Response Generation For An Open-Ended Conversational Agent

by

Nouha Dziri

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

Conversation plays a key role in maintaining humans well-being. It constitutes the most natural way of interacting verbally with each other. Over the past decade, dialogue systems have become omnipresent in our daily lives, assisting our daily schedule and routine. Recently, the emergence of neural network models has shown promising results in solving problems such as scalability and language-independence that conventional dialogue system fail to cope with.

In particular, Sequence-to-Sequence (Seq2Seq) models have witnessed a notable success in generating natural conversational exchanges by sampling words sequentially conditioned on previous words. However, these models still lag far behind human capabilities in terms of the conversations that they can perform. Notwithstanding the syntactically well-formed responses generated by Seq2Seq models, they are prone to be generic, dull and off-context such as " *i don't know*" or " *i'm not sure what you're talking about*".

In this work, we introduce a Topical Hierarchical Recurrent Encoder Decoder (THRED), a novel, fully data-driven, multi-turn response generation system intended to produce contextual and topic-aware responses. Our model is built upon the basic Seq2Seq model by augmenting it with a hierarchical joint attention mechanism that incorporates topical concepts and previous interactions into the response generation. We demonstrate that incorporating conversation history and topic information with our novel method improves generated conversational responses. To train our model, we provide a clean and high-quality conversational dataset mined from Reddit comments. Addi-

tionally, we propose two novel quantitative metrics for measuring the quality of the generated responses, dubbed Semantic Coherence and Response Echo Index. Our experiments on these quantitative metrics along with human evaluation demonstrate that the proposed model is able to generate more diverse and contextually relevant responses compared to the strong baselines. In contrast to the widely used OpenSubtitles dataset, we exhibit that Reddit dataset can be considered as a better resource for training future conversational systems. Furthermore, we show that both quantitative metrics agree reasonably with human judgment, making a step towards a good automatic evaluation procedure.

Preface

Prior to evaluating our work, we obtained an ethics approval from the University of Alberta Ethics Board, Project Name "Response Generation For An Open-Ended Conversational Agent", No. Pro00075657, October 24th 2017.

To my parents and my sister,

For being my support, my inspiration. For teaching me the values and attitudes that have shaped the person I am today.

If you want to succeed, double your failure rate.

– Thomas J. Watson, IBM Chairman.

Acknowledgements

First and formost, I would like to thank my supervisor Professor Osmar Zaiane for his constant encouragement and valuable guidance. He has been extremely supportive every time when I experienced down times in graduate school. Moreover, I would like particularly to thank Ehsan Kamalloo and Kory Mathewson for the great discussions. Our meetings have fostered insightful ideas and have been an integral part of my graduate experience at the Computing Science department.

During two years at the University of Alberta, I was fortunate to meet amazing friends who have shared with me their experiences and stories. More precisely, I would like to thank Roberto Vega, Megha Panda, Victor Nascimento, Melissa Woghiren, Chris Solinas, Saeed Sarabchi, Shrimanti Ghosh, Erick Ochoa, Douglas Rebstock, Mehran Mahmoudi, Sara Farazi, Thea Wang and Fatima Davelouis. It was great to have fun time and interesting discussions together.

I would like also to thank the professors Davood Rafiei and Greg Kondrak, who were part of the committee and who took the time to read my manuscript and assess my work.

Finally, I thank my lovely family for their continuous support, encouragement. This journey would not have been possible without them.

Contents

List of Tables

List of Figures

Acronyms

ADALINE Adaptive Linear Neuron. 9

Adam Adaptive Moment Estimation. 11

ADEM Automatic Dialogue Evaluation Model. 4

AI Artificial Intelligence. 1

ANA Automatic Nursing Agent. 2

BLEU Bilingual Evaluation Understudy. 35, 36, 59

CBOW Continuous Bag Of Words. 14

CPU Central Processing Unit. 19

EOS End of Sentence. 21

GloVe Global Vectors. 14, 51

GRU Gated Recurrent Unit. 19–21

HRED Hierarchical Recurrent Encoder Decoder. 30, 32, 56, 60, 64

IR-based Information Retrieval-based. 27

LCS Longest Common Subsequence. 36

LDA Latent Dirichlet Allocation. 6, 24, 25, 50, 53, 56, 58, 73

LM Language Models. 16, 18

LSA Latent Semantic Analysis. 24

LSTM Long Short Term Memory. 19–21

MDP Markov Decision Process. 34

MER Mean Evaluation Rating. 70

METEOR Metric For Evaluation Of Translation with Explicit Ordering. 35, 36

ML Machine Learning. 26

MLE Maximum-Likelihood Estimation. 3, 31

MMI Maximum Mutual Information. 31, 32

MSE Mean-Squared-Error. 11

NLM Neural Language Model. 16, 17

NLP Natural Language Processing. 13, 14, 28, 37, 38, 41, 44, 59

OOV Out of Vocabulary. 15

POMDP Partially Observable Markov Decision Process. 34

REI Response Echo Index. 63, 68

RNN Recurrent Neural Network. 17–21, 29, 30, 32, 43

ROUGE Recall-Oriented Understudy For Gisting Evaluation. 35, 36

SC Semantic Coherence. 60

Seq2Seq Sequence-to-Sequence. 3, 6, 20, 22–24, 28–33, 47, 49, 50, 56, 60, 63, 64

SMT Statistical Machine Translation. 28

SOS Start of Sentence. 20

TA-Seq2Seq Topic-Aware Sequence-to-Sequence. x, 32, 56, 60, 63, 65

tf-idf term frequency–inverse document frequency. 27

THRED Topical Hierarchical Recurrent Encoder Decoder. 6, 49, 50, 52, 55, 56, 59–64, 66, 67, 70, 73, 74

VHRED Latent Variable Hierarchical Recurrent Encoder Decoder. 32

Chapter 1

Introduction

Conversation plays a vital role in human life. It represents the most natural, powerful yet the most complex way of interacting verbally with each other. As human beings, we are very accustomed to the daily routines of conversing as a way of sharing opinions, exchanging ideas and expressing emotions. It eases the spread of knowledge and builds relationships between people. From the very young age, we develop this skill and we become habituated to this powerful ability to the extent that we take it for granted. Whether we are talking to our families and friends, booking a flight or ordering a pizza for lunch, we may deem this intuitive to have the capacity of understanding languages, but transferring this capability to machines has seemed an insurmountable hurdle for natural language researchers so far.

The attempt of generating conversations indistinguishable from human ones dates back to the early stages of Artificial Intelligence (AI) in 1950 where Alan Turing introduced an empirical test, known as the Turing Test [71], that tests the intelligence level of a machine. The machine tries to fool a human judge into believing that the generated response is indeed generated by a human. If the evaluator fails to distinguish between the machine's response and the human's response, the machine passes the Turing Test.

Ever since the Turing test was introduced, many researchers started following this path of investigation so as to mimic human behavior in generating fluent and engaging responses. Nowadays, chatbots are gaining popularity worldwide and big companies are increasingly investing millions of dollars to

build the most sophisticated agent. Amazon Alexa[1], Apple Siri[2], Google Assistant[3] and Microsoft Cortana[4] are all examples of popular conversational agents that perform a wide range of tasks such as calling a friend or setting up schedules and reminders.

Admittedly, the long-standing goal of human-machine interaction is to build an open-ended dialogue system that can conduct conversations about any topic. Having such system can ease people's lives in all age groups. For example, chatbots can provide support to the elderly people. In fact, loneliness and social isolation represent the main issues for aging people living at home. As families live apart, older people can become socially disconnected from their children and their friends and can experience depression. As a result, their mental health will decline and they will have poor quality of life. A dialogue agent would be capable of interacting verbally with seniors in an intelligent way, and thereby helping them with various daily tasks such as conversing with them, contacting relatives, reminding them to take their medication, etc. Some attempts have been made in this direction such as ANA [27]. ANA is an automatic nursing agent that serves as a companion for the elderly. It is capable of performing many tasks such as reminding the elderly of their events, send messages, create and update to-do lists, etc.

In the past three years, we have witnessed a revolution in the capability of computers to understand natural language text and to generate plausible responses to conversations. However, current dialogue systems still lag far behind human capabilities in terms of the conversations that they can perform. Modelling human-like behaviour that can respond smoothly and continuously, with no apparent gaps between dialogue turns is a challenging problem that researchers have been striving to solve. Indeed, sophisticated dialogue systems require a deep understanding of human languages ranging from morphology to semantic. Such systems should be able to recognize the variation and the structure of each word (e.g., plural versus singular). Apart from the word-level

[1] https://en.wikipedia.org/wiki/Amazon_Alexa
[2] https://www.apple.com/ca/ios/siri/
[3] https://assistant.google.com/intl/en_ca/
[4] https://www.microsoft.com/en-ca/windows/cortana

morphology, dialogue systems should be capable of understanding the meaning of individual words and generating syntactically correct sentences by grouping words together and by resolving ambiguities. Moreover, in the highest abstraction level, they should be able to account for the context in the conversation to generate context-wise and fluent responses. All of these requirements combine together to form a notoriously difficult task for computers to understand natural language text and to interact verbally with humans.

With the recent success of deep neural networks in natural language processing tasks such as machine translation [105] and language modelling [74], there has been growing research interest in building data-driven dialogue systems. Previous approaches rely on hand-crafted rules and they are often restricted to specific domains such as flight booking, restaurant reservation or technical support service [33], [133]. One major issue with these approaches is that they cannot scale up to new domains because manually encoding all features that a user might refer to in a conversation is extremely hard and time consuming [11]. In order to improve the robustness as well as the scalability of dialogue systems, attention has turned to learning conversational utterances from a gigantic amount of data. Fortunately, innovation in deep learning architectures and the availability of large public datasets have produced fertile ground for the data-driven approaches to become feasible and quite promising. In particular, Sequence-to-Sequence (Seq2Seq) neural networks model [105] has witnessed substantial breakthroughs in enhancing the performance of conversational agents from interpreting to generating natural language text. Such model succeeds in learning the backbone of the conversation but lacks any aptitude for producing context-sensitive and diverse conversations for the following reasons. First, Seq2Seq model conditions the prediction of the next utterance solely on the previous dialogue turn and thus, do not retain contextual information throughout conversation exchanges [96]. Second, the usage of the Maximum-Likelihood Estimation (MLE) as objective function within the Seq2Seq model is unsuitable and fails to teach it how to converse engagingly and interestingly [59]. Instead, the model tends to generate generic and commonplace responses like "*i'm not sure*" or " *i don't know*". Although these

responses are grammatically correct, they are dull and carry little information [59].

Instinctively, humans tend to adapt conversations to their interlocutor not only by looking at the last utterance but also by considering information and concepts covered in the conversation history [24]. Such adaptation increases the smoothness and engagement of the generated responses. We speculate that incorporating conversation history and topic information with our novel model and method will improve generated conversational responses.

In this work, we introduce a novel, fully data-driven, multi-turn response generation system intended to produce context-aware and diverse responses. Our model builds upon the basic Seq2Seq model by combining conversational data and external knowledge information trained through a hierarchical joint attention neural model. An important line of research that we also address in this work is automatically evaluating the quality of dialogue responses. Devising quantitative metrics allows rapid testing of dialogue models and reduces the burden of expensive human evaluation. Significant works have looked into this challenge. Examples include ADEM [65], an evaluation model that learns to score responses from an annotated dataset of human responses scores. Venkatesh et al. [111] proposed a number of metrics based on user experience, coherence, and topical diversity and have showed that these metrics can be used as a proxy for human evaluation. However, engagement and coherence metrics are estimated via recruiting evaluators. In this work, we propose to directly calculable approximations of human evaluation grounded in conversational theories of accommodation and affordance [24]. We show that such metrics conform reasonably well with human judgment, making a step towards a good automatic evaluation procedure.

Furthermore, the lack of good conversational corpora is an impediment to end-to-end dialogue generation systems mainly because no matter how effective a model is, as long as the input data is flawed, the system would produce absurd results[5]. To cope with this issue, we present a high-quality conversational dataset compiled from the Reddit data. We find that our method leads to both

[5]Originated from the classic "garbage in, garbage out" principle

diverse and contextual responses compared to the literature strong baselines. Our method has been shown to perform better when trained on the Reddit dataset compared to the OpenSubtitles dataset, a well-known existing corpus collected from movie scripts.

1.1 Thesis Statement

In this dissertation, we address two major issues related to building an open-ended dialogue system: diversity and context-awareness. We speculate that:

> *A dialogue system can more closely imitate human-level performance by learning a response generation not only from the response but also from the conversation history and the topics talked about in the conversation.*

To attain our objective, we explore deep learning techniques to foster a more sustained dialogue system. More specifically, we explore the widely used Seq2Seq approach by conditioning the responses on the context of the conversation and on external facts derived from a topical model. Doing so, we would have an interesting and an engaging chit-chat system which is able to generate responses that are not only topically diverse but also contextually relevant.

1.2 Thesis Contribution

In this dissertation, we will explore how to make the conversation more consistent, interesting, fluent and diverse. In summary, the key contributions of this work are as follows:

- We devise a fully data-driven neural conversational model that leverages conversation history and topic information in the response generation process through a hierarchical joint attention mechanism; making the dialogue more diverse and engaging.
- We evaluate the model quantitatively and qualitatively and we show that the introduced automated metrics correlate well with human judgment.

- We collect, parse and clean Reddit data to construct a high-quality conversational corpus.

1.3 Thesis Organization

The thesis is organized as follows: in Chapter 2, we provide a thorough background information about deep learning approaches for natural language processing. In particular, we start by explaining the basics of neural networks and then we move forward to discuss more advanced techniques such as Recurrent Neural Network and Seq2Seq architectures. We also give a detailed related work about the existing dialogue systems. In Chapter 3, we give an overview of current available datasets that are suitable for training data-driven dialogue systems. We detail how these corpora have been developed and we emphasize the impact of the corpus size and quality on the generated responses. Moreover, we introduce our collected conversational dataset from Reddit. Details about data collection, preprocessing steps will be provided too in this Chapter.

