would receive the information from the encoder directly. The parameters in the encoder and in the decoder are shared in order for them to be able to generalize across all training data. The entire architecture is illustrated in the *Appendix (A)*. The general framework – without specific layers – is illustrated in *Figure 4.9*.

The hierarchical structure is set up to enable the model to capture more information from the input sequences. Based on that additional information the generated responses can potentially be more informative by leveraging the additional input.

4.5.1. Training

In training the Adam optimizer has been shown to speed up training compared to Stochastic Gradient Descent (SGD) without noticeably decreasing accuracy.

The non-hierarchical geometrical model has been trained for 3 epochs on a *Nvidia 1080 TI* GPU. The hierarchical version trained for 4 epochs on the same hardware. The hierarchical model required roughly twice the time to train for one epoch, thus leaving room for optimization of the concatenation of both encoder outputs.

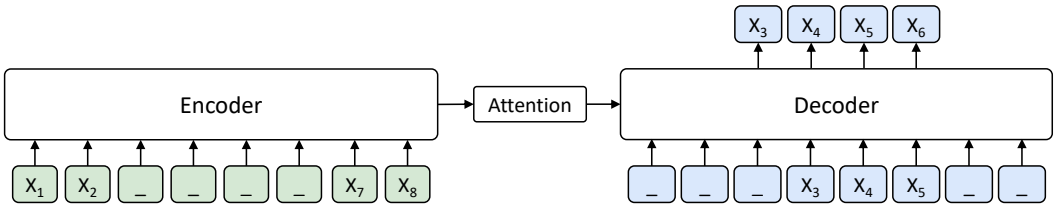

Figure 4.10.: Masking scheme of the Transformer encoder-decoder framework. [71]

4.6. Transformer Approach

Starting in 2018, pre-training has become a very active area of research [20, 61]. Based on the *Transformer* several works have started pre-training models on large amounts of unlabeled data available on the internet. Especially in language understanding *BERT*, *GPT* and *XLNet* have set a new state-of-the-art [20, 85]. For generation tasks those models have not had the same significant impact [74].

With the *Masked Transformer* Song et al., 2019 introduced a *Transformer*-based pre-training approach that outperforms previous architectures in several S2S tasks [71].

In the following the *(Masked) Transformer* approach is introduced, followed by the novel *Geometrical Transformer* approach, which is designed to increase diversity in response generation.

The *Transformer* has an encoder-decoder architecture. Both parts are based on a *Transformer*. Song et al., 2019 proposed a masking scheme in the training process, which is discussed in the following section.

The adversarial model build their vector representation of words based on *word2vec* [54], which is a static word embedding. For the recurrent approaches embeddings of size 128 were trained from scratch. In the *Transformer* architecture we employ pre-trained contextualized word embeddings. The embeddings of the non-*Transformer* models are context-independent, which means one word always has the same static representations. The contextualized embeddings of the *Transformer's* have a dynamic representations based on the surrounding words [13]. Note that in a recurrent network words are fed word by word and in the adversarial approach we employ convolution. Both make use of the sequence order. In the *Transformer* however information about the position needs to be additionally provided [76].

The *Masked Transformer* model consists of 6 encoder and decoder layers respectively. See *Section 2.1.2 Transformer* for more details. The masking scheme proposed by Song et al., 2019 [71] is explained below.

4.6.1. Training

The Transformer encoder and decoder are jointly trained with a masking scheme. This is opposed to BERT, which employs separate training [20]. For each input sequence $x_{u,v} \in \mathbb{X}$

(see chapter *2.1.1 Sequence to Sequence Learning*) the tokens from position u to v are masked with $0 < u < v < |x|$. Let $k = v - u + 1$ be the number of tokens being masked.

In pre-training the model is fed $x_{u,v}$ and predicts the masked sub-sequence $x_{u,v}$. Figure 4.10 illustrates the training process for $u = 3$ and $v = 6$, where the model decoder outputs the initially masked tokens $x_{3,6}$, based on the decoder input x_3, x_4, x_5. Note that if only one token of the encoder input is masked, the decoder input is entirely masked.

Song et al. argue that the masking scheme they propose "forces the encoder to understand the meaning of the unmasked tokens", while encouraging "the decoder to extract useful information from the encoder side". [71]

As for the training data of the *Transformer* we used weights that were pre-trained on the *Wikipekia* and *BookCorpus* data. The fine-tuning on the *Reddit* dataset trained for 10 days on a *Nvidia 1080 Ti*.

4.7. Geometrical Transformer Approach

The novel approach proposed in this work adapts the multi-task setting to structure the latent space discussed in 4.4. The architecture of both encoders are based on the *(Masked Transformer)*, see *Figure 4.8* for the architecture.

As in the recurrent approach, the encoder models transform a input sequence into hidden representations. More specifically both encoders take a sequence and a masked fragment as input and output a hidden state and a padding mask.

Source and target sequence are transformed into hidden states h_{src} and h_{tgt} by two separate encoders. Each encoder consists of 6 Transformer layers with multihead attention. Based on Vaswani et al., 2017 dropout and layer normalization is applied after the self-attention layers [76].

Based on Gao et al., 2019 the hidden representations of source and target sentence h_{src} and h_{tgt} are interpolated:

$$h_{interpolated} = uh_{src} + (1 - u)h_{tgt},$$

with the uniform distribution $u\ U(0, 1)$.

The resulting three hidden representations of source and target sentences h_{src}, h_{tgt}, and $h_{interpolated}$ are passed to three decoders. All of them share weights. Using this information, the decoder generates a response. The architecture of the decoder, like the encoder, is based on 6 *Transformer* layers.

In this multi-task setting there are 3 different loss terms that are combined to adjust the network's weights and biases, see equation (4.1).

4.7.0.1. Training

While the basic *(Masked) Transformer* from the previous section was trained with a batch size of 2000 and a maximum token size of 4096, the architecture with two encoders was too large to fit those batches into memory. Therefore the maximum token size was reduced to

256 tokens, to allow for a batch size of 2000. Since the *Reddit* data has an average sequence length of 12 tokens, this did not come with a decrease in accuracy.
The model has 189 million parameters, compared to 123 million parameters of the model with just one encoder. As with the other models the training dataset included 5 million source-target pairs of reddit data.
For reference this implementation is based on the *Transformer* with a embedding hidden size of 768, feed-forward size 3072, and 16 attention heads, while Song et al., 2019 uses models with hidden size 1024 for the embeddings and 4096 in the feed-forward layers. For this work this model proved not feasible since this architecture wold not fit in the 11.7 Gigabyte (GB) memory of a *Nvidia GTX 1080TI*.

The Transformer encoders and the decoder were pre-trained on the *2016 Conference on Machine Translation (WMT16) News Crawl* datasets with 190 million sentences in English language.

5. Evaluation

The models from the previous chapter are evaluated in the following.

All evaluation is done using multi-reference conversations from *Reddit*. The automated evaluation on our *Reddit* corpus is based on 1000 source sentences and an average number of 14.35 references.

5.1. Metrics

Finding appropriate automated evaluation metrics for neural response generation is an active area of research [47]. Most publications rely on BLEU [58] in addition to human evaluation [47, 72, 77]. Liu et al., 2016 showed that automated metrics such as BLEU only show a small correlation with human judgments when looking at a single reference [47].

This is in contrast to machine translation, where BLEU has a significant positive correlation with human evaluation [24]. For translation systems the evaluation typically factors in multiple references.

Therefore the evaluation setup for this work is based on more than one reference.

For BLEU and METEOR the evaluation of this work employs a metric based on Gao et al., 2019 that calculates $Precision_f$ as an approximation of informativeness, and $Recall_f$ for diversity. Given N_r references for a context x, a single source is evaluated as follows:

$$Precision_f = \frac{1}{N_r} \sum_{i=1}^{N_r} \max_{j \in [1,N_r]} f(r_j, h_i)$$

$$Recall_f = \frac{1}{N_r} \sum_{j=1}^{N_r} \max_{i \in [1,N_r]} f(r_j, h_i)$$

$$F1_f = 2 \frac{Precision_f * Recall_f}{Precision_f + Recall_f}$$

where f is the respective metric 4-gram BLEU or METEOR.

Further evaluation for diversity is solely based on hypotheses. This is opposed to above metrics that take into account the references from the dataset. We calculate those reference-only scores for two underlying metrics:

$$Diversity_g = g(h_0 \ldots., h_{N_r}),$$

where g stands for the two metrics *Entropy* and *Distinct*.

While above metrics evaluate multiple hypotheses to the same source sentence, in a typical use case, the interaction with one source sentence would only require one generated

	$Precision_{BLEU-1}$	$Precision_{BLEU-2}$	$Precision_{BLEU-3}$	$Precision_{BLEU-4}$	$Precision_{METEOR}$
Baseline	0.1311	0.0763	0.0608	0.0547	0.1106
GAN	0.0006	0.005	0.004	0.004	0.0026
Geometrical	0.2253	0.1294	0.0945	0.0792	0.2052
Geometrical Hierarchy (ours)	**0.2723**	0.1627	0.1228	0.106	0.2267
Masked Transformer	0.2288	0.1381	0.1112	0.101	0.1773
Geometrical Transformer (ours)	0.2561	**0.1806**	**0.1539**	**0.14229**	**0.2271**

Table 5.1.: Informativeness of all models compared.

hypothesis. For all references $(r_0, ..., r_{N_r})$ we calculate the score of a metric f with the single highest ranked hypothesis:

$$Single_i = h(r_0, ..., r_{N_r}, h_{best}),$$

where h_{best} is the hypothesis with the highest rank, and i refers to *BLEU*, *METEOR*, and *Rouge*.

In summary there are 11 metrics to rate informativeness:

- $Precision_{BLEU-1}$
- $Precision_{BLEU-2}$
- $Precision_{BLEU-3}$
- $Precision_{BLEU-4}$
- $Precision_{METEOR-1}$
- $Precision_{METEOR-2}$
- $Precision_{METEOR-3}$
- $Precision_{METEOR-4}$
- $Precision_{Rouge-1}$
- $Precision_{Rouge-2}$
- $Precision_{Rouge_l}$

As well as 10 values to asses the diversity:

- $Recall_{BLEU-1}$
- $Recall_{BLEU-2}$
- $Recall_{BLEU-3}$
- $Recall_{BLEU-4}$
- $Recall_{METEOR-1}$
- $Recall_{METEOR-2}$
- $Recall_{METEOR-3}$
- $Recall_{METEOR-4}$
- $Diversity_{Entropy}$
- $Diversity_{Distinct}$

The informativeness and diversity metrics are combined by harmonic mean to provide a single value to asses the model quality:

- $F1_{BLEU-1}$
- $F1_{BLEU-2}$
- $F1_{BLEU-3}$
- $F1_{BLEU-4}$
- $F1_{METEOR}$

To further rate the general quality of the model outputs we calculate 5 scores based on a single hypothesis, while above metrics worked on multiple hypotheses:

- $Single_{BLEU-4}$
- $Single_{METEOR-4}$
- $Single_{Rouge-1}$
- $Single_{Rouge-2}$
- $Single_{Rouge-l}$

5.2. Baseline

A encoder-decoder model with GRU is the baseline model for this work. After training for 3 epochs, the model evaluates to a $F1_{BLEU-4}$ of 0.059.

The encoder-decoder architecture in this baseline and in other works however has shown to generally learn universal replies (*dull response problem* [6, 43, 69]).

This bears the question why replies are this generic. The reasoning behind this has been studied in previous works. Wu et al., 2018 have decomposed the problem into two sub-problems:

1. Target word selection: based on the hidden representation of the input a set of target tokens is selected
2. Word ordering: The selected tokens need to be ordered in order to be grammatically correct.

As for the first part, the tokens with the highest frequency in the training data have the highest probability to appear in any sentence. Thus often used words are predicted more often than specific ones that could convey more information.

This happens because of the conditional likelihood objective. The objective has shown to be suitable for machine translation [6], where source and target sentence share the same semantics. In response generation however, there can be a variety of different answers to the same source sentence and employing the likelihood objective leads to dull responses.

As for the word order Wu et al. attribute generic responses to the fact that tokens are generated based on the previous token and are selected by the high transition probability from the training data [81].

We observe this behaviour in our baseline results as well. Not only are responses often generic such as "I don't know", they frequently start with the same tokens. The generated sequences for the test dataset start with the letter "I" in 57.3 % of cases and in 41.1 % of the cases the word is followed by the token "am".