Chapter 4 is dedicated to explain in details our introduced model, Topical Hierarchical Recurrent Encoder Decoder (THRED). THRED extends the Seq2Seq model to condition the response generation on conversation history captured from previous utterances and on topic words acquired from a topical model. During encoding, our model maps dialogue utterances into hidden states vectors and acquires topic words from a pre-trained LDA model. Then, an utterance-level encoder is added on top of the word-level encoder to encode conversation history into a fixed-length vector representation. We further model the conversation history and the topic words using a two-level attention mechanism to enrich the response generation with topic words that are consistent to the context. In decoding, each word is generated through a joint attention mechanism and a modified generation probability that bias the model towards generating topic words.

In Chapter 5, we exhaustively evaluate our model quantitatively and qualitatively. In addition, we highlight the weaknesses in the existing automated metrics and we present novel metrics that are in-line with human judgment.

In particular, we introduce Semantic Coherence and Response Echo Index as good tools of better automated evaluation metrics for future dialogue system developments.

Finally, in Chapter 6, we summarize the results and the proposed contributions and we explore future work that has to be done in the direction of generating high-quality and informative conversational responses.

Chapter 2

Background And Related Work

In the past decades, conventional machine learning algorithms were shown limited in their ability to process natural language text in the raw form [55]. Much of the effort in deploying machine learning techniques were dedicated to the representation of raw data into suitable patterns (or features), from which the learning models would be capable of predicting the output or classifying input data samples. Such feature engineering is time consuming, labor intensive and requires reasonable domain expertise. It also highlights the weaknesses of traditional machine learning and demonstrates the need for human prior knowledge and ingenuity to design specific data representations. Consequently, the performance of machine learning methods depends heavily on the phase of data representation. To facilitate the applicability of machine learning models, attention has turned to deep learning techniques which have shown promising results by automatically learning representations from the data [55]. Automatically learned representations often outperform hand-crafted feature representation and adapt more easily to new tasks which lead to avoiding human intervention. In this chapter, we will review the technical background of deep learning techniques dedicated for natural language processing. Furthermore, the existing dialogue systems will be explored.

2.1 Artificial Neural Networks

Artificial Neural Networks, as the name "neural" suggests, are inspired by the biological neural networks, and are intended to mimic the way humans, we

speculate, learn. Such neural algorithms learn tasks automatically by looking into examples without being explicitly programmed. They learn by deriving meaning from unstructured data and by capturing high-level representations that are considered too complex for either humans or other computer techniques. The idea dates back to 1943 when McCulloch and Pitts [73] introduced a simplified model of the human neuron as a mathematical linear function that receives a set of n input values $\{x_1, ..., x_n\}$ and linearly transform them to an output y. This model learns a set of weights $\{w_1, ..., w_n\}$ and calculates the output $y = f(x, w) = x_1w_1 + \ldots + x_nw_n$. The McCulloch-Pitts neuron predicts two different groups of inputs by checking whether $f(x, w)$ is positive or negative.

In the following subsections, we will detail more advanced neural networks algorithms.

2.1.1 A Neuron

Neuron is a computational unit that takes as input a set of variables and produces a single output. It is the fundamental building block of neural networks. In the 1950s, the perceptron algorithm [90] was the first model that could learn the weights $\{w_1, \ldots, w_n\}$ given examples of inputs from each category. The model can be viewed as a binary classifier that maps its input x to a single binary value $f(x)$.

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b < 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

where $w \cdot x = \sum_{i=1}^{n} w_i x_i$ such that n denotes the number of input variables and b is a bias. In 1960, the Adaptive Linear Neuron (ADALINE) [121] was invented to improve the previous neural networks models by predicting a real number. More precisely, the neuron takes an n dimensional input vector x, associated with a weight vector w and a bias vector b. The output of the neuron is then:

$$f(x) = \sigma(\sum_{i=1}^{n} w_i x_i + b) \tag{2.2}$$

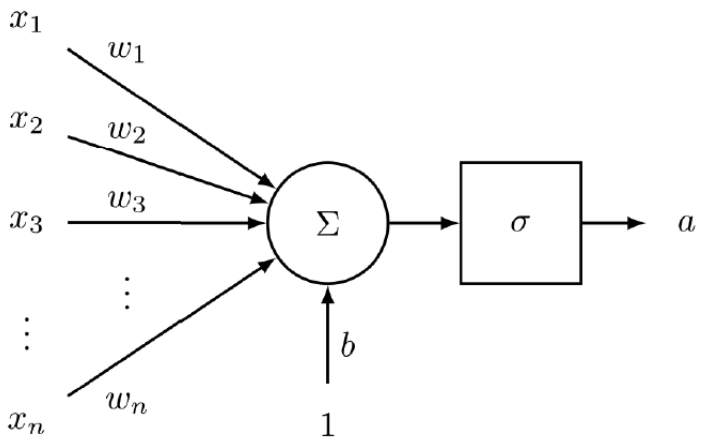


Figure 2.1: A neuron architecture

where σ is non-linear activation function. Therefore, $f(x)$ can be written as :

$$f(x) = \frac{1}{1 + exp(-(w^T x + b))} \tag{2.3}$$

Figure 2.1 delineates the architecture of the neuron and the visualization of the formulation mentioned above.

2.1.2 Feed Forward Neural Network

A feed forward neural network is a multi-layer network where the outputs from neurons in each layer are fed to the neurons in the next layer. The network has three types of layers: input layer, hidden layer and output layer. Each layer is fully connected, meaning that each layer takes as input all the outputs from the previous layer. Also, there is no link connecting units in the same layer. The units in the input layer are scalar values whereas the units in the hidden layer correspond to neural units, computing a weighted sum of their inputs and then applying a non-linear activation function. More formally, the output of the hidden layer is as the following:

$$h = f(Wx + b) \tag{2.4}$$

where f is a non-linear activation function (such as *tanh* or *sigmoid*), $x \in \mathbb{R}^{d_{in}}$ is a vector of real numbers representing the inputs with d_{in} the num-

ber of inputs; $b \in \mathbb{R}^{d_h}$ denotes the bias and $W \in \mathbb{R}^{d_h \times d_{in}}$ represents the weight matrix.

Afterwards, the output layer will compute a final output based on the representation value $h \in \mathbb{R}^{d_h}$. This output value can be a real number or probability distribution across the vocabulary words, it depends actually on the task that the network is going to achieve. Similarly to the hidden layer, the output layer has a weight matrix U and often does not have a bias vector. The network multiply weight matrix U by the hidden vector h to generate an output z as follows:

$$z = Uh \tag{2.5}$$

where $z \in \mathbb{R}^{d_{out}}$ with d_{out} is the number of output units and $U \in \mathbb{R}^{d_{out} \times d_h}$. If the task defines classification, the output cannot be a real-valued number but instead it should be a vector of probabilities. To this end, we should normalize the vector to a vector that ranges between 0 and 1 and sums to 1. A convenient function for such normalization is what we call **softmax** function. The softmax is defined as:

$$softmax(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{k} \exp(z_j)}, 1 \leq i \leq d_{out} \tag{2.6}$$

Training

The goal from training is to learn the optimal weights that minimize the distance between the model output $\hat{y}$ and the reference output y. A popular loss function is the Mean-Squared-Error (MSE) between $\hat{y}$ and y. For probabilistic classifiers, the common used loss function is the **negative log likelihood** J which ensures that maximal probability is assigned to correct answers and minimal probability is assigned to bad answers. The loss J is defined as follows:

$$J(\theta) = -\sum_{j=1}^{|V|} y_j \log \hat{y} \tag{2.7}$$

To find the minimum of the loss function, optimization methods such as the stochastic gradient descent [89] or Adam [49] could be employed. The intuition

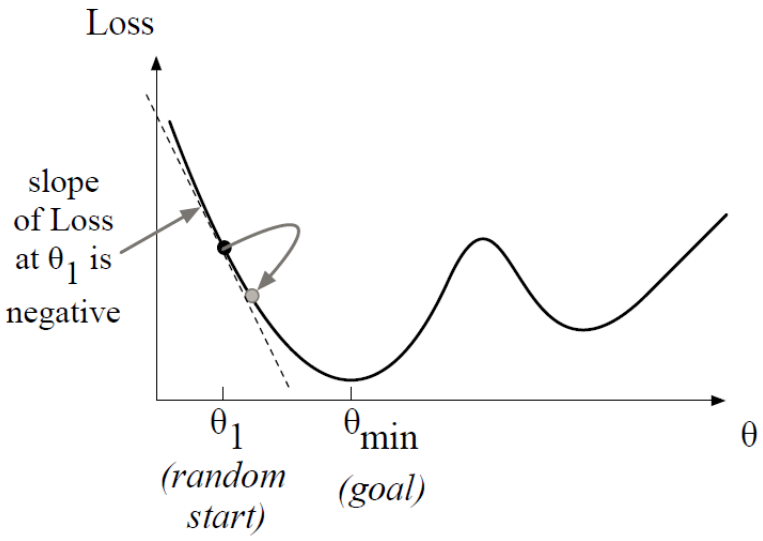


Figure 2.2: To find the minimum of the loss function, we move θ in the opposite direction from the slope of the function [44].

of these algorithms is to find the minimum of a function by identifying in which direction the function's slope is increasing the most steeply and then moving in the opposite direction [43] (See figure 2.2). To move the gradient towards the minimum value, the algorithm requires a learning rate η. η should be tuned carefully because if it is too small, the learning will take too long and if it is too large, the weight updates can over-shoot the minimum and diverge.

The model's parameters θ are thus updated as the following:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_{\theta^{(t)}} J(\theta^{(t)}) \tag{2.8}$$

The backpropagation algorithm [92] uses the chain rule of differentiation to compute the gradient by taking the partial derivative of the loss function $\nabla_{\theta^{(t)}} J(\theta^{(t)})$ with respect to each parameter in the model. The algorithm works as follows: (1) Propagate the input through the network. (2) Calculate the average of the overall loss for a chunk of data. (3) Compute gradients and update the output layer weights. (4) Propagate the error backwards and update the weights of the input and hidden layers (6) repeat with the next training data. If the loss function J is within tolerances, terminate. Otherwise, continue with an another epoch (i.e., a complete presentation of the dataset).

2.2 Deep Learning For Natural Language Processing

Natural Language Processing (NLP) is the analysis and use of human languages by a machine. It helps computers interact with humans by typically reading and generating natural text. As humans, we may speak and write in English, French, Arabic or other languages. Yet, such languages are predominantly incomprehensible to computers as their native language corresponds to millions of ones and zeros and not words. NLP offers an elegant way to fill the gap between human communication and computer understanding. Understanding human language is hard, we express ourselves in different ways making the conversation notably diverse and complex. Aside from the large number of existing languages, each language has its specific vocabulary, rules and grammar. Thanks to NLP techniques, it has become possible for computers to process speech, understand text, recognize and express emotions.
The big interest in human-to-machine communication has allowed the technology to rapidly progress. Many NLP applications are based on language models that compute a probability distribution over sequences of words and characters. Deep Learning techniques have been successfully applied to various NLP tasks ranging from speech processing to semantic interpretation [34], [38]. The most challenging tasks include:

- Machine Translation [4], [51], [83] (e.g., translating from English to French)
- Dialogue Generation [59], [60], [66], [101], [113]
- Semantic Analysis: corresponds to the study of the meaning of a query statement [26], [53], [54]
- Coreference Resolution: is the task of finding all expressions that correspond to the same entity in a text [56], [100]
- Summarization [28], [82], [93]
- Question Answering [3], [120], [127]

Arguably, to achieve excellent performance across NLP tasks, input words should be represented as dense vectors before feeding them to neural models. The process corresponds to learn **embeddings** for each target word. That is, mapping words to vectors of real numbers. Indeed, word vectors allow us to perform some notion of similarity (e.g., Jaccard similarity, Cosine distance, Euclidean distance, etc).

2.2.1 Word Embeddings

Word embeddings have been shown efficient in capturing semantic meaning and large number of precise word relationships that are useful for various NLP tasks. The idea of learning a distributed representation for words was first introduced by Bengio et al. [8], where they showed that word vectors delineate powerful representation for words. The method draws inspiration from the neural language models. The intuition is that words with similar meanings tend to occur near each other in the text and therefore they would have similar vector representations in the space. The neural model will learn an embedding by initializing the network by random vectors and then iteratively move them to resemble to embeddings of the proximity words. **Word2vec** [76] and **GloVe** [84] are among the most popular methods for such representation. Word2vec implements two approaches: Skip-Gram and Continuous Bag Of Words (CBOW). Both methods learn embeddings by training a feed-forward neural network to predict the surrounding words. Skip-gram works toward predicting the context words based on the center word. CBOW does the opposite by predicting the center word based on the neighboring words.
While word2vec method succeeds to capture complex semantic patterns, it fails to use the global co-occurrence statistics [84]. GloVe alleviates this issue by defining a weighted least squares model that trains on global word-word co-occurrence counts from a corpus, thus making use of the statistics values.

2.2.2 Language Models

Language Modelling consists of predicting the upcoming words giving the previous context words. It estimates the distribution of natural language text by

assigning probabilities to sequences of words. Such model would assign higher probabilities to sentences that are grammatically correct and highly frequent in text. For example, the following sentence "deep learning is part of a broader family of machine learning methods"[1] has a higher probability of appearing in a corpus than "part learning family of methods machine learning is deep a broader of". Formally, given a sentence $s = (w_1, \cdots, w_m)$ of length m, the language model defines a probability $P(s)$ by individually predicting each token within the sequence given all the previous words. Using the chain rule of probability, we compute the joint probability of a sequence $P(w_1, w_2, \ldots, w_m)$ as the following:

$$P(s) = P(w_1)P(w_2|w_1)P(w_3|w_1w_2)\cdots P(w_m|w_1\cdots w_{m-1}) \tag{2.9}$$

where $P(w_m|w_1, \ldots, w_{m-1})$ represents the probability of predicting the word w_m given all the preceding words $(w_1, w_2, \ldots, w_{m-1})$. However, the chain rule does not help in computing the conditional probability of a word given a long sequence of previous words. To alleviate this issue, n-gram model has been proposed to approximate the history of words by just the last N words. For example, a bigram model approximates the probability $P(w_m|w_1\cdots w_{m-1})$ by using the joint probability of only the proceeding word, leading to the Markov assumption [46]:

$$P(w_m|w_1\cdots w_{m-1}) \approx P(w_m|w_{m-1}) \tag{2.10}$$

We estimate the conditional probability $P(w_m|w_{m-1})$ by computing the count of the bigram w_mw_{m-1} and scale by the unigram count for the word w_{m-1}:

$$P(w_m|w_{m-1}) = \frac{C(w_mw_{m-1})}{C(w_{m-1})} \tag{2.11}$$

Some smoothing techniques have been applied for the task of modelling languages such as Laplace Smoothing [29], Backoff and Interpolation [103], Kneser-Ney Smoothing [50].