The following models in this work try to mitigate this behaviour by producing more diverse responses that carry relevant information.

5.3. Adversarial Approach

During training the GAN approach (based on the original author's published code [89]) did not converge. While the results lack semantic meaning, for the sake of completeness the evaluation results are included in tables 5.2, 5.3, and 5.1 anyways.

While the results that take references into account are worse than the baseline, the results on the hypothesis-only metrics are outperforming other models.
In 4-gram $Diversity_{Entropy}$ the GAN archives higher scores than the baseline and the *Transformer*. As for 4-gram $Diversity_{Distinct}$, the model archives the second highest score behind the *Geometrical Hierarchy* model. During testing we did not find any sentences that are grammatically correct or carry semantic meaning.

As an example: for the source sentence *"tom cruise is not doing the reboot of 'the mummy'"* the GAN generates *"multiplayer multiplayer accuse accuse baffled baffled baffled baffled*

	$Recall_{BLEU-1}$	$Recall_{BLEU-2}$	$Recall_{BLEU-3}$	$Recall_{BLEU-4}$	$Recall_{METEOR}$
Baseline	0.1202	0.071	0.0578	0.0528	0.097
GAN	0.008	0.006	0.005	0.005	0.0028
Geometrical	0.1766	0.1039	0.0804	0.0709	0.1621
Geometrical Hierarchy (ours)	**0.218**	**0.1328**	0.1046	0.0936	**0.1772**
Masked Transformer	0.1397	0.0865	0.0724	0.0674	0.1027
Geometrical Transformer (ours)	0.1733	0.1199	**0.1055**	**0.1013**	0.1267

Table 5.2.: Evaluation results for diversity of all models.

baffled baffled bosses bosses bosses bosses bosses bosses bosses bosses bosses bosses bosses bosses bosses bosses endorsement".
The possibly high *Diversity* scores for outputs such as the one above show the importance of multiple metrics and a human evaluation, since a high score in some diversity metrics can be archived with responses that are semantically incorrect.

5.3.1. Training

GANs have shown to be difficult to train in multiple works [31, 53]. This is mostly due to the fact that training two models, namely generator and discriminator, makes the training process increasingly volatile [31, 29, 40, 53]. Parameter tuning for this work proved difficult, since training was limited to one GPU. For comparison: Zhang et al., 2018 trained on 8 GPUs [89]. However their results show that semantically coherent responses can be generated in an adversarial framework. They archive a *BLEU-4* of 0.016.

In the set up for this work however the GAN did not output any semantically coherent sentences. After pre-training for 3 epochs the losses in joint training of both models decreases, however the output was either random words or in some scenarios no words at all. Increasing the learning rate from 0.0001 to 0.001 and 0.01 did not yield better results, neither did switching between *word2vec* and one-hot encoding.

5.4. Geometrical Approach

This section evaluates the results of the model based on the geometrical approach introduced by Gao et al., 2019 [26], while the section after discusses the *Hierarchical Geometrical Approach*.

After training for 3 epochs the non-hierarchical recurrent model archived an $F1_{BLEU-4}$ score of 0.075. Except for $Diversity_{Distinct}$ and $Diversity_{Entropy}$, the model outperforms the baseline and the GAN in both diversity and informativeness metrics.

The model's main improvement has been proposed by Gao et al., 2019. It is a regularization that structures the hidden space. This results in a mapping from distance in the hidden space to relevance, meaning closer vectors are more relevant. Also, sampling in different directions leads to potentially diverse answers. The evaluation on the *Reddit* dataset backs this theory.

For all diversity metrics except *Distinct* the geometrical approach is superior to the baseline further suggesting the benefits of structured latent spaces that Gao et al. reported [26].

5.5. Geometrical Hierarchical Approach

Gao et al. structure the hidden space to make informative responses have a small distance from the hidden representation of the input. This brings improvements over the encoder-decoder baseline. This work tries to further improve the amount of information that is conveyed in the hidden representation of the input. This is done to detect longer dependencies in the input sequence. The hierarchical structure does this by adding multiple levels, where each level creates an input representation of a different dialogue turn.

This leads to an increase of $Recall_{BLEU-4}$ by 32 % from 0.071 to 0.094. Compared to the three models above (baseline, adversarial, and non-hierarchical), this model shows the strongest results in the general metrics, except for $Single_{BLEU-4}$ and $Single_{METEOR}$. Since the latter metrics compare n-grams in hypothesis and references this result could have two explanations:

- The model produces an informative and diverse response with n-grams that do not appear in any references. This could very well be, since the number of references is limited and does not cover all possible meaningful answers.
- The geometrical approach has some shortcomings when it is evaluated on a single response, and its strength lies in generating variety in multiple hypotheses, but does not outperform the baseline when only looking at the highest-ranked response. This would make the approach less useful in real-world user scenarios.

When looking at the results for diversity in *Table 5.2* the model proves superior to all previous results. While it is less surprising that the diversity scores lie above the ones for baseline and *GAN*, the improvement over the non-hierarchical geometrical approach was not expected. The hierarchical structure was introduced to be able to process long-range dependencies in the input, thus improving informativeness, but it seems to also improve diversity.

That implies that informativeness and diversity do not stand in a trade-off relationship. Zhang et al., 2017 argue that "responses of a system may be diverse but uninformative (e.g.,“I don’t know”, “I haven’t a clue”, “I haven’t the foggiest”, “I couldn’t tell you”), and conversely informative but not diverse" [89]. Our results suggest that an increase in informativeness can imply an increase in diversity. Since the encoder of the hierarchical model captures more information, the diversity of the response might increase due to the higher density of information available.

While our expectation was that the hierarchical structure only increases informativeness, it did actually increase diversity as well. Since there was no specific advancement for diversity compared to the non-hierarchical model, it seems that the model produces more diverse results thanks to the higher information density in the hidden state.

5.6. Transformer Approach

The *Transformer* has been pre-trained on the *WMT16 News Crawl* datasets with 190 million sentences, while the other models have only been trained on the *Reddit* data.