N-gram language modelling is straightforward and fast to implement but struggles with handling long term dependency and generalizing to unseen context and Out of Vocabulary (OOV) words [43].

[1] `https://en.wikipedia.org/wiki/Deep_learning`

2.2.3 Neural Language Models

Neural Language Model (NLM) comes as a remedy for Language Models (LM) as they offer the ability to handle much longer history and they are capable of managing the data sparsity without any need for smoothing techniques. NLM was first introduced by Bengio et al. [8] and was the basis of many models such as machine translation, summarization and dialogue systems. The paradigm represents a large-scale deep learning model that captures a long context via learning a distributed representation of words. However, such improved performance comes at the cost of a slower training process compared to the standard language model.

An NLM is a feedforward neural network having a moving window that goes through a text and takes as input at time t previous n words $(w_{t-1}, w_{t-2}, \cdots, w_1)$ and generates a probability distribution over potential next words. In other words, it approximates the probability of the next word given the prior words $P(w_t|w_{t-1}, \cdots, w_1)$. More precisely, NLM multiplies the concatenated word embeddings of the context $(e(w_{t-1}), e(w_{t-2}), \cdots, e(w_1))$ by a matrix W and adds a bias vector b and then passes the output through an activation function to produce a hidden layer h. The process is equivalently formulated as the following:

$$h_{t-1} = f(We + b) \tag{2.12}$$

where f is a non-linear function such as tanh, $W \in \mathbb{R}^{V \times K}$ and $b \in \mathbb{R}^V$; V is the vocabulary size and K is the word embedding size.

The hidden layer h is then multiplied by another weight matrix U as follows:

$$z_t = Uh \tag{2.13}$$

To generate the conditional probability distribution of the next word w_t, a softmax layer is added atop the hidden layer. The softmax maps the scalar vector z into a vector of probability distribution:

$$P(w_t|w_{t-1}, \cdots, w_1) = \frac{\exp(z_t)}{\sum_{z \in V} \exp(z)} \tag{2.14}$$

In summary, NLM represents an elegant alternative for traditional LM as they have the ability to tackle the data sparsity issue and to account for more context words. However, with the increase of the window size n, the memory requirements of the system grows exponentially, making training a large model practically impossible [74].

2.2.4 Recurrent Neural Networks

As opposed to NLM, Recurrent Neural Networks (RNN) are capable of conditioning the next word on much longer context words. They were particularly introduced to handle sequential data and have been shown successful in addressing a variety of natural language processing tasks [74], [75], [77]. An RNN takes as input a sequence of words $(w_1, \cdots, w_n)$, where each token is associated with the corresponding word embedding representation x_t. Figure 2.3 delineates the RNN architecture. At each time step, RNN maps each word to a hidden vector representation h_t, which summarizes information of all previous words. Precisely, the output of the previous hidden state h_{t-1} along with the next word embedding x_t are fed into the hidden layer to generate a prediction output $\hat{y}$:

$$h_t = f(W_{hh} \cdot h_{t-1} + W_{hx} \cdot x_t) \tag{2.15}$$

$$\hat{y_t} = softmax(W_s \cdot h_t) \tag{2.16}$$

Below, we will detail all parameters mentioned in the previous equations:

- $x_t \in \mathbb{R}^K$ with K the dimension of the word embedding vector
- $W_{hh} \in \mathbb{R}^{D_h \times D_h}$ is the weights matrix that is employed to condition the output of the previous hidden state h_{t-1}
- $W_{hx} \in \mathbb{R}^{D_h \times K}$ is the weights matrix that is used to condition the input word vector x_t
- $h_{t-1} \in \mathbb{R}^{D_h}$ is the previous hidden state.
- f is a non-linear activation function. The famous choices for f are $sigmoid$ and $ReLU$ functions.

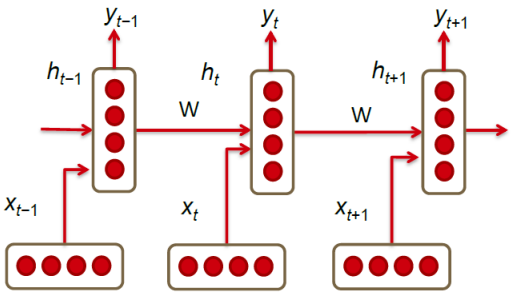


Figure 2.3: A Recurrent Neural Neural Network (RNN) architecture derived from `http://cs224d.stanford.edu/`

.

To sum up, RNNs offer many advantages over standard LM:

1. Unlike standard LM, the RNN model size does not increase for longer input. While the size of W_{hh} could be very large, it does not increase with the size of the corpus.

2. Theoretically computation for step time t can use information from many steps back.

3. Weights are shared across time steps which means that sentences representations are shared

2.2.5 Gated Recurrent Unit & Long Short Term Memory

Despite the promising advantages, training RNNs is difficult and accessing information from too many steps back is practically unfeasible for the following reasons. Backpropagation algorithm allows RNNs to propagate weight matrices from one step to the next. However, for long sentences, the gradient values gradually vanish, if the weights are small, when the training loss is back-propagated over few time-steps. This issue is called the *Vanishing Gradient Problem*. Another issue with RNNs is the *Gradient Explosion Problem*, where the gradient values grow extremely large, if the weights are big, during backpropagating the error over time. These two problems deteriorate the learning quality of the model for far-away words. Typical feed-forward neural

nets can cope with these effects because they only have a few hidden layers. Nonetheless, in an RNN trained on long sequences (e.g., 150 sequences), the gradients can easily explode or vanish.

To remedy to these major shortcomings, two architectures have been proposed: Long Short Term Memory (LSTM) [35] and Gated Recurrent Unit (GRU) [22]. The key idea of GRU and LSTM is to map each time step to different types of gates. These gates are carefully designed to avoid long-term dependency problem and to control the flow of the information. LSTM was first introduced by Hochreiter and Schmidhuber in 1997 [35] and further studied by many other works ([18], [22], [42], [47], [106], [128]). The following equations are the mathematical formulation of the LSTM units:

$$
\begin{aligned}
i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{t-1}) && \text{(Input gate)} \\
f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_{t-1}) && \text{(Forget gate)} \\
o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{t-1}) && \text{(Output gate)} \\
\widetilde{c}_t &= tanh(W^{(c)}x_t + U^{(c)}h_{t-1}) && \text{(New memory cell)} \\
c_t &= f_t \circ c_{t-1} + i_t \circ \widetilde{c}_t && \text{(Final memory cell)} \\
h_t &= o_t \circ tanh(c_t)
\end{aligned}
$$

The input gate employs the input word and the past hidden state to check whether or not the input is worth preserving. As for the forget gate, it is quite similar to the input gate except it does not determine the importance of the input words with regard to the generation of the next word, instead it assesses the importance of the past memory cell for the computation of the current memory cell. The output gate determines the important parts of the memory c_t which needs to be present in the hidden state h_t.

GRU [22] was introduced recently (2014) and it was found that it outperforms LSTM architecture both in terms of convergence in CPU time and in terms of parameter updates and generalization. It was designed in a way to have more persistent memory, thus capturing longer context information. Unlike LSTM, it has four fundamental stages: update gate, reset gate, new

memory and hidden state:

$$
\begin{aligned}
z_t &= \sigma(W^{(z)}x_t + U^{(z)}h_{t-1}) && \text{(Input gate)} \\
r_t &= \sigma(W^{(r)}x_t + U^{(r)}h_{t-1}) && \text{(Forget gate)} \\
\widetilde{h_t} &= \tanh(r_t \circ Uh_{t-1} + Wx_{t-1}) && \text{(New memory)} \\
h_t &= (1 - z_t) \circ \widetilde{h_t} + z_t \circ h_{t-1} && \text{(Hidden state)}
\end{aligned}
$$

2.2.6 Generative Sequence-to-Sequence models

A Sequence-to-Sequence (Seq2Seq) is a relatively new paradigm. It was first introduced by Sutskever et al. [105] in 2014 and succeeded to achieve a good performance in the task of machine translation ([4], [14], [48], [67]–[69], [94], [124], [125]). Furthermore, it has achieved breakthrough progress in other natural language generation tasks such as parsing ([67], [112]), text summarization ([20], [82], [93], [134]) and dialogue generation ([59], [96], [113]). The Seq2Seq model can be perceived as an extension of a language model where it is composed of two RNNs: an encoder RNN and a decoder RNN.

- **Encoder**: which takes the input message and encodes it into a fixed-length vector representation, also called a *context* vector **c**.
- **Decoder**: which uses the *context* vector as a trigger from which it generates the next sentence given the previous utterance.

Figure 2.4 represents the overall architecture of the Seq2Seq model. More precisely, the encoder reads the input words sequentially and encodes them into a fixed-length vector representation **c**. To achieve this, the encoder employs a series of LSTMs or GRUs layers where each layer reads one token at a time. The final layer generates the context vector c. The decoder consists also of an LSTM/GRU network. As a first step, we initialize the first hidden state with the context vector c. Afterwards, we feed the network a special Start of Sentence (SOS) token, namely $<SOS>$, to trigger the start of the output

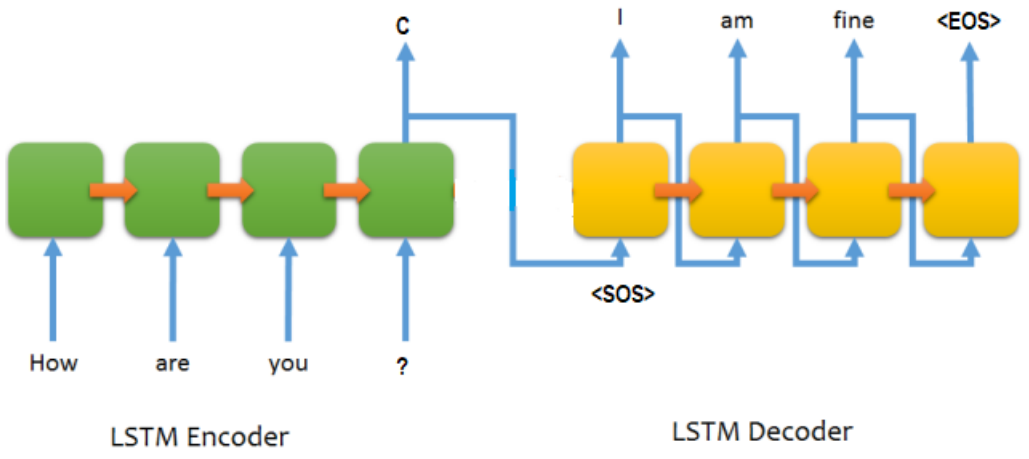


Figure 2.4: A seq2seq architecture showcasing the task of response generation where the message is "How are you ?" and the response is "I am fine"

generation. The decoder stops generating words once it encounters the End of Sentence (EOS) $<EOS>$.

More formally, given an input message $X = \{x_1, x_2, ..., x_n\}$ and an output response $Y = \{y_1, y_2, ..., y_{n'}\}$ where each x_i and y_i represents a word, the Seq2Seq model maximizes the probability of generating the target answer Y given the source sentence X:

$$P(y_1, .., y_{n'}|x_1, .., x_n) = P\left(y_1|c\right) \prod_{t=2}^{n'} P(y_i|c, y_1, .., y_{t-1}) \tag{2.17}$$

At each time step, the encoder reads a word and updates its hidden state h_t:

$$h_t = f(h_{t-1}, e_t) \tag{2.18}$$

where

- f is a parametrized non-linear function which can correspond to the sigmoid, the GRU or the LSTM.
- e_t is an embedding vector representation for an individual text token x_t.

During decoding, the *context* vector is fed into the decoder RNN to generate a probability distribution of the next word in the sentence at every time step.

The probability generation is computed using a softmax function:

$$P(Y|X) = \prod_{t=1}^{n'} \frac{\exp(f(h_{t-1}, e_{y_t}))}{\sum_{y'} \exp(f(h_{t-1}, e_{y'}))} \tag{2.19}$$

Training

To train the Seq2Seq model, we need a dataset where each source sentence x is aligned with a target sentence y. The goal from the training is to optimize the objective function $P(Y|X)$ so the output sentence for each training instance learns to be as close as possible to the corresponding ground-truth target sentence. The learning objective corresponds to minimize the negative log-likelihood of generating the next word in the target sentence y given all the previous words in the source message x:

$$J = -\sum_{t=1}^{t=N_y} log(y_t|c, y_1, y_2, \cdots, y_{t-1}) \tag{2.20}$$

Testing

During testing, the pre-trained model generates a response sentence y given an unseen input sentence x. Two popular search algorithms are usually used to produce a sequence of words with the largest probability: greedy search and beam search.

Greedy search: consists of feeding the most likely word predicted at the previous step x_t to the next step. In other words:

$$x_t = argmax P(x_t|x_1, \cdots, x_n) \tag{2.21}$$

While this technique seems efficient and easy to implement, the response might be far from optimal. This is because a small part of the search space has been explored. Furthermore, if the decoder chooses the incorrect word at one time step, the rest of the sentence will be impacted.

On the other hand, decoding a word sequence with the highest probability involves searching the space through all the possible output sequences based

on their likelihood. Considering all the possibilities is unfeasible because the search problem is exponential in the length of the output sequence. To avoid this issue, we consider a window of words. This technique is called beam search.

Beam search: the idea lies in maintaining K candidates at each time step. As we move forward in time, the decoder expands each of the K candidates represented as $Y_{t-1}^k = \{y_1^k, \cdots, y_{t-1}^k\}$ with $k \in [1, K]$. The process is done by curating the most probable K candidates. Doing so, the model has to consider $K \times K$ new hypothesis. At the end, the top ranked K hypothesis are selected from the $K \times K$ hypothesis computed previously.
However, beam search algorithm can only explore a small number of candidates in the search space [61]. Also, increasing beam width can result in generating inconsistent responses and most of the generated sequences would look very similar to each other. To address these weaknesses, some works were suggested such as [61] and [98] to foster diversity in neural generation. The value of K is tuned by experiments.

2.2.7 Attention Mechanism

The traditional Seq2Seq model struggles with generating consistent and coherent responses because the encoder fails to keep track of long-term dependency in its fixed-length final hidden state vector. This limitation constitutes the bottleneck problem in Seq2Seq models. An effective way to tackle such an issue is to use the Attention mechanism ([4], [21], [41], [79], [99]).
The attention mechanism can be viewed as a technique that connects the current decoding process with each input time-step as a way to attend to important words in the input that are most responsible for the current decoding time-step.

More formally, during decoding, we compute the hidden states s_i using the following recursive formula:

$$s_i = f(s_{i-1}, y_{i-1}, c_i) \tag{2.22}$$

where s_{i-1} is the previous hidden vector, y_{i-1} is the predicted word in the pre-

vious time-step and c_i is the context vector that captures relevant information in the input message for the i^{th} decoding step.
Let $h_1, \cdots, h_n$ be the hidden states vectors that represent the input sentence. For each hidden vector, we calculate a score as follows:

$$e_{i,j} = \eta(s_{i-1}, h_j) \tag{2.23}$$

where η is a multi-layer perceptron. Afterwards, we scale up the scores $(e_{i,1}, \cdots, e_{i,n})$ into a vector $\alpha_i = (\alpha_{i,1}, \cdots, \alpha_{i,n})$ by a softmax function:

$$\alpha_{i,j} = \frac{exp(e_{i,j})}{\sum_{k=1}^{n} exp(e_{i,k})} \tag{2.24}$$

Then, we compute the context vector c_i as the following:

$$c_i = \sum_{j=1}^{n} \alpha_{i,j} h_j \tag{2.25}$$

2.3 Topic Modelling

To deal with the issue of diversity and dullness of the dialogue systems, we incorporate topic words into the Seq2Seq model as explained later in Chapter 4 in Section 4.1. Topic words are acquired via a Latent Dirichlet Allocation (LDA) model.
Topic modelling is an unsupervised machine learning technique that predicts sets of topics from a large collection of documents. Several probabilistic topic modelling methods have been successfully applied for tasks like document classification [110], information retrieval [110] and dialogue generation ([72], [78], [126]). LDA is the most popular topical model [9] and it is more accurate than other topic modelling techniques such as Latent Semantic Analysis (LSA) [9]. In the following subsection, we briefly overview the LDA model.

Latent Dirichlet Allocation

In LDA, the main objective is to assign topics, characterized by a distribution over words, to documents. Topics are captured through the following probabilistic generative process wherein documents are identified as a mixture of

topics. Let K denote the number of topics. Each topic k is picked following a Dirichlet distribution (i.e., $\phi_k \sim \text{Dir}(\beta)$). Each document d is further generated by acquiring its topics from a Dirichlet distribution (i.e., $\theta_d \sim \text{Dir}(\alpha)$). Then, each word $w_{d,i}$ in the document is sampled based on a Multinomial probability conditioned on the topic $z_{d,i}$ where $z_{d,i}$ is drawn from a Multinomial distribution over topic weights θ_d.

Using the Collapsed Gibbs sampling [85], LDA draws the topic distribution from representations of each document. Initially, each word w in each document d is assigned randomly to one of the K topics. Then, the algorithm modifies the topic mapping by approximating the posterior probabilities $p(k|d)$ and $p(w|k)$, with $p(k|d)$ being the probability of words in document d that are currently assigned to topic k, and $p(w|k)$ being the probability of assignments to topic k over all documents that come from this word w. This process continues iteratively until the assignment reaches a steady state.

2.4 Overview Of Existing Dialogue Systems

Dialogue systems can be roughly categorized into two groups: non-task-oriented dialogue systems (a.k.a chatbots or open-ended dialogue systems) and task-oriented dialogue systems. Although both approaches do have goals, the performance measure of the task-oriented dialogue systems is well-defined since it depends on the accomplishment of the task at the end of the conversation. In this work, following on the footsteps of several researchers in deep learning and natural language processing, we have tried to move out of the mold of dealing with only highly structured dialogue tasks in order to cope with open-ended conversational agent. That is, a dialogue system that can converse fluently and engagingly with humans about any topic. In the following subsections, we will give an overview of the existing dialogue systems and we will explain how researchers are striving to alleviate their setbacks.

2.4.1 Chatbot systems

Chatbots, called also chit-chat systems, are dialogue systems that are designed to mimic human-behaviour by conversing coherently and engagingly with humans on a range of different events and topics. They focus typically on interacting verbally with humans on open domains. Generally, three main approaches have been adopted for chit-chat systems: Rule-Based systems, Information-Retrieval-Based systems and Generative systems.

Rule-Based systems

The chatbot generates a response based on hand-crafted rules engineered by humans. The system matches the message to one of the pre-defined list of rules based on simple pattern matching, if-else conditions or more advanced Machine Learning (ML) techniques. ELIZA [116] is one of the most successful rule-based chatbot that dates back to 1966 where it was designed to influence people's lives in a positive way and especially those who suffer from psychological issues. Consequently, it aids doctors in diagnosing patients' condition and working on their treatment. ELIZA begins processing user's utterances by searching for a keyword that occurs in a predefined dictionary. If the keyword is found, the utterance is mapped to the rule that transforms the statement into a response. Otherwise, ELIZA outputs a generic response such as "*I see*", "*Please go on*", or "*that's very interesting*" or uses an utterance from the conversation history. Few years later, PARRY chatbot [23] appeared in 1971 with a similar psychological focus as ELIZA but with the aim of investigating schizophrenia. PARRY built upon ELIZA by adding an attitude to the bot like fear and anger. Unlike ELIZA, PARRY has an additional emotional state that controls the response generation process. If the human's utterance expresses anger for example, PARRY would choose to output a response from a predefined set of hostile responses. While these approaches may seem effective and promising, they fail to generate an appropriate response in most of the cases. Rule-based systems are not scalable and cannot interpret human language, the responses are based on some hand-crafted rules that sound un-

natural and most importantly do not account for contextual information in the conversation.

Information-Retrieval-based systems

Given the user's message, Information Retrieval-based (IR-based) chatbots rely on choosing a response from a corpus of unstructured conversational text using any information retrieval algorithm [39], [58], [129], [130]. Formally, IR-based systems take as input a user's query **q**, and a conversational corpus **c** and return a response **r** that is relevant to **q**. Therefore, the task can be defined as ranking a repository of responses to find the most suitable response. The retrieval process can be done by scoring utterances in **c** using any similarity function (e.g., cosine similarity between **q** and **r** by employing tf-idf or word embedding). Non-dialogue text can also be used to extract responses based on the messages (e.g., COBOT chatbot [37]). Although IR-based systems generate always grammatical responses (because responses are taken from the training dataset), they fail to generate diverse responses and most importantly they fail to handle the context of the conversation of natural language. Moreover, they lack the ability to distinguish between the semantics of different words. This is why, researchers have turned their attention to neural generative dialogue systems.

Neural Generative Dialogue Systems

Neural Generative dialogue systems generate utterances word by word producing natural sounding sentences that could have never appeared in the training dataset; as opposed to IR-based dialogue systems which copy an utterance from the corpus and send it to the user. The availability of a large amount of conversational data such as movie scripts and social media websites has opened the gate to many researchers to train and build data-driven dialogue systems. A response generation process can be deemed a message-response mapping problem where the model has to learn a coherent response given previous message utterances.

This path of investigation was first initiated by Ritter et al. [88] where they

model the task of generating dialogue responses as a phrase-based Statistical Machine Translation (SMT) problem. While this approach seemed promising, it has a potential problem. The responses are not semantically aligned with the posts as in the problem of machine translation. Moreover, the wide range of plausible responses make generating conversational responses dramatically more arduous than translating between different languages. Luckily, the recent success of deep learning methods in various NLP tasks has spurred research to investigate further end-to-end dialogue models.

End-to-End Dialogue Systems: Neural dialogue models are often dubbed end-to-end dialogue systems for the following reasons: First, they do not require to learn any sub-components such as Dialogue State Tracker or Natural Language Generator [95]. This is in contrast to the traditional dialogue systems [132] (architecture shown in Figure 2.5), where each component needs to be trained separately and to maximize an intermediate objective (e.g., training the State Tracker component to minimize the cross-entropy error of predicting the slot-values). Thus, end-to-end dialogue systems do not need labeled dataset about the user intention or the dialogue state labels. Second, they are trained to optimize a single objective function through a conversational dataset. More specifically, they maximize the log-likelihood of the generated utterance conditioned on the conversation history [113]. In summary, end-to-end dialogue systems do not demand human feature engineering. Instead, all the standard dialogue components are learned directly from human-human dialogues. Additionally, they are not restricted to a specific domain, as it is the case for traditional systems, and they generalize to open-ended conversations.

Achieving human-level performance with dialog systems requires both personalization and accounting for contextual information discussed throughout the conversation ([60], [96]). In other words, dialog agents should be able to not only adapt the response to the conversation history but also to the speaker's background, personal information, speaking style, etc. In an attempt to ameliorate the quality of the response generation process, many neural generative models have been suggested. Vinyals et al. [113] made use of the Seq2Seq

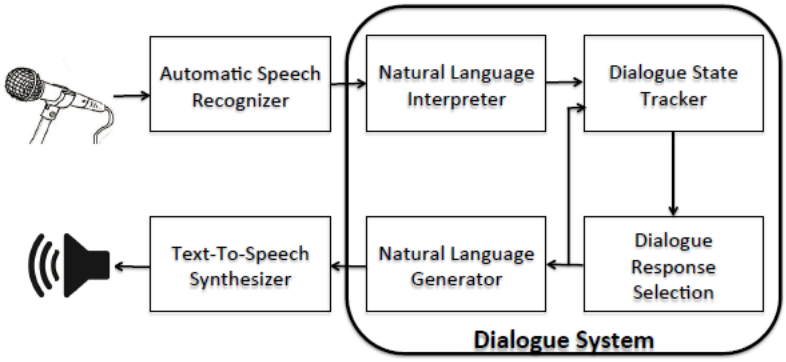


Figure 2.5: Architecture of a traditional dialogue system [95].

model proposed in [105] by conditioning the response on the previous input message. As opposed to conventional dialogue systems which typically require a lot of domain-specific handcrafting rules, their architecture is data-driven and end-to-end. The model is able to generate basic and coherent sentences by relying on learning the structure of the sentences from a gigantic open-domain dataset. Despite the enormous success achieved by the Seq2Seq model in generating grammatically structured responses, the model fails to output utterances that are sensitive to the context of the conversation.

Sordoni et al. [101] argued that constructing active and engaging dialogue systems requires taking into account previous utterances in the conversation. The dialogue system is trained on Twitter conversations in which the response generation is conditioned on past dialog utterances that provide contextual information. The dataset consists of short dialog conversations composed of triples (c,m,r) consisting respectively of three sentences: context, message and response. The authors built two context-sensitive response generation models. In the first model, they used a simple approach that concatenates c and m and computes a single bag-of-words representation which is fed into a multi-layer neural network to produce a fixed-length representation. Thereafter, they feed the resulting vector into an RNN [74] to generate the response. While this simple approach seems promising, it underestimates the dependency that exists between m and r and does not distinguish between c and m. To address the problem, they concatenated the two bag-of-words representations of c and m and fed it into the RNN. This way, they have an order-sensitive

representations of both message and context.

An issue with these approaches is that if the context encompasses multiple dialog turns, the concatenated sentences will be very long on average and RNN cannot keep track of such long-range dependencies. As a result, most of the dialogue context is lost. Serban et al. [96] introduced a Hierarchical Recurrent Encoder Decoder (HRED) neural network to further alleviate this issue. Precisely, the model captures dependencies over a three-turn conversation history introducing an additional context RNN on top of the RNN encoder. It is an improvement over the standard Seq2Seq model [105] that conditions the prediction of the next dialogue turn on all the previous utterances in the dialogue. The key idea underlying this approach is to decompose a dialogue into two-level hierarchy: the first level encoder RNN maps each utterance to a fixed-length vector representation. More precisely, the encoder takes as input each word embedding representation in the utterance and updates its recurrent states. The final state in the encoder RNN can be viewed as an order-sensitive summary of all the information processed up to the final token. The second-level context RNN processes iteratively each resulted utterance vector, representing a summary of the conversation up to that dialogue turn. At this point, the decoder RNN takes as input the context-level recurrent states and generates a context-aware response. As opposed to the standard Seq2Seq model, the context RNN state is updated only every utterance in the dialogue. Consequently, deploying an additional context RNN on top of the encoder RNN forms a hierarchy of RNNs that captures long history context and generates more context-wise responses. HRED has shown promising results in generating content-wise responses that take into account the context in previous dialogue utterances.

Neural generative models have been improved through several techniques:

Diversity: Recent works have demonstrated that it is possible to train conversational model on an end-to-end and completely data-driven fashion. But, these approaches generate short-sighted and generic responses like "*I don't know what you're talking about*", "*I'm not sure* " or "*I'm OK*" making the con-

versation neither engaging nor diverse. Having such generic behaviour could be explained by the distribution of the words in the conversational dataset where trivial phrases tend to have a high frequency, dominating the dataset. However, informative sentences tend to be relatively sparse. Serban et al. [95] argued that the problem also lies in the fact that conversations are naturally multi-modal and ambiguous, which force the model to fall back on generic responses. Li et al. [59] noticed that diverse and interesting sentences can be found in the N-best candidate list but may rank at the bottom. This is caused by the objective function MLE that tend to give higher probability to "safe responses". Training on such unbalanced distribution, the Seq2Seq model fails to represent the semantic of sentences in a good way. Instead of solely conditioning the response on the preceding utterance, Li et al. suggested capturing the dependency of the responses based on the previous messages and vice versa. Therefore, they used the MMI [5], [15] as an objective function rather than the traditional MLE, to grasp the mutual information between inputs and outputs. They generalized the MMI by introducing a hyper-parameter that controls how much to penalize generic responses as the following:

$$P(y_1, .., y_{n'}|x_1, .., x_n) = argmax(1 - \lambda)logP(y|x) + \lambda logP(x|y) \tag{2.26}$$

Nonetheless, such strategy leads to generate ungrammatical responses during decoding. More precisely, the second term $logP(x|y)$ is not computationally feasible during decoding since the model needs the target to predict the source. To mitigate this issue, they employed an approximation approach in which they generate initially N-best lists responses using the standard MLE objective function and then re-rank the generated responses using the second term of the MMI equation. This technique secures generating syntactically correct sentences since the Seq2Seq models typically output well-formed structure. Therefore, the method can be formalized as a scoring function that tries to re-rank utterances according to the dependency of targets on sources. While this approach might help producing diverse responses, it relies heavily on the aptitude of the traditional objective function MLE to generate satisfactory

diverse responses. overall, studies on both automatic evaluation and human judgment show that using MMI yield better responses.