	$F1_{BLEU-1}$	$F1_{BLEU-2}$	$F1_{BLEU-3}$	$F1_{BLEU-4}$	$F1_{METEOR}$	$Single_{Rouge-1}$	$Single_{Rouge-2}$	$Single_{Rouge-l}$	$Single_{BLEU-4}$	$Single_{METEOR}$
Baseline	0.1254	0.0736	0.0593	0.0538	0.1034	0.1408	0.0215	0.1201	0.09784	0.24
GAN	0.007	0.005	0.005	0.004	0.0027	0.0126	0	0.0101	0.0003	0.0021
Geometrical	0.198	0.1152	0.0869	0.0748	0.1811	0.3637	0.1575	0.3168	0.074	0.1576
Geometrical Hierarchy (ours)	**0.2421**	**0.1463**	0.113	0.0994	**0.1989**	**0.3863**	**0.177**	**0.3424**	0.0963	0.2168
Masked Transformer	0.1735	0.1063	0.0877	0.0809	0.13	0.2679	0.0894	0.2295	0.0948	0.1717
Geometrical Transformer (ours)	0.2067	0.1441	**0.1252**	**0.1186**	0.1627	0.3178	0.1329	0.2794	**0.1699**	**0.2514**

Table 5.3.: Results of the general evaluation of all models.

The informativeness metric $Precision_{BLEU-4}$ of 0.101 is very close to the performance of the hierarchical geometrical mode, while the $Recall_{BLEU-4}$ is slightly behind. The *METEOR* results are weaker compared to the geometrical models, however it still clearly outperforms the baseline.

That relevance is increased with this architecture aligns with other works that have shown that transformer-based models exceed recurrent models at capturing long-range dependencies [90].

We used the pre-trained weights from Song et al., 2019 [71] before fine-tuning on our *Reddit* corpus. Since we used the same vocabulary as Song et al. 11.6 % of tokens in the dataset are replaced by the "UNK" token, the so called out-of-vocabulary problem [10]. Out-of-vocabulary describes all words that appear in a corpus, that are not in the vocabulary that a model is trained with [10]. Since the pre-training process used books and *Wikipedia* articles the words that are commonly used differ from the more informal language on *Reddit*. The *Transformer* results could be further improved when working on a vocabulary that covers more words from the *Reddit* corpus.

5.7. Geometrical Transformer Approach

This model combines the regularization to structure the latent space with the *Transformer* architecture that outperformed the GRU encoder-decoder baseline.

The *Geometrical Transformer* proves superior in the general metrics $F1_{BLEU-4}$, $Single_{BLEU-4}$, $Single_{METEOR}$. However falls behind the recurrent models in lower n-gram BLEUs, METEOR, and Rouge scores.

In diversity scores the model archives the highest score for $Recall_{BLEU-3}$ $Recall_{BLEU-4}$, but falls behind the recurrent models in $Recall_{METEOR}$.

For informativeness the $Precision_{BLEU-4}$ and $Precision_{METEOR}$ scores are the highest out of all models. The performance is slightly weaker in *BLEU* with n-grams 1 to 3, however most other works focus on 4-gram *BLEU* for evaluation.

In summary, the *Geometrical Transformer* shows superiority in informativeness, however not making significant improvements in diversity compared to the recurrent geometrical approach. The increase in informativeness can be attributed to the increased size in parameters: 189 million parameters compared to the 12 million parameters of the hierarchical recurrent approach. Only by using the pre-trained weights of Song et al., 2019 [71] was it possible to train (or fine-tune) the much larger *Transformer* models in a similar amount of time on the same hardware as the recurrent models.

It seems that the results can be largely attributed to pre-training, and diversity results could be further improved by fine-tuning on the *Reddit* dataset, given that the pre-trained weights used for the *Autoencoder* are pre-trained as an encoder, not as an *Autoencoder* of target sentences. Due to time and resource constraints further fine-tuning was not possible for this work.

Source Sentence:	Green day got their set cut from 45 to 25 minutes at the iheart radio festival so usher could have more time on stage, here is billie joe 's reaction... Note how the bassist instantly follows billie joe's reaction and starts smashing his bass. teamwork. I wish it would've panned over to show mike smashing his bass. They showed it at the end of the clip looking pristine, i don't think he did much damage.
Baseline	It's a lot of people.
Hierarchical Geometrical	I've been listening to this.
	I think he's a douche.
	That's the best thing i've ever seen.
Geometrical Transformer	This is the best thing i have seen all day.
	I love this album so much.
	Or i love youtube.

Table 5.4.: Model outputs compared

5.8. Comparison

This work compares 6 models. The underlying neural network architeture can be divided into three categories: RNN, GAN, and Transformer. The following gives a summary of results the different technologies have archived. *Table 5.4* shows an example source sentence and the respective outputs of the baseline compared to our novel approaches.

The different approaches to response generation discussed in this work show a fundamental difference when trying to archive diversity and relevance in dialogues. While the adversarial approach archives high diversity in some scores, it performs poorly for relevance. Recurrent and *Transformer*-based approaches archive higher relevance, but – especially for the baseline – diversity is low compared to the models that are introduced in this work.

Generating diverse and relevant dialogues comes with two different challenges for all architectures. When working with encoder-decoder models and the geometrical approach, diversity needs to be improved. For the GAN approach, the challenge is to provide relevant responses.

While other works formulate diversity and relevance as a clear trade-off [89], the results in this work show that improving one objective does not mean the other objective will suffer.

We have seen clear improvements when combining pre-training, *Transformers*, and Gao et al., 2019's geometrical approach.

The number of trainable parameters has made hyperparameter tuning easier for the recurrent approaches on limited hardware resources. However the large pre-trained model outperforms all other models in most metrics and still has room for improvement with further scaling and additional fine-tuning.

This work has proposed two novel approaches, namely the *Hierarchical Geometrical (Recurrent)* and the *Geometrical (Masked) Transformer* that combine different techniques to improve diversity and informativeness.

We have presented evaluation results for 5 informativeness metrics, 7 diversity, metrics, and 10 general metrics that combine both informativeness and diversity. In all metrics the highest score has been archived by one of the novel approaches.