Another significant work that deals with the diversity issue in the response generation process is the VHRED model, ascribed in [97]. The authors propose a model that expands upon the HRED model by adding an extra component: a high dimensional stochastic latent variable at every dialogue utterance. This model tries to mitigate the "*shallow*" sequential generation issue that previous approaches had. They called it shallow generation process because the model has to generate at every time step all high-structure representation of previous conversation history. The traditional Seq2Seq model struggles with generating consistent and coherent responses since the decoder has to keep track of previous information in its fixed-length hidden state vector. Particularly, the model is very likely to favor generating short-term predictions to long-term predictions specially when it has to encode sequences with high-variability. To address these problems, VHRED uses alongside the context vector, obtained from the context RNN layer, a multivariate Gaussian variable as input to the decoder RNN. The authors argue that the latent variable facilitates encoding long context and allows modelling ambiguity and uncertainty in the dialogue. Consequently, it helps generating more diverse responses.
HRED and VHRED focus mainly on capturing conversation information by modeling the hierarchy of the context. However, they do not investigate how to attend to important words that are crucial for generating plausible responses. Ignoring this step may result in losing important information in context and producing irrelevant responses.

Xing et al. [126] introduced the Topic-Aware Sequence-to-Sequence (TA-Seq2Seq) model which targets generating diverse responses by augmenting the content of utterances by topics. It is based on the basic Seq2Seq model and brings topic information using a joint attention mechanism and a biased generation probability. The intuition behind this idea is that in conversations, people often relate a response to concepts in their mind. Thus, making the conversation more content rich and informative. The topics are obtained using a Twitter-LDA model [137] which is trained on an enormous social media

dataset.

Shao et al. [98] addressed further producing diverse responses. They performed a minimal change over the Seq2Seq model by fixing the length of the decoder. They show that such slight modification yields noticeable improvement in the quality of the responses. They also introduced a stochastic decoding with Segment-by-Segment reranking. Rather than selecting the top K candidates as in the standard beam search, the new decoding method select the candidates by a stochastic sampling procedure to incorporate variation. Li et al. [62] used deep reinforcement learning to generate highly-rewarded responses by considering three dialogue properties: ease of answering, informativeness and coherence.

Personalization: Li et al. [60] addressed the challenge of personalizing the dialogue system by modeling human-like behaviour. They presented a persona-based model that aims at handling the speaker consistency by integrating a speaker-level vector representation into the decoder part of the Seq2Seq model. According to the authors, a PERSONA vector encodes information that captures human characteristics such as age, gender, speaking style, etc. They presented two PERSONA models namely SPEAKER and SPEAKER-ADDRESSEE model. The former approach consists of a seq2seq model that integrates a speaker-level vector representation into the encoder part of the seq2seq model. Correspondingly, the latter model encodes the interaction between two individuals by building an embedding representation based on their utterances. Zhang et al. [135] proposed a method that makes the dialogue more engaging by conditioning on profile information. They also introduced the PERSONA-CHAT dataset and show that models trained on it are more engaging and interesting.

In this work, we focus on comparing the proposed model (explained in Chapter 4) against Seq2Seq, HRED and TA-Seq2Seq. We do not compare against VHRED because the results achieved do not present a substantial improvement over the ones achieved by HRED. However, we compare against HRED because it captures the contextual information through the additional context layer added on top of the encoder RNN. Moreover, we compare against

TA-Se2Seq model because it biases the probability distribution towards leveraging topic words in the responses. We do not focus on personalizing the responses and therefore we do not evaluate our model against the PERSONA model [60].

2.4.2 Task-oriented dialogue systems

Task-oriented dialogue systems correspond to systems in which a specific task should be accomplished at the end of the conversation. These systems are usually designed to get the message from the user and accomplish a specific task within a limited number of dialogue exchanges [12].

The modern task-oriented dialogue systems are usually based on frames or domain ontology [43], called the frame-based systems. They were first proposed by Borrow et al. [10] in 1977 for travel planning. Modelling dialogues is guided essentially by the frames, which control the information at different stages of the conversation. A simple frame-based system corresponds to a finite-state machine that asks the user a list of questions based on the frames and continue to the next question as long as the user provides an answer. The problem with such systems is that they are unable to decide the state of the conversation (e.g., the user asks a clarification question or rejects a suggestion). Therefore, these systems struggle with taking an action based on the conversation progress. To tackle these issues, significant works have been proposed. Examples include State-Based dialogue systems ([87], [104], [115]) where the dialogue modelling is based on two concepts: Dialogue States and Dialogue Acts. The former indicates the progress of the conversation (e.g., intentions of the speakers, context history, etc.) and the latter indicate the category of an utterance. The key idea behind State-Based dialogue systems is to map a state to the corresponding act. The problem can be formulated as learning the optimal mapping to maximize the conversation success.

To this end, reinforcement learning techniques such as MDP or POMDP ([57], [123], [131]) have been widely used to learn such mappings. Recently, advances in neural networks have pushed the boundaries of task-oriented dialogue systems to become more consistent and successful ([81], [117]–[119]).

2.4.3 Evaluation Metrics

A challenging task of building conversational agents lies in evaluating the quality of their responses. Automatically evaluating dialogue systems has seemed notoriously hard. Typically, evaluating goal-oriented dialogue systems is done via human-generated judgment like a task completion test or user satisfaction score ([80], [114]). However, evaluating open-ended dialogue systems is still an open problem that has been receiving recently increased attention. Recent works in response generation have adopted the BLEU metric [83] from the machine translation task, the ROUGE metric [63] from the automatic summarization field. Following these metrics, researcher have proposed the METEOR metric [7] as an improved version of BLEU.

BLEU: The BLEU metric [83] was proposed by IBM researchers in 2002 and was one of the most reliable evaluation methods for translating between different languages. It is the most common automatic metric used in dialogue response generation. It uses n-gram matches to see how much the translated response resembles the ground truth. More specifically, it computes a precision score to evaluate the strength of the match. Afterwards, the algorithm penalizes the BLEU score by a brevity penalty so that small sentences with precision 1.0 would not be deemed good translations. More formally, let k be the maximum n-gram that one would like to evaluate the BLEU score on. The precision score is defined as follows:

$$P_n = \frac{\text{number of matched } n\text{-grams}}{\text{number of } n\text{-grams in the candidate translated response}} \tag{2.27}$$

The brevity penalty is defined as follows:

$$\beta = e^{\min(0,1-\frac{L_{gt}}{L_{mt}})} \tag{2.28}$$

where L_{gt} represents the length of the ground truth translation and L_{mt} represents the length the machine translation response. Finally, a geometric weighting $w_n = \frac{1}{2^n}$ is computed for the precision of the n^{th} gram. Thus, the BLEU score is formulated as:

$$\text{BLEU} = \beta \prod_{i=1}^{k} P_i^{w_n} \tag{2.29}$$

The score turned out to correlate well with human judgment for the machine translation task. However, it does have many drawbacks. A zero precision score will zero the whole BLEU score. Moreover, comparing a machine translation response with only a single reference translation is not sufficient for representing the matched n-grams.

METEOR: The METEOR metric [7] has been proposed to address several shortcomings with the BLEU metric. As opposed to considering n-gram overlap, the translation is evaluated by computing a score based on exact tokens, stemmed tokens and synonyms matches between the generated and the reference sentences. In particular, the metric creates a word alignment between the two sentences by mapping between words. An alignment is a mapping between same unigrams such that every unigram in each translated sentence is assigned to zero or one unigram in the source sentence. Once the set of alignments have been identified, the METEOR metric computes harmonic mean of precision and recall between the proposed and ground truth sentence.

ROUGE: The ROUGE metric [63] corresponds to a set of evaluation metrics. The most used metric for evaluating dialogue systems is ROUGE-L, which corresponds to an F-measure based on the Longest Common Subsequence (LCS) between a candidate and a reference sentence; LCS is a set of words which occur in two sentences with the same order. As opposed to n-gram, the words are not required to be adjacent, additional words between the LCS sequence can exist.

BLEU, METEOR and ROUGE metrics were shown to be effective metrics for machine translation task. However, Liu et al. [64] have showed that these metrics correlate very weakly with human evaluation when applied for dialogue systems, primarily because an utterance can have many possible re-

sponses. These word-overlapping metrics achieve best results when the space of responses is small and lexically overlapping [83] which is not the case for dialogue systems responses.

Recently, a wide number of works have looked into how to automatically evaluate response generation models. Lowe et al. [65] built a classifier to predict how appropriate the responses are, from the dialogue context and the words in the responses. The classifier was trained on a set of responses labeled by humans, indicating their quality. Browman et al. [13] proposed a new evaluation paradigm called adversarial evaluation, adopted from the Turing Test. The key idea is to train an evaluator classifier to recognize between human responses and machine-generated responses.

Although, there is a bunch of automated metric measuring the quality of the responses, the most reasonable way is to have humans manually evaluate the appropriateness, the engagement and the fluency of the system. The bottleneck of this evaluation technique is that it is labor-intensive and prohibitively costly.

2.5 Conclusion

In this chapter, we presented a thorough background knowledge about neural deep networks approaches when applied on NLP tasks. Moreover, we gave a detailed related works about open-ended dialogue systems (or chatbots) and a brief overview about task-oriented dialogue systems. In this thesis, we do not address task-oriented dialogue systems. Instead, we focus on open-ended neural dialogue systems.

Chapter 3

Datasets

A wide range of deep learning approaches have been proved to be effective for various NLP tasks and in particular for modelling dialogue systems. Much of the progress achieved is due to a combination of several factors including the computational capability of machines, innovation in deep neural networks approaches and the availability of enormous public datasets. One of the primary bottlenecks in training end-to-end dialogue systems, and in scaling them to various domains, is the scarcity of good conversational datasets. Indeed, the quality of the dataset can have a compelling influence on the response generation process of open-ended conversational agents. In this chapter, we give an overview of datasets that are available to train end-to-end dialogue systems. We discuss in details how these corpora have been compiled and we emphasize the repercussion that they may have on the quality of the generated responses. Moreover, we introduce a high-quality dataset developed from Reddit. We explain the different steps that we employ to pre-process it, making it a good dataset for training future dialogue systems.

3.1 Overview of existing end-to-end datasets

While there exist a number of publicly available conversational datasets, the NLP research community is still struggling to build an ideal chit-chat corpus that highly resembles general human-human dialogue. Currently, the most used datasets for training end-to-end systems are either taken from movie scripts or from micro-blogging websites like Twitter, which is not satisfying

for generating natural sounding conversations. Collecting a realistic human-human conversation is a major challenge in the development of dialogue systems. Ideally, conversations between individuals should be recorded and then transcribed in the pursuit of having natural true interactions [95]. However, for data privacy considerations, this procedure could not be feasible without the consent of humans participating in the conversation. Indeed, researchers have to inform individuals that they are being recorded. If they confirm their participation, participants might be asked to talk about a specific topic while conversing together. Unintentionally, they will bias the conversation towards the task being asked and they will adjust their language to fulfill the requirement [95]. Such artificial datasets are being more ubiquitous especially with the increased use of crowdsourcing platforms such as Amzaon Mechanical Turk [45]. As a result, the conversation loses its natural and spontaneous behavior which leads to deteriorate the overall quality of the collected corpus. Apart from that, acquiring a big enough corpus will take a long time as well as tremendous efforts. Nevertheless, one cannot deny the usefulness of the existing conversational datasets as they follow a consistent, engaging and fluent flow [30]. In the following subsections, we present some current existing conversational datasets.

3.1.1 Fictional Datasets

Some current data-driven approaches are using corpora drawn from fiction such as movie scripts [109] or television series [60]. Despite, the sheer size of these datasets, learning dialogue exchanges can be difficult because the model has to account for external events that are not mentioned in the dialogue. The same problem can be replicated in social media websites (e.g, Twitter, Weibo). But, regardless of the dependence on external information, Forchini et al. [30] advocates that scripted language in movies are very similar to spontaneous face-to-face conversations with respect to a wide range of linguistic characteristics. In fact, movie subtitles represent a prominent resource for language variety. They span various genres and combine different spoken language structure together including dialectal expressions, idiomatic expressions

and slang [30]. Dialogue turns in different scenes can involve different actors with different personalities, backgrounds and intentions. Having such dataset can allow data-driven models to personalize the conversation by exploiting the personality endowed in each character in the movie [60].
Nowadays, the web is full of thousands of scripted corpora sourced from movies or TV series making the task of compiling a gigantic dataset a relatively fast and less challenging task. The available corpora can be grouped into two categories: 1) data with speakers annotation explicitly providing the appropriate speaker for every spoken utterance. 2) data without the speaker annotation bringing only the actual scripts. Here are some of the available fictional corpora:

OpenSubtiles Dataset [108]: It is a huge collection of movies taken from the OpenSubtitles website [1], having over one billion words. It spans multiple movies genre including romance, family, comedy, science fiction, action, etc. It consists of movie conversations encoded in the form of XML. Despite being quite large, this dataset lacks speakers annotation which does not secure illustrating conversations between two individuals.

Subtle Dataset [2]: It is also based on the OpenSubtitles website but unlike the OpenSubtiles corpus, the Subtle dataset comprises Interaction-Response pairs. The primary purpose behind building the Subtle corpus is to help dialogue systems dealing with out-of-domain interactions. It is a much smaller corpus than the OpenSubtiles dataset as it contains 20M tokens.

MovieDic Dataset [6]: The corpus is relatively small as it was extracted from 753 movies found in the Internet Movie Script Data Collection [2]. It contains roughly 133K dialogues. Unlike OpenSubtitles and Subtle datasets, each utterance in the MovieDic conversations is annotated with the appropriate speaker. Moreover, the context written in the original script is carried over to the corpus.

MovieTriple Dataset [96]: It was extracted also from the MovieDic

[1] `https://www.opensubtitles.org`
[2] `http://www.imsdb.com`

dataset and contains dialogue scripts collected from 614 movies. It has roughly 345,296 utterances. Each line in this dataset consists of three dialogue turns between two interlocutors.