$Recall_{BLEU-4}$	$Recall_{METEOR}$	*Entropy*				*Distinct*			
		1	2	3	4	1	2	3	4
-0.1248	0.1172	0.1721	0.2281	0.3066	0.3380	0.0165	0.0597	0.1013	0.1506

Table 5.5.: Diversity correlation of human evaluation and automated metrics

$Precision_{BLEU-4}$	$Precision_{METEOR}$	Rouge					
		1 P	1 R	1 F	2 P		
0.0160	-0.0001	-0.1212	0.0283	-0.0707	-0.0762		
Rouge						$Single_{BLEU}$	$Single_{METEOR}$
2 R	2 F	L P	L R	L F			
0.0283	-0.0399	-0.1266	0.0317	-0.0880		-0.0150	0.0638

Table 5.6.: Correlations between human and automated evaluation of informativeness

5.9. Metric Evaluation

Automated evaluation metrics are an active area of research. Since multiple works state that there is no clear indication of a proper metric for informativeness and diversity in response generation [47, 89, 26], this work evaluates the correlation to human evaluation for the metrics that have been employed.

Therefore, we randomly select 100 source sentences from the *Reddit* test dataset. With the hierarchical geometrical model we generate five hypotheses per source. The automated evaluation of responses is based on five references per source.

For the human evaluation, we annotate each hypothesis with an informativeness score $[1, 5]$, and rank the overall diversity of the answers in the same range.

The Pearson correlation between human evaluation and the respective metrics are shown in *Table 5.5* for diversity and *Table 5.1* for informativeness.

The experiment shows the most significant correlation for informativeness with the human evaluation for the *Rouge-L* recall, followed by *Rouge-1* recall.

For diversity all four *Entropy* results showed the highest correlation, followed by *Distinct*, and $Recall_{METEOR}$.

Out of all models in automated evaluation the highest *Entropy* scores were archived by the *Geometrical Transformer*, the highest $Recall_{METEOR}$ by the *Geometrical Hierarchical* model. The latter was also the best in terms of the *Rouge* diversity metrics.

The *Recall* diversity metrics are based on both reference and hypothesis, while *Distinct* and *Entropy* are based only on the hypothesis. The latter can rate random responses without semantic meaning as diverse.

Distinct has shown to be the least coherent metric. Since all the distinct n-grams are divided by the number of words, models that make use of a very small vocabulary can archive high evaluation scores, which does not lead to diverse and interesting answers according to human evaluation.

For informativeness we have seen the highest correlation with *Rouge* and therefore propose this for future evaluation rather than BLEU. For diversity *Entropy* has shown the

highest correlation. Since a high score can be archived with semantically meaningless responses we advise against using this metric without a human evaluation or other metrics that factor in hypotheses.

Our correlation results are not conclusive when comparing single hypotheses metrics ($Single_{BLEU-4}$, $Single_{METEOR}$) with multi hypotheses metrics ($Recall_{BLEU-4}$, $Recall_{METEOR}$) for informativeness. We observe higher correlations with multiple hypotheses when using *BLEU*, however when using *METEOR* the single hypotheses evaluation shows higher positive correlation.

6. Conclusion

6.1. Discussion

The *Hierarchical Geometrical* model we proposed leverages structured latent spaces and a hierarchical encoder and provides better evaluation results than our baseline model.

This work has shown that the pre-trained *Transformer* architecture [76, 20, 71] leads to significant improvements to diversity and informativeness in response generation.

Combining the regularization that structures the hidden space with the pre-trained *Transformer* has further improved results on our evaluation scheme and both novel approaches beat all other models in different metrics.

While previous works have seen diversity and informativeness as a trade-off [89, 91], the results of this work suggest that they can be jointly optimized for S2S models.

A human evaluation to compare the automated metrics has shown the highest correlations with the *Rouge* score for rating informativeness, and *Entropy* for diversity. We have seen little positive correlation for the often used [47, 89, 26] BLEU metric.

6.2. Future Work

This work created a model for more diverse and informative dialogues. While the S2S approaches already provide a strong baseline in terms of informativeness, this could further be boosted with additional information – such as the textual information behind the URLs that are posted [25] – that has been neglected in this work. While additional context or persona information have been studied, combining this with a diversity-promoting objective is yet to be explored to the best of our knowledge.

Without additional information the *Transformer*-based approached could be further improved in two ways: the out-of-vocabulary problem can be avoided by switching from word-level to Byte Pair Encoding (BPE) [90, 62], and the informativeness of answers could be further improved by re-ranking based on mutual information estimation (see *4.3.1.3 Mutual Information*).

Structuring latent spaces for more diversity is highly dependent on the underlying training data. Creating datasets with multiple references that specifically contain diverse responses would benefit the existing architectures.

While this work has leveraged a multi-reference dataset for automated evaluation, the automated evaluation on single-reference data needs to be further explored, as there is currently no metric with significant positive correlation to human evaluation [47]. Embedding-based metrics have shown to be promising in some works [89].

While the above stated future work is purely focused on neural response generation, it is also possible to extend the frameworks to a dialogue system with a Automatic Speech Recognition (ASR) component.

Bibliography

[1] Jay Alammar. *Transformer Illustrations*. http://jalammar.github.io/illustrated-transformer/. Accessed: 2019-11-14.

[2] Google Code Archive. *word2vec: Tool for computing continuous distributed representations of words.* https://code.google.com/archive/p/word2vec/. Accessed: 2020-01-21.

[3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. "Wasserstein Generative Adversarial Networks". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. International Convention Centre, Sydney, Australia: PMLR, June 2017, pp. 214–223. URL: http://proceedings.mlr.press/v70/arjovsky17a.html.

[4] Philip Bachman and Doina Precup. "Data generation as sequential decision making". In: *Advances in Neural Information Processing Systems*. 2015, pp. 3249–3257.

[5] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *CoRR* abs/1409.0473 (2014).

[6] Ashutosh Baheti et al. "Generating More Interesting Responses in Neural Conversation Models with Distributional Constraints". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. 2018, pp. 3970–3980. URL: https://www.aclweb.org/anthology/D18-1431/.

[7] Lalit R Bahl et al. "Maximum mutual information estimation of hidden Markov model parameters for speech recognition". In: *proc. icassp*. Vol. 86. 1986, pp. 49–52.

[8] Satanjeev Banerjee and Alon Lavie. "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*. 2005, pp. 65–72.