Cornell Movie-Dialogue Dataset [24]: Similar to the previous datasets, the Cornell dataset consists of short utterances based on movie scripts. It is a relatively small corpus in which 305K utterances were extracted from 617 movie scripts. However, what makes it distinguishable from other corpora is that it comes with a good amount of metadata for each movie such as release year, IMDB rating and for each character such as gender, position of the character on movie credits.

TVD Dataset [91]: This dataset was built based on the drama TV show Big Bang Theory and the comedy TV show Game of Thrones. It comes with raw scripts along with crowd-sourced textual descriptions (brief episode summaries, longer episode outlines) and meta-data (speakers, shots, scenes). They employed a text alignment algorithm to attach the crowd-sourced description and the meta-data to the corresponding dialogue in each script.

The Corpus of American Soap Operas [25]: This corpus contains 100 millions of tokens extracted from 22,000 transcripts of the American soap operas TV show from 2000 until 2012. Unlike OpenSubtitles dataset, this corpus does not have a variety of genres as it consists of only dramatic vocabulary. It was primarily collected to provide insight into informal, colloquial American speech. Although the dataset is quite big, it does not have speakers labels.

3.1.2 Real Datasets

In addition to the fictional data, there is a number of spontaneous Human-Human conversations that are collected from real-interaction websites such as micro-blogging websites or forums. One can mention the Twitter Corpus [88] and the Ubuntu Dialogue Corpus [66] which were extensively used by the NLP community. However, most of these datasets endure a challenging issue: conversations often depend on external events and topic that are not present in the dataset. Hereby, the system should infer this information from an external knowledge base, which makes the task of generating dialogue difficult.

Name	# of dialogues	# of utterances	# of tokens
OpenSubtitles	36M	140M	1B
MovieDic	132K	764K	6M
MovieTriples	245K	736K	13M
Cornell Movie-Dialogue Corpus	220K	305K	9M
Subtle	3.35M	6.7M	20M
TVD Dataset	10K	60K	600K
The corpus of American Soap Operas	1.2M	10M	100M
Twitter Dataset	1.3M	2.6M	125M
Ubuntu Dialogue Corpus	930K	7.17M	100M

Table 3.1: Human-Human dialogue interactions drawn from movies, TV shows, Twitter and Ubuntu chat forum. [95]

Twitter Corpus [88]: The corpus consists of 1.3 million conversations drawn from tweets. Generally, utterances tend to be short as there is a restriction on the maximum number of characters per tweet (140 characters). The dataset was crawled for a 2 month-period in the summer of 2009 where the conversations were built based on the posts and their replies. The twitter data span multiple topics and events making it an open-domain datset. Notwithstanding the sheer size of the corpus, utterances tend to break some linguistic conventions (e.g., abbreviations, slang, typos and punctuation are not used properly and so forth). Twitter data suffers not only from lexical variations but also from incorrect grammar. Moreover, users use an enormous amount of hashtags that does not reflect human-like conversation but instead makes the conversation sounds artificial and unnatural. This is why, an extensive preprocessing should be done prior to training data-driven dialogue systems.

Ubuntu Dialogue Corpus [66]: It is a dataset containing roughly 1 million multi-turn conversations with over 100 million words and 7 million utterances. As opposed to the Twitter corpus, this dataset represents a goal-oriented technical support domain extracted from the Ubuntu Internet Relayed Chat channel. Users who have a specific Ubuntu technical problem head towards the chat channel to chat about a solution for their issues. The technical

interactions range from software-related issues and hardware-related issues to informational needs. While this dataset presents a good opportunity for researches to train data-driven specific-domain dialogue agents, it hinders scaling up to new domains. In addition, given the large technical diversity of the Ubuntu Dialogue Corpus, there is an enormous number of rare words which requires a large vocabulary; thus, making the model big and the training more difficult.

3.1.3 Corpus size

Training deep neural networks on large-scale dialogue datasets is crucial for modelling dialogue systems [113]. There are mainly two point of views on the significance of the corpus size [95]: one comes from a machine learning perspective and the other one comes from an NLP perspective. From a machine learning viewpoint, training on large corpora helps the statistical machine learning models to generalize well to unseen data [40]. Luckily, the era of big data has made building deep neural networks much easier by alleviating the burden of compelling enormous amount of data [55]. Serban et al. [95] argue that training with few examples of dialogue utterances may require structural priors to be added to the model architecture. In a recent talk, Yann LeCun[3] and Christopher Manning[4] discussed how much innate structure is required for AI models in general, and in particular what innate priors should researchers build into the architecture of deep learning systems[5]. Arguably, there are at least two types of structures: structure integrated into the model as innate prior such as the recursive assumption in the RNN models, and structure acquired naturally and dynamically from the data such as the alignments calculated by the attention mechanism. Yann LeCun contends that all structure should be learned from the environment by observing the data examples and grasping knowledge. However, Christopher Manning is a prominent advocate for integrating more linguistic structure into the models. To the best of our

[3] a Deep Learning pioneer

[4] a Natural Language Processing pioneer

[5] `https://www.youtube.com/watch?v=fKk9KhGRBdI`

knowledge, there is no current highly structured deep learning models. Such situation has pushed most researchers to invest more time on building large-scale datasets to train deep learning models [55]. In general, LeCun et al. [55] explain that supervised deep learning models typically achieve acceptable performance when trained with around 5000 examples. However, these models will nearly reach human performance when trained with a dataset having more than 10 million examples.

From an NLP point of view, the number of training examples required for training a machine learning model will grow with the linguistic diversity and the number of topics of a corpus [95]. Since conversations are a steady back and forth of linguistic interactions where responses are chosen depending on the utterances received from the other interlocutor, dialogues can be extremely ambiguous ([17], [52]), thereby having vast amount of training examples, may indemnify the statistical complexity of a corpus. Many works have succeeded to generate good dilaogue responses when trained on a sufficiently large conversational dataset [97], [98], [112].

Nevertheless, working with large datasets is hard for several reasons. First, they are computationally expensive to process, and the time of training grows as the size of the dataset grows, which increases the learning cost. In addition, running large-scale datasets that fit in memory can be exorbitantly costly. This is why, scalable learning techniques such as parallel infrastructures and optimized codes are needed to alleviate the hurdle of dealing with gigantic dataset. In the following section, we will discuss our techniques for preparing the Reddit dataset and the preprocessing steps that we followed.

3.2 Reddit Dataset

One of the main weaknesses of dialogue systems is caused by the paucity of high-quality conversational datasets. The well-known OpenSubtitles dataset [109] lacks speaker annotations, thus making it more difficult to train conversation systems which demand high quality speaker and conversation level tags. Therefore, the assumption of treating consecutive utterances as turn

exchanges uttered by two persons [113] could not be viable. To enable the study of high-quality and large-scale dataset for dialogue modeling, we have collected a corpus of 35M conversations drawn from the Reddit data[6], making a step forward into building a good chit-chat corpus that resembles human conversational dialogue.

Reddit is a social news and entertainment website where people can post their questions, connect, discuss different topics and exchange ideas in an open environment. Members can submit their posts in the form of natural language text, photos, videos or links. Users can then rank the submissions by voting "up" or "down". Reddit organizes content by subject into user-created areas of interest dubbed *subreddits* which cover a wide range of topics including news, food, sports, movies, fitness, music, politics, etc. Posts that receive high scores will be ranked first on the subreddit's front page. While Reddit has strict rules which forbid offensive posts such as harassment, violence or spams; some members do break the consent. Luckily, Reddit administrators keep doing their best to maintain the website by filtering low-quality content. One major benefit that Reddit offers is the grammatical quality of sentences since most of the subreddits are monitored by moderators who make sure to keep discussions free of harassment and filled with quality content.

The Reddit dataset is composed of posts and comments, where each comment is annotated with rich meta data (i.e., author, number of upvotes and downvotes, number of replies, user's comment karma (a reward earned for posting popular content), etc.[7]). To harvest the dataset, we curated 95 English subreddits out of roughly 1.2M subreddits[8] including: "/r/worldnews", "/r/sports", "/r/movies", "/r/televisions", "/r/politics", "/r/Canada", "/r/education", "/r/business", etc. Our choice was based on the top-ranked subreddits that discuss topics such as news, education, business, politics and sports. We processed Reddit for a 12 month-period ranging from November 2016 until December 2017 excluding June and July. For each post, we retrieved all

[6] `https://redd.it/3bxlg7`

[7] `https://github.com/reddit-archive/reddit/wiki/JSON`

[8] As of March 2018

Year	Month	Comments		Submissions	
		Original	Processed	Original	Processed
2016	Nov	71.02M	8.02M	8.66M	376.5K
2016	Dec	72.94M	8.04M	8.92M	409.4K
2017	Jan	78.95M	11.70M	9.22M	423.8K
2017	Feb	70.61M	10.37M	8.59M	379.0K
2017	Mar	79.72M	11.21M	9.62M	409.5K
2017	Apr	77.48M	10.90M	9.21M	389.4K
2017	May	79.81M	9.10M	9.50M	396.2K
2017	Aug	84.65M	11.10M	9.59M	381.8K
2017	Sep	83.17M	10.54M	9.79M	358.8K
2017	Oct	85.82M	8.29M	10.28M	360.0K
2017	Nov	84.97	8.34M	10.38M	365.3K
2017	Dec	85.97M	11.18M	10.57M	389.7K
Total		955.1M	118.8M	115.1M	4.65M

Table 3.2: Statistics about the size of the Reddit comments and submissions before and after preprocessing.

comments and we recursively followed the chain of replies of each comment to recover the entire conversation. The sheer size of the dataset renders it as an interesting candidate for building a high-quality dataset.

3.2.1 Pre-processing

Pre-processing is a key procedure towards weeding out the noise that exist in the Reddit dataset. The Reddit comments and submissions were originally encoded in Markdown/HTML language. Initially, we proceeded by stripping off all the Markdown/HTML tags to obtain plain text. While the style of writing used in Reddit is widely varied, much of the text contains url links. Consequently, we replaced all the urls with $<URL>$ placeholders. As a next step, we removed all the punctuation, emojis and emoticons. As opposed to Tweets, Reddit dataset is often semantically well-structured and is not filled with spelling errors thanks to moderator's efforts. Therefore, we do not perform any spelling correction procedure. When harvesting the dataset, we solely extracted conversations from the English subreddits but we noticed that some of them may contain replies or posts in other languages. To remedy the issue,

4-gram	Frequency
i do n't think	137,673
i do n't know	131,961
! ! ! !	125,462
. i do n't	109,297
, i do n't	67,760
i do n't have	57,505
i 'm not sure	57,214
if you do n't	55,629
i 'm going to	54,554
do n't want to	50,283

Table 3.3: 4-grams with frequency higher than 50K in Reddit dataset. As part of preprocessing, some of the dialogues containing these 4-grams are omitted until they become less dominant.

we employed a simple function word-driven filter to remove non-English posts and messages. Some comments in Reddit might be sometimes very long especially when users are replying to a controversial question or expressing their thoughts. Subsequently, a comment can expand to one paragraph or more. Since Seq2Seq models struggle with keeping track of long-dependency structure, we employed a simple heuristic that picks the first sentence from each paragraph and we restricted the length of sentences to be no more than 150 characters. Table 3.2 shows the original size and the processed size of both comments and submissions for the duration of 12 months.

As explained in Chapter 2 (Subsection 2.2.6), the Seq2Seq model tend to generate safe responses which can be ascribed to the relative frequency of generic responses in contrast with the relative sparsity of diverse responses [59]. To prevent the most frequent sentences from dominating the training process, we set a threshold for the maximum frequency of 4-grams (i.e., 50K). We want the most frequent 4-grams not to exceed the threshold. Hence, the additional dialogues subsuming 4-grams with surplus frequency are omitted from the data. Note that setting up such threshold depends on the size of the dataset as well as the distribution of words.

The most frequent 4-grams in the Reddit dataset are represented in Ta-

ble 3.3. Afterwards, we removed the conversation whose utterances shorter than 3 words to help the model produce longer outputs. After the different preprocessing steps, we were left with 35M dialogues of three utterances.

3.3 Conclusion

Training deep neural networks with high-quality conversational dataset is key for generating fluent and engaging dialogue responses. In this chapter, we gave a detailed overview of the currently available datasets suitable for training data-driven dialogue systems. In addition, we discussed the repercussion of the corpus size and quality on the training procedure. Finally, we presented our dataset extracted from Reddit. We believe that this dataset is a good resource for training different components of future dialogue systems, making a step towards aiding researcher in building consistent and engaging open-ended conversational agents.

Chapter 4

THRED

In this chapter, we introduce our novel method, dubbed THRED, that tackles the problem of diversity in dialogue responses by enriching the responses with new topic words and by accounting for the context of the conversation.
Mathewson and Mirowski [72] introduced an artificial improvisor, which captures the general theme of the dialogue by integrating low dimensional learned topic representations into the Seq2Seq model. Their work was inspired from [78], where the authors enriched a neural language model by using additional inputs such as topic information. In this direction, Xing et al. [126] used a similar idea but added an extra probability value in the decoder to bias the overall distribution towards leveraging topic words in the generated responses. However, their architecture does not focus on capturing conversation history. All of these improvements are motivated by the scarcity of diversity and informativeness of the responses. Our work follows on from these works with the additional aim of generating context-aware responses by using a hierarchical joint attention model. The hierarchy part (see section 2.4 in Chapter 2) allows the model to catch long-term conversation dependency and the attention part (see section 2.2.7 in Chapter 2) focuses on modeling topic words and on attending to important words in the utterances. Therefore, the overall structure of the proposed model conditions jointly the response on additional topic words and past utterances exchanges.

The model's idea is inspired from the observation of our daily conversations, where we relate a response to a specific topic based on what we have said

before. Subsequently, given the topic of the conversation, our mind selects topic words that are relevant to the conversation history. For example, consider the following conversation taken from the Reddit test dataset:

Speaker A: *sanctions are an act of war*

Speaker B: *why do you think that ?*

THRED model replies by the following response: *because it's really a* ***theory*** *that supports* ***terrorism*** *. and this has an effect on the idea of a* ***regime*** *that isn't the same as a* ***government***. However, the Seq2Seq model generates the following answer: *because it 's an unpopular opinion , and that 's why it 's a bad thing to say .*

In the first response, "*theory*", "*terrorism*", "*regime*" and "*government*" are words that are drawn for the topic words acquired from a pre-trained LDA model. It is evident that the above topic words follow relatively the context of the conversation. We can see that the model succeeds in enriching the response with topic words that represent people's prior knowledge in conversations. It uses these words as a building block for the response. Such enrichment makes the conversation more informative, diverse and engaging. On the other hand, the generated response from the Seq2Seq model seems semantically poor and generic and does not provide a plausible response to the input message.

4.1 Topical Hierarchical Recurrent Encoder Decoder

The Topical Hierarchical Recurrent Encoder Decoder (THRED) can be viewed as a hybrid model that conditions the response generation on conversation history captured from previous utterances and on topic words acquired from a LDA model [9]. The proposed approach extends the standard Seq2Seq model by leveraging topic words in the process of response generation and accounting for conversation history. Figure 4.1 delineates an overall picture of our model.

THRED essentially focuses on modeling jointly two specifications that presumably make the task of response generation successful: context-awareness and diversity [136]. More specifically, we jointly model the history and the

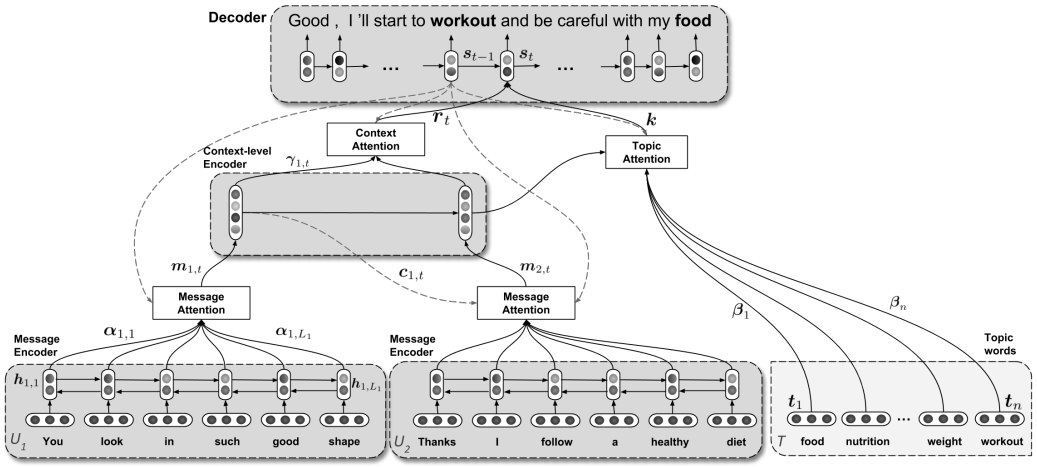


Figure 4.1: THRED model architecture in which we jointly model two specifications: context-awareness (modeled by **Context Attention**) and diversity (modeled by **Topic Attention**).

topic information of what has been said throughout the conversation. To this end, we employ a two-level attention mechanism: *message attention* and *context-topic joint attention*.

In encoding, the model encodes every input message in the conversation history as hidden vectors by the message encoder and further processes the hidden vectors by message attention to highlight important parts that contribute to produce an appropriate response during the generation process. The resulting attentional vector (i.e., the utterance representation vector) is then uploaded to the *context-level encoder* to capture long-term contextual information. As the conversation continues and the context grows, this hierarchy allows the model to make coherent and smooth predictions of the next utterance, thus fulfilling the condition of context-awareness and avoiding the dullness of the responses generated from the standard Seq2Seq model.

During decoding, the model obtains embeddings of the topic words via a pre-trained GloVe model [84]. These embeddings are then summarized as a topic vector by *topical attention*. At this point, the context-topic joint attention forms an attentional vector by unifying the utterances representation vectors and their topical information and feed it to the decoder to generate the response. We detail in the following subsections the components of our model.

4.1.1 Message Encoder

Let D be a sequence of N utterances within a dialogue $D = \{U_1, ..., U_N\}$. Every utterance $U_i = \{w_{i,1}, ..., w_{i,L_i}\}$ contains a random variable L_i of sequence of words where $w_{i,k}$ represents the word embedding vector at position k in the utterance U_i. The message encoder sequentially accepts the embedding of each word in the input message U_i and updates its hidden state at every time step t by a bidirectional GRU-RNN [19] according to:

$$h_{i,t} = GRU(h_{i,t-1}, w_{i,t}), \forall t \in \{1, ..., L_i\} \tag{4.1}$$

where $h_{i,t-1}$ represents the previous hidden state.

4.1.2 Message Attention

Different parts of the conversation history have distinct levels of importance that may influence the response generation process. The message attention in THRED operates by putting more focus on the salient input words with regard to the output. It employs a looking-back strategy by glimpsing at the entire input sequence at every decoding step. The decoder can then decide what message words are more relevant for the current decoding step. It computes, at step t, a weight value $\alpha_{i,j,t}$ for every encoder hidden state $h_{i,j}$ and linearly combines them to form a vector $m_{i,t}$ according to Bahdanau attention mechanism [4]. Formally, $m_{i,t}$ is calculated as:

$$m_{i,t} = \sum_{j=1}^{L_i} \alpha_{i,j,t}\, h_{i,j},\ \forall i \in \{1, ..., N\} \tag{4.2}$$

where $\alpha_{i,j,t}$ is computed as:

$$\alpha_{i,j,t} = \frac{\exp(e_{i,j,t})}{\sum_{k=1}^{L_i} \exp(e_{i,k,t})}; e_{i,j,t} = \eta(s_{t-1}, h_{i,j}, c_{i,t}) \tag{4.3}$$

where s_{t-1} represents the hidden state of the decoder (further details are provided later), $c_{i,t}$ delineates the hidden state of the context-level encoder (computed in Equation (4.4)), η is a multi-layer perceptron having $tanh$ as activation function. Unlike the Bahdanau attention mechanism, the attentional

vector $m_{i,t}$ is based on both the hidden states of the decoder and the hidden states of the context-level encoder. We are motivated by the fact that $c_{i,t}$ may carry important information that could be missing in s_{t-1}. In summary, the attentional vector $m_{i,t}$ is an order-sensitive information of all the words in the sentence, attending to more important words in the input messages.

4.1.3 Context-Level Encoder

The context-level encoder takes as input each utterance representation ($m_{1,t}$, $\cdots$, $m_{N,t}$) and calculates the sequence of recurrent hidden states as shown in Equation (4.4):

$$c_{i,t} = GRU(c_{i-1,t}, m_{i,t}), \forall i \in \{1, ..., N\} \tag{4.4}$$

where $c_{i-1,t}$ delineates the previous hidden state of the context-level encoder and N represents the number of utterances in the conversation history. The resulted $c_{i,t}$ vector summarizes all past information that have been processed up to position i.

4.1.4 Context-Topic Joint Attention

Context Attention: On top of the context-level encoder, a context attention is added to attend to important utterances in the conversation history. Precisely, the context attention assigns weights $(\gamma_{1,t}, ..., \gamma_{N,t})$ to $(c_{1,t}, ..., c_{N,t})$ and forms a vector r_t as

$$r_t = \sum_{j=1}^{N} \gamma_{j,t} c_{j,t} \tag{4.5}$$

where:

$$\gamma_{j,t} = \frac{\exp(e'_{j,t})}{\sum_{i=1}^{N} \exp(e'_{i,t})}; e'_{i,t} = \eta(s_{t-1}, c_{i,t}) \tag{4.6}$$

Topic Attention: In order to infuse the response with information relevant to the input messages, we enhance the model with topic information. We assign a topic T to the conversation context using a pre-trained LDA model [36]. Further details about the LDA model are provided in Chapter 2. The LDA parameters were estimated using the collapsed Gibbs sampling algorithm

[137]. We provide further details on how we train this model in Chapter 5. In our case, the conversation history is a short document, so we believe that the most probable topic will be sufficient to model the dialogue. After acquiring topic words for the entire history, we pick the n highest probable words under T (we choose $n = 50$ in our experiments). The topic words $\{t_1, \cdots, t_n\}$ are then linearly combined to form a fixed-length vector k. The weight values are calculated as the following:

$$\beta_{i,t} = \frac{exp(\eta(s_{t-1}, t_i, c_{N,t})}{\sum_{j=1}^{n} \exp(\eta(s_{t-1}, t_j, c_{N,t}))}, \forall i \in \{1, ..., n\} \tag{4.7}$$

where $c_{N,t}$ is the last hidden state of the context-level encoder and s_{t-1} is the $i - 1^{th}$ hidden state in the decoder. The topic attention uses additionally the last hidden state of the context-level encoder $c_{N,t}$ in order to diminish the repercussion of impertinent topic words and feature the relevant ones to the message. Unlike [126], our model employs the final context-level encoder hidden state $c_{N,t}$ in order to account for conversation history in the generated response. In summary, the topic words are summarized as a topic vector k representing prior knowledge for response generation. The key idea of this approach is to affect the generation process by avoiding the need to learn the same conversational pattern for each utterance but instead enriching the responses with topics and words related to the subject of the message even if the words were never used before in the training utterances.

Decoder

The decoder is responsible for predicting the response utterance U_{m+1} given the previous utterances and the topic words. Following [126], we biased the generation probability towards generating the topic words in the response. In particular, we added an extra probability to the standard generation probability, enforcing the model to account for the topical tokens. Consequently, the generation probability is defined as the following:

$$p(w_i) = p_V(w_i) + p_K(w_i) \tag{4.8}$$

where K and V represent respectively topic vocabulary and response vocabulary. p_V and p_K correspond to Equation (4.9) and Equation (4.10) respectively.

$$p_V(w_i) = \frac{1}{M} \exp(\sigma_V(s_i, w_{i-1})) \tag{4.9}$$

$$p_K(w_i) = \frac{1}{M} \exp(\sigma_K(s_i, w_{i-1}, r_i)) \tag{4.10}$$

where $s_i = f(w_{i-1}, s_{i-1}, r_i, k)$ and

$$M = \sum_{v \in V} \exp\left(\sigma_V(s_i, w_{i-1})\right) + \sum_{v' \in K} \exp\left(\sigma_K(s_i, w_{i-1}, r_i)\right)$$

4.2 Conclusion

In this chapter, we introduce THRED, a novel and multi-turn dialogue system, aiming at generating context-aware and diverse responses. Our model builds upon the traditional Seq2Seq model by adding a hierarchical attention mechanism that conditions the responses on conversation history and on topic words derived from a topical model.

Chapter 5

Experiments And Results

In this chapter, we evaluate THRED qualitatively and quantitatively. We focus on the task of evaluating the next utterance given the conversation history. We compare THRED against three baselines, namely Standard Seq2Seq with attention mechanism [4], HRED [96], and TA-Seq2Seq [126]. For Standard Seq2Seq and TA-Seq2Seq, we concatenate the dialogue history to account for the context in a multi-turn conversation [62].
Furthermore, we introduce novel automatic metrics that can be adopted for any dialogue generation model. Moreover, human judgment is exploited to assess the quality of THRED and the baselines. We also demonstrate that the introduced metrics are in-line with human judgement. Finally, we investigate the impact of two training datasets (i.e., OpenSubtitles and Reddit) on the response generation process by contrasting them in terms of human judgment and automated metrics.

5.1 Experimental Setup

In this section, we present the datasets that we used for training THRED alongside the baselines. We also explain in details the training procedure for the deep learning models and for the topical LDA model.

5.1.1 Implementation

We implemented our model using the open-source deep learning framework TensorFlow [1]. Moreover, we implemented the baselines HRED and TA-

Model	Original PPL	Replicated PPL
HRED [97]	26.8	29.5
TA-Se2Seq [126]	122.8	126.6

Table 5.1: Original perplexity results vs. replicated perplexity results [95]

Seq2Seq models ourselves because the original code is deprecated. More specifically, we were not able to run the code given two major problems: 1) the code uses old deep leaning framework Theano and 2) the libraries are not compatible with the GPU drivers we had access to. To ensure the replication of the code, we computed the Perplexity score on the original datasets and we found similar results as claimed by the authors in the original papers.

5.1.2 Training Procedure

To train the neural network models, we used two datasets: OpenSubtitles and Reddit, discussed in details in Chapter 3. OpenSubtitles and Reddit are gigantic datasets containing each 36M and 35M dialogues respectively. Due to resource limitations, we randomly sampled from each dataset 6M dialogues as training data, 700K dialogues as development data, and 40K dialogues as test data. Each dialogue corresponds to three turn exchanges. The model parameters are learned by optimizing the log-likelihood of the utterances via Adam optimizer [49], with a learning rate of 0.0002. We followed [70] for decaying the learning rate; after reaching halfway through training, we start halving the learning rate for 4 times. The dropout rate [102] is set to 0.2 for both the encoder and the decoder to avoid overfitting. All the parameters were learned through the backpropagation algorithm [92]. We set the number of training epochs to 15 while adhering to early stopping on the validation set. The normalized gradient is rescaled whenever its norm exceeds 5 to prevent from gradient explosion. Finding the best hyperparameters for our models was done by conducting several experiments searching for the best perplexity over various settings.

The effectiveness of an LSTM/GRU network can be improved by increasing

hidden units and adding more layers [32] at the expense of having additional parameters and an increased runtime. The weights were initialized by sampling from the uniform distribution $[-0.1, 0.1]$. We experimented hidden state units with the size of 1024 for all the baselines. Similarly, for our model, we tested with encoder, decoder and context hidden state units of size 1024. Unfortunately, we faced frequent out-of-memory issues as the Seq2Seq model is already extremely memory-intensive, and adding the hierarchical layer with the attention made it even more so. Consequently, we experimented with only 800 hidden units size for all the three layers. Our mini-batch size for all the models is fixed to 128, and the vocabulary size is limited to 50K in both Reddit and OpenSubtitles. During inference, we experimented with the standard beam search with the beam width 5 and the length normalization $\alpha = 1$ (was usually found to be best) [125]. We noticed that applying the length normalization resulted in a more diverse and longer sentences . Without the length normalization, the beam search algorithm favors shorter responses over longer ones. This is because, for longer sentences, a negative log-probability is added at each step producing a lower score. More specifically, we define below the scoring function s that we employed to rank the candidate responses:

$$s(Y|X) = \frac{\log P(Y|X)}{lp(Y)} \tag{5.1}$$

$$lp(Y) = \frac{(5 + |Y|)^\alpha}{(5 + 1)^\alpha} \tag{5.2}$$

As the models are based each on a large-scale system leveraging deep learning network, we ran the training on over 4 Titan X GPUs during roughly 20 days.