[9] David Barber and Felix Agakov. "The IM Algorithm: A Variational Approach to Information Maximization". In: *Proceedings of the 16th International Conference on Neural Information Processing Systems*. NIPS'03. Whistler, British Columbia, Canada: MIT Press, 2003, pp. 201–208.

[10] Issam Bazzi and James Glass. "Modelling Out-of-Vocabulary Words for Robust Speech Recognition". AAI0804528. PhD thesis. USA, 2002.

[11] Y. Bengio, Paolo Frasconi, and Patrice Simard. "Problem of learning long-term dependencies in recurrent networks". In: Feb. 1993, 1183–1188 vol.3. DOI: 10.1109/ICNN.1993.298725.

[12] Samuel R. Bowman et al. "Generating Sentences from a Continuous Space". In: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 10–21. DOI: 10.18653/v1/K16-1002. URL: https://www.aclweb.org/anthology/K16-1002.

[13] José Camacho-Collados and Mohammad Taher Pilevar. "From Word To Sense Embeddings: A Survey on Vector Representations of Meaning". In: *Journal of Artificial Intelligence Research* 63 (Dec. 2018), pp. 743–788. DOI: 10.1613/jair.1.11259.

[14] Boxing Chen and Colin Cherry. "A systematic comparison of smoothing techniques for sentence-level bleu". In: *Proceedings of the Ninth Workshop on Statistical Machine Translation*. 2014, pp. 362–367.

[15] Kyunghyun Cho et al. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. DOI: 10.3115/v1/D14-1179. URL: https://www.aclweb.org/anthology/D14-1179.

[16] Junyoung Chung et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling". English (US). In: *NIPS 2014 Workshop on Deep Learning, December 2014*. 2014.

[17] *Computing BLEU scores, Universtiy of Washington*. http://ssli.ee.washington.edu/~mhwang/pub/loan/bleu.pdf. Accessed: 2020-01-15.

[18] Antonia Creswell et al. "Generative adversarial networks: An overview". In: *IEEE Signal Processing Magazine* 35.1 (2018), pp. 53–65.

[19] Li Deng, Dong Yu, et al. "Deep learning: methods and applications". In: *Foundations and Trends® in Signal Processing* 7.3–4 (2014), pp. 197–387.

[20] Jacob Devlin et al. "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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. 2019, pp. 4171–4186. URL: https://www.aclweb.org/anthology/N19-1423/.

[21] Emily Dinan et al. "Wizard of Wikipedia: Knowledge-Powered Conversational Agents". In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. 2019. URL: https://openreview.net/forum?id=r1l73iRqKm.

[22] Robin IM Dunbar, Anna Marriott, and Neil DC Duncan. "Human conversational behavior". In: *Human nature* 8.3 (1997), pp. 231–246.

[23] Maxine Eskénazi, Laurence Devillers, and Joseph Mariani, eds. *Advanced Social Interaction with Agents : 8th International Workshop on Spoken Dialog Systems*. Cham, 2019.

[24] Michel Galley et al. "deltaBLEU: A Discriminative Metric for Generation Tasks with Intrinsically Diverse Targets". 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 2: Short Papers*. 2015, pp. 445–450. URL: https://www.aclweb.org/anthology/P15-2073/.

[25] Michel Galley et al. "Grounded Response Generation Task at DSTC7". In: *AAAI Dialog System Technology Challenges Workshop*. 2019.

[26] Xiang Gao et al. "Jointly Optimizing Diversity and Relevance in Neural Response Generation". In: *NAACL-HLT 2019* (2019).

[27] Jonas Gehring et al. "Convolutional sequence to sequence learning". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 1243–1252.

[28] *Generative Adversarial Networks architecutre*. https://skymind.ai/wiki/generative-adversarial-network-gan. Accessed: 2020-01-10.

[29] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[30] Ian Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 2672–2680. URL: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[31] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[32] Karthik Gopalakrishnan et al. "Topical-Chat: Towards Knowledge-Grounded Open-Domain Conversations". In: Sept. 2019, pp. 1891–1895. DOI: 10.21437/Interspeech.2019-3079.

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

[34] Mansi Gupta et al. "AmazonQA: A Review-Based Question Answering Task". In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*. 2019, pp. 4996–5002. DOI: 10.24963/ijcai.2019/694. URL: https://doi.org/10.24963/ijcai.2019/694.

[35] huong. ho. "Neural Conversational Model with Mutual Information Ranking". In: 2017.

[36] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[37] Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. "Challenges in Building Intelligent Open-domain Dialog Systems". In: *arXiv preprint arXiv:1905.05709* (2019).

[38] Ferenc Huszár. "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" In: *arXiv preprint arXiv:1511.05101* (2015).

[39] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 2014. URL: http://arxiv.org/abs/1312.6114.

[40] Naveen Kodali et al. "On convergence and stability of gans". In: *arXiv preprint arXiv:1705.07215* (2017).

[41] Bernd J. Kröger. *Neuronale Modellierung der Sprachverarbeitung und des Sprachlernens : Eine Einführung*. Berlin, Heidelberg, 2018.

[42] Jiwei Li et al. "A Diversity-Promoting Objective Function for Neural Conversation Models". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 110–119. DOI: 10.18653/v1/N16-1014. URL: https://www.aclweb.org/anthology/N16-1014.

[43] Jiwei Li et al. "A Persona-Based Neural Conversation Model". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. 2016. URL: https://www.aclweb.org/anthology/P16-1094/.

[44] Jiwei Li et al. "Adversarial Learning for Neural Dialogue Generation". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 2157–2169. DOI: 10.18653/v1/D17-1230. URL: https://www.aclweb.org/anthology/D17-1230.

[45] Jiwei Li et al. "Deep Reinforcement Learning for Dialogue Generation". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 1192–1202. DOI: 10.18653/v1/D16-1127. URL: https://www.aclweb.org/anthology/D16-1127.

[46] Yanran Li et al. "DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset". In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 986–995. URL: https://www.aclweb.org/anthology/I17-1099.

[47] Chia-Wei Liu et al. "How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 2122–2132. DOI: 10.18653/v1/D16-1230. URL: https://www.aclweb.org/anthology/D16-1230.

[48] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. "Recurrent Neural Network for Text Classification with Multi-Task Learning". In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. IJCAI'16. New York, New York, USA: AAAI Press, 2016, pp. 2873–2879. ISBN: 9781577357704.

[49] Ryan Lowe et al. "The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems". In: *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Prague, Czech Republic: Association for Computational Linguistics, Sept. 2015, pp. 285–294. DOI: 10.18653/v1/W15-4640. URL: https://www.aclweb.org/anthology/W15-4640.

[50] Yi Luan et al. "Multi-Task Learning for Speaker-Role Adaptation in Neural Conversation Models." In: *IJCNLP(1)*. Ed. by Greg Kondrak and Taro Watanabe. Asian Federation of Natural Language Processing, 2017, pp. 605–614. ISBN: 978-1-948087-00-1. URL: http://dblp.uni-trier.de/db/conf/ijcnlp/ijcnlp2017-1.html#LuanBDGG17.

[51] Andrew Maas. *Stanford University CS 224S: Spoken Language Processing*. https://web.stanford.edu/class/cs224s/lectures/224s.17.lec10.pdf. Accessed: 2019-11-20.

[52] Warren S McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[53] Lars M. Mescheder. "On the convergence properties of GAN training". In: *CoRR* abs/1801.04406 (2018). arXiv: 1801.04406. URL: http://arxiv.org/abs/1801.04406.

[54] Tomas Mikolov et al. *Computing numeric representations of words in a high-dimensional space*. US Patent 9,037,464. May 2015.

[55] Mor Naaman, Jeffrey Boase, and Chih-Hui Lai. "Is it really about me?: message content in social awareness streams". In: *Proceedings of the 2010 ACM conference on Computer supported cooperative work*. ACM. 2010, pp. 189–192.

[56] Andrew Ng. *C5W3L07: Attention Model Intuition*. https://www.youtube.com/watch?v=SysgYptB198. Accessed: 2019-09-30.

[57] *Number of digital voice assistants in use worldwide from 2019 to 2023, Statista*. https://www.go-gulf.com/blog/virtual-digital-assistants/. Accessed: 2019-09-05.

[58] Kishore Papineni et al. "BLEU: a method for automatic evaluation of machine translation". In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2002, pp. 311–318.

[59] Ofir Press et al. "Language generation with recurrent generative adversarial networks without pre-training". In: *arXiv preprint arXiv:1706.01399* (2017).

[60] Lianhui Qin et al. "Conversing by Reading: Contentful Neural Conversation with On-demand Machine Reading". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 5427–5436. DOI: 10.18653/v1/P19-1539. URL: https://www.aclweb.org/anthology/P19-1539.

[61] Alec Radford et al. "Improving language understanding by generative pre-training". In: *URL https://openai.com/blog/language-unsupervised/* (2018).

[62] Alec Radford et al. "Language models are unsupervised multitask learners". In: *OpenAI Blog* 1.8 (2019).

[63] Alan Ritter, Colin Cherry, and William B Dolan. "Data-driven response generation in social media". In: *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics. 2011, pp. 583–593.

[64] *RNN, LSTM, GRU*. `http://dprogrammer.org/rnn-lstm-gru`. Accessed: 2020-01-20.

[65] Lina Maria Rojas-Barahona et al. "A Network-based End-to-End Trainable Task-oriented Dialogue System". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*. 2017, pp. 438–449. URL: `https://www.aclweb.org/anthology/E17-1042/`.

[66] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[67] Tim Salimans et al. "Improved Techniques for Training GANs". In: *CoRR* (2016). arXiv: `1606.03498`. URL: `http://arxiv.org/abs/1606.03498`.

[68] Iulian Vlad Serban et al. "A survey of available corpora for building data-driven dialogue systems: The journal version". In: *Dialogue & Discourse* 9.1 (2018), pp. 1–49.

[69] Iulian V. Serban et al. "Building End-to-End Dialogue Systems Using Generative Hierarchical Neural Network Models". In: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI'16. Phoenix, Arizona: AAAI Press, 2016, pp. 3776–3783.

[70] Heung-Yeung Shum, Xiao-dong He, and Di Li. "From Eliza to XiaoIce: challenges and opportunities with social chatbots". In: *Frontiers of Information Technology & Electronic Engineering* 19.1 (2018), pp. 10–26.

[71] Kaitao Song et al. "MASS: Masked Sequence to Sequence Pre-training for Language Generation". In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. 2019, pp. 5926–5936. URL: `http://proceedings.mlr.press/v97/song19d.html`.

[72] Alessandro Sordoni et al. "A Neural Network Approach to Context-Sensitive Generation of Conversational Responses". In: *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*. 2015, pp. 196–205. URL: `https://www.aclweb.org/anthology/N15-1020/`.

[73] I Sutskever, O Vinyals, and QV Le. "Sequence to sequence learning with neural networks". In: *Advances in NIPS* (2014).

[74] Xu Tan. "Introducing MASS – A pre-training method that outperforms BERT and GPT in sequence to sequence language generation tasks". In: *URL https://www.microsoft.com/en-us/research/blog/introducing-mass-a-pre-training-method-that-outperforms-bert-and-gpt-in-sequence-to-sequence-language-generation-tasks/* (2019).

[75] Jörg Tiedemann. "News from OPUS-A collection of multilingual parallel corpora with tools and interfaces". In: *Recent advances in natural language processing*. Vol. 5. 2009, pp. 237–248.

[76] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[77] Oriol Vinyals and Quoc Le. "A neural conversational model". In: *arXiv preprint arXiv:1506.05869* (2015).

[78] Ritika Wason. "Deep learning: Evolution and expansion". In: *Cognitive Systems Research* 52 (2018), pp. 701–708. ISSN: 1389-0417. DOI: https://doi.org/10.1016/j.cogsys.2018.08.023. URL: http://www.sciencedirect.com/science/article/pii/S1389041717303546.

[79] John A Waterworth and Mike Talbot. *Speech and language-based interaction with machines: Towards the conversational computer*. Ellis Horwood Chichester, 1987.

[80] Ronald J. Williams. "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning". In: *Mach. Learn.* 8.3–4 (May 1992), pp. 229–256. ISSN: 0885-6125. DOI: 10.1007/BF00992696. URL: https://doi.org/10.1007/BF00992696.

[81] Bowen Wu et al. "Why Do Neural Response Generation Models Prefer Universal Replies?" In: *arXiv preprint arXiv:1808.09187* (2018).

[82] Chen Xing et al. "Topic aware neural response generation". In: *Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

[83] Can Xu et al. "Neural Response Generation with Meta-words". In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. 2019, pp. 5416–5426. URL: https://www.aclweb.org/anthology/P19-1538/.

[84] Jingjing Xu et al. "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*. 2018, pp. 3940–3949.

[85] Zhilin Yang et al. "XLNet: Generalized Autoregressive Pretraining for Language Understanding". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 5754–5764. URL: http://papers.nips.cc/paper/8812-xlnet-generalized-autoregressive-pretraining-for-language-understanding.pdf.

[86] Lantao Yu et al. "Seqgan: Sequence generative adversarial nets with policy gradient". In: *Thirty-First AAAI Conference on Artificial Intelligence*. 2017.

[87] Guang Zhang et al. "Functional analysis of an APSES transcription factor (GlSwi6) involved in fungal growth, fruiting body development and ganoderic-acid biosynthesis in Ganoderma lucidum". In: *Microbiological Research* (2018), pp. 280–288. ISSN: 0944-5013. DOI: https://doi.org/10.1016/j.micres.2017.12.015. URL: http://www.sciencedirect.com/science/article/pii/S0944501317309758.

[88] Saizheng Zhang et al. "Personalizing Dialogue Agents: I have a dog, do you have pets too?" In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 2204–2213. DOI: 10.18653/v1/P18-1205. URL: https://www.aclweb.org/anthology/P18-1205.

[89] Yizhe Zhang et al. "Adversarial feature matching for text generation". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 4006–4015.

[90] Yizhe Zhang et al. "DialoGPT: Large-Scale Generative Pre-training for Conversational Response Generation". In: *CoRR* abs/1911.00536 (2019). arXiv: 1911.00536. URL: http://arxiv.org/abs/1911.00536.

[91] Yizhe Zhang et al. "Generating informative and diverse conversational responses via adversarial information maximization". In: *Advances in Neural Information Processing Systems*. 2018, pp. 1810–1820.

[92] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. "Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, July 2017, pp. 654–664. DOI: 10.18653/v1/P17-1061. URL: https://www.aclweb.org/anthology/P17-1061.

[93] Kangyan Zhou, Shrimai Prabhumoye, and Alan W. Black. "A Dataset for Document Grounded Conversations". In: *EMNLP*. 2018.

A. Appendix

A.1. Recurrent Hierarchical Architecture

Section *4.5 Recurrent Hierarchical Approach* introduces a novel RNN that combines structured latent spaces with a hierarchical encoder. The following illustrates the specific neural network architecture.

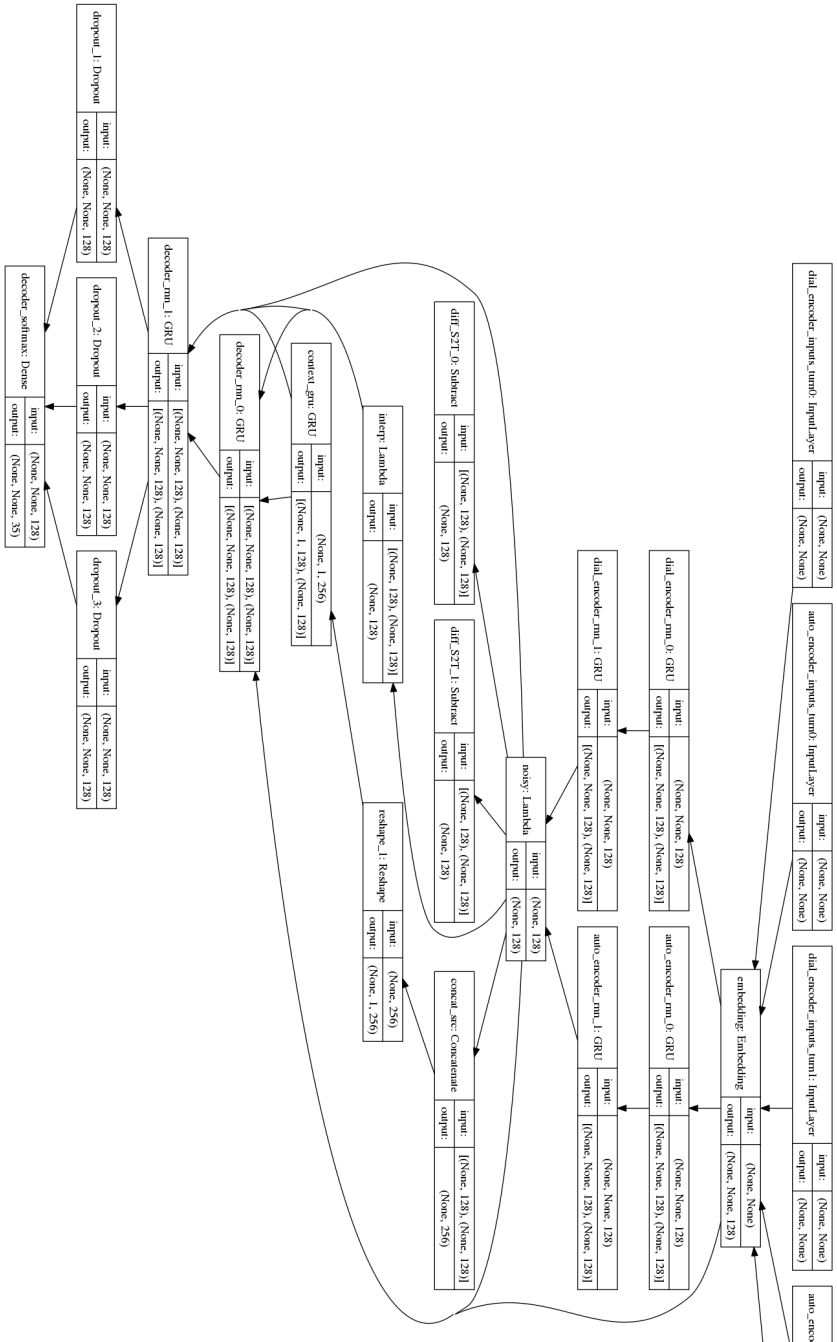

Figure A.1.: Architecture of the hierarchical, recurrent model