Training LDA model

We trained two LDA models[1] [86]: one trained on OpenSubtitles and the other one trained on Reddit. Both of them were trained on 1M dialogues. We set the number of topics to 150, α to $\frac{1}{150}$ and γ to 0.01. We filtered out stop words

[1]We used LDA model developed in Gensim library.

and universal words such as “thank” and “you”. We also discarded the 1000 words with the highest frequency from the topic words.

5.2 Quantitative Evaluation

Evaluating dialogue systems has been heavily studied, but researchers are still on the quest for a strong and reliable metric that highly conforms with human judgment. In dialogue systems, automated metrics tend to be borrowed from other NLP tasks such as BLEU [83] from machine translation and ROUGE [63] from text summarization. Yet, such metrics fail, mainly because they are focusing on the word-level overlap between the machine-generated answer and the human-generated answer, which can be inconsistent with what humans deem a plausible and interesting response [64]. Although human evaluation represents a reasonable way to evaluate the quality of the responses, it has some limitations [65]. Typically, a dialogue system has many hyper-parameters to be optimized. Tuning parameters by running human experiments for every parameter setting is impractical, time consuming, and expensive. Online crowd-sourcing platforms (e.g., Amazon Mechanical Turk), have mitigated these scalability issues but they can be unreliable. Recruited workers may lack the motivation to accurately rate dialogue responses and their attention may wander during the process [133].
Ideally, we would like to have a well-designed automated metric that provides an accurate evaluation of the system without any human intervention or any constraint regarding scalability or time.

In the following subsections, we focus on evaluating the predicted utterance given the conversation history. The models trained on Reddit are tested with topic words derived from an LDA model trained as well on Reddit (we did the same on OpenSubtitles). We introduce two metrics that can impartially evaluate THRED and compare against the different baselines. Then, we report the results based on response diversity metric, derived from [59]. These metrics were tested on 5000 dialogues randomly sampled from the OpenSubtitles and

Reddit test dataset. It is worth mentioning that we present word perplexity on test data in Table 5.3 (along with the diversity metric). However, we do not believe that it represents a good measure for assessing the quality of responses [97]. This is because perplexity captures how likely the responses are under a generation probability distribution, and does not measure the degree of diversity and engagingness in the responses.

5.2.1 Semantic Coherence

A good dialogue system should be capable of sustaining a coherent conversation with a human by staying on topic and by following a train of thoughts [111]. The Semantic Coherence (SC) metric estimates the correspondence between the utterances in the conversation history and the generated response. The intuition behind this metric is that plausible responses should be consistent with the context and should maintain the topic of the conversation.

Our response generator THRED generates an utterance based on the 2 previous utterances in the dialogue (i.e., Utt1 and Utt2). We compute the cosine distance between the embedding vectors of the test utterances (Utt.1 and Utt.2) and the generated responses from the different models (i.e., THRED, TA-Seq2Seq, HRED and Seq2Seq). Therefore, a low score denotes a high coherence. To render the semantic representation of an utterance, we leverage the Universal Sentence Encoder [16] wherein a sentence is projected to a fixed dimensional embedding vector. The goal of the Universal Sentence Encoder is to learn low-dimensional sentence representations that can be easily and effectively employed as word embeddings ([16], [122]).

For each triple in the test dataset, we explore two scenarios: (1) we compute the semantic coherence of each generated response with respect to Utt.1 and (2) we compute the semantic coherence of each generated response with respect to Utt.2.

However, dull and generic responses such as "*i'm not sure*" tend to be semantically close to many utterances, hindering the effectiveness of the metric. To cope with this negative effect, we manually compiled a set of 55 dull responses (examples provided in Table 5.2) and computed the SC score by multiply-

Dull responses
i don't know .
i don't know what you 're talking about .
i don't know what you mean .
i 'm not sure .
i 'm not sure if that 's a joke or not .
i 'm not sure if that 's a good idea or not .
i 'm not sure what you 're saying .
i 'm not sure what you 're talking about .
i 'm not sure what you 're trying to say .

Table 5.2: Examples of generic responses selected from the set of generated responses from different models (THRED, TA-Seq2Seq, HRED, Seq2Seq)

ing the cosine distance score with the following penalty factor (akin to length penalty in Eq. (5.2) [125]).

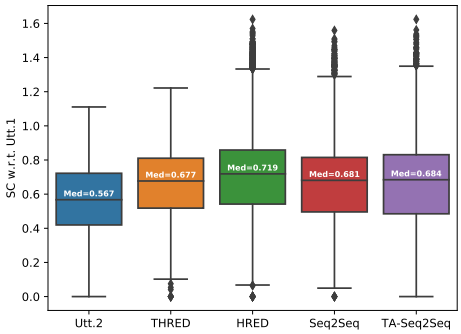
$$P = 1 + \log \frac{2 + L'}{2 + L''} \tag{5.3}$$

where L' indicates the length of the response after dropping stop words and punctuation and L'' stands for the length of non-dull part of the response after dropping stop words. The intuition here is that the longer utterances, with nearly the same similarity, communicate the intention unequivocally since it takes more words to convey the same meaning. The penalized Semantic Coherence score $SC_{penalized}$ is therefore defined as:

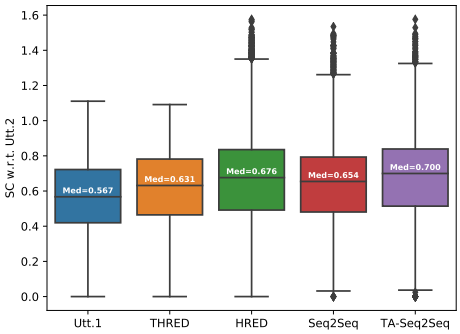
$$SC_{penalized} = P \times SC \tag{5.4}$$

The results are shown in Figure 5.1. The box plots represent the semantic coherence of each generated response from different models with respect to Utt.1 and Utt.2. The experiment is conducted on Reddit and OpenSubtitles datasets. Utt.1 and Utt.2 are semantically close to each other because they are drawn from the reference dialog. Note that the SC of Utt.1 with respect to Utt.1 is zero as the distance of a vector with itself is zero.

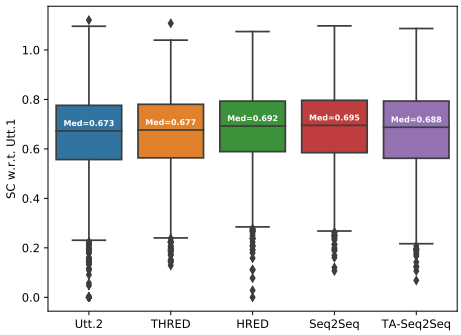
We can observe that THRED is able to generate responses than can follow the topic and semantics of the input utterances on both datasets. The metric



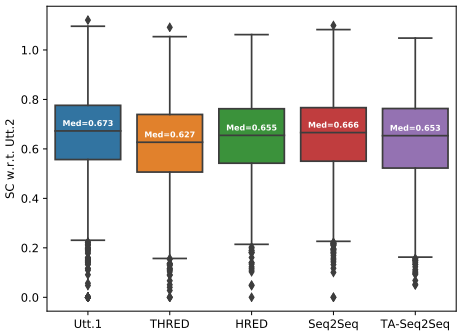
(a) SC with regard to Utt.1 (Reddit)



(b) SC with regard to Utt.2 (Reddit)



(c) SC with regard to Utt.1 (OpenSubtitles)

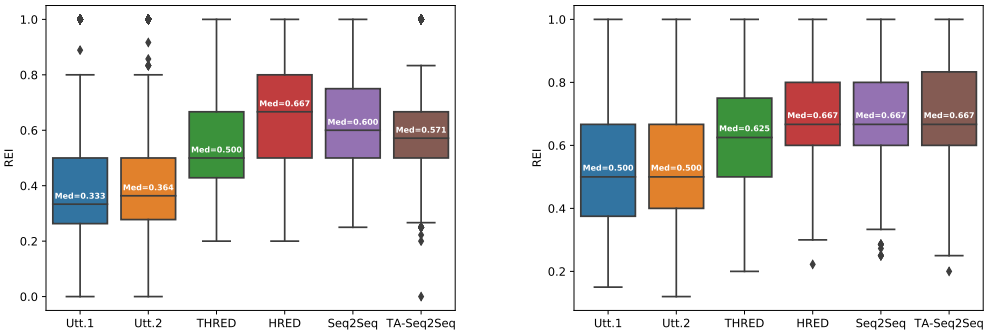


(d) SC with regard to Utt.2 (OpenSubtitles)

Figure 5.1: Box plots showcasing the performance of the generated responses from different models based on the Semantic Coherence metric with respect to Utt.1 and Utt.2. From left to right, the labels in horizontal axis are Utt.1, Utt.2, THRED, HRED, Seq2Seq, TA-Seq2Seq. THRED surpasses all baselines in coherence with Utt.2, and works mildly better in coherence with Utt.1.

looks more stable when tested on Reddit as the number of outliers is visibly larger in OpenSubtitles. To ensure the statistical significance of THRED, we conducted Student's t-test over the average values of SC metric. THRED with p-value < 0.001 outperforms all baselines especially when the comparison is made against the second utterance (Utt.2).

On the other hand, THRED is level with compared models in semantic distance with regard to the first utterance (Utt.1). This makes sense because in a multi-turn dialogue, speakers are more likely to address the last utterance spoken by the interlocutor, which is why the SC score of the generated responses from THRED tend to be closer to the second utterance over the first



(a) Reddit (b) OpenSubtitles

Figure 5.2: Performance results of the generated responses from different models based on REI. From left to right, the labels in horizontal axis are Utt.1, Utt.2, THRED, HRED, Seq2Seq, TA-Seq2Seq.

one. Additionally, the roughly similar distances for both utterances in standard Seq2Seq and TA-Seq2Seq exhibit that by concatenating context as single input, these models cannot distinguish between early turns and late turns.

5.2.2 Response Echo Index

The goal of the Response Echo Index (REI) metric is to detect overfitting to the training dataset. More specifically, we want to measure the extent to which the responses generated by our model THRED repeat the utterances appearing in the training data. Our approach is close to sampling and finding the nearest neighbour in image generative models [107].
We randomly sampled 10% of the training data of both OpenSubtitles and Reddit. We considered only 10% of the data because finding nearest neighbor for each test sample over the entire training corpus takes a tremendous amount of time.

Each utterance is represented by lemmatized bag-of-words where stop words and punctuation marks are omitted. REI is expected to be low since the generated responses should be distant from the nearest neighbor, denoting that they do not copy sentences from the training corpus.

According to the results, presented in Figure 5.2, the Jaccard similarity

scores of Utt.1 and Utt.2 are clearly the lowest since both utterances are derived from the ground truth test dataset. Moreover, we notice that THRED is able to generate unique responses which appear to be drawn from the input distribution, while they are measurably far from the input dataset. This strength in THRED is attributed to the topic attention and incorporating topic words in the response generation. Due to the same reason, standard Seq2Seq and HRED fall short in this metric.

5.2.3 Degree Of Diversity And Perplexity

To account further for diversity in generated responses, following [59], we calculated *distinct*-1 and *distinct*-2 metrics by counting unique unigrams and bigrams in generated responses, normalized by the total number of generated words to prevent from favoring long responses. The two metrics measures the informativeness and the diversity of the generated responses. The results, given in Table 5.3, indicate that THRED yields content rich and diverse responses, mainly ascribed to incorporating new topic words into response generation. On Reddit dataset, THRED surpasses all the baselines with a gain of 5% in *distinct*-1 and 37% in *distinct*-2 over TA-Seq2Seq (second best). A good performance boost is also observed from THRED over the baselines on OpenSubtitles, suggesting a 30% and 45% jump for *distinct-1* and *distinct-2* respectively. Further, in perplexity, THRED performs slightly better. We do not consider perplexity as a good metric for measuring conversational diversity success but it still can capture good responses by assigning high probability to convenient word choices.

5.3 Human Evaluation

Besides the quantitative measures, 4-scale and side-by-side human evaluation were carried out. In order to conduct experiments in which humans are involved, the study should be approved by the University of Alberta Ethics Board. To this end, we submitted an ethical application following the guidelines provided by the University of Alberta. Luckily, our investigation has

Method	perplexity	*distinct-1*	*distinct-2*
OpenSubtitles			
Seq2Seq	74.37	0.0112	0.0258
HRED	74.65	0.0079	0.0219
TA-Seq2Seq	75.92	0.0121	0.0290
THRED	**73.61**	**0.0157** (*+30%*)	**0.0422** (*+45%*)
Reddit			
Seq2Seq	62.12	0.0082	0.0222
HRED	63.00	0.0083	0.0182
TA-Seq2Seq	62.40	0.0098	0.0253
THRED	**61.73**	**0.0103** (*+5%*)	**0.0347** (*+37%*)

Table 5.3: Performance results of diversity and perplexity on Reddit test data and OpenSubtitles test data. The numbers in the bracket indicate the gain of *distinct*-1 and *distinct*-2 over the second best method (i.e., TA-Seq2Seq). Further, in perplexity, TA-Seq2Seq performs slightly better in both datasets.

been reviewed and approved for its adherence to ethical guidelines.

For the purpose of evaluating the quality of the responses, five human raters were recruited. They were fluent, native English speakers: one is a librarian, two are post-secondary students and two are improvisers. In order to proceed to the evaluation process, recruiters were asked to sign a consent in which they confirm their participation to the research study. They were well-instructed for the judgment task to ensure quality rating. We showed every judge 300 conversations (150 dialogues from Reddit and 150 dialogues from OpenSubtitles) and two generated responses for each dialogue: one generated by our THRED model and the other one generated by one of our baselines. The source models were unknown to the evaluators. The generated responses were ordered in a random way to avoid biasing the judges. Figure 5.3 shows a screenshot of our evaluation website where judges evaluated the responses generated from different models.

Additionally, Fleiss' Kappa score is exploited to gauge the reliability of the agreement between human evaluators. Following [98], we consider a major agreement if two out of the three judgments are the same. Examples of generated responses from the OpenSubtitles dataset and the Reddit dataset are provided in Table 5.7.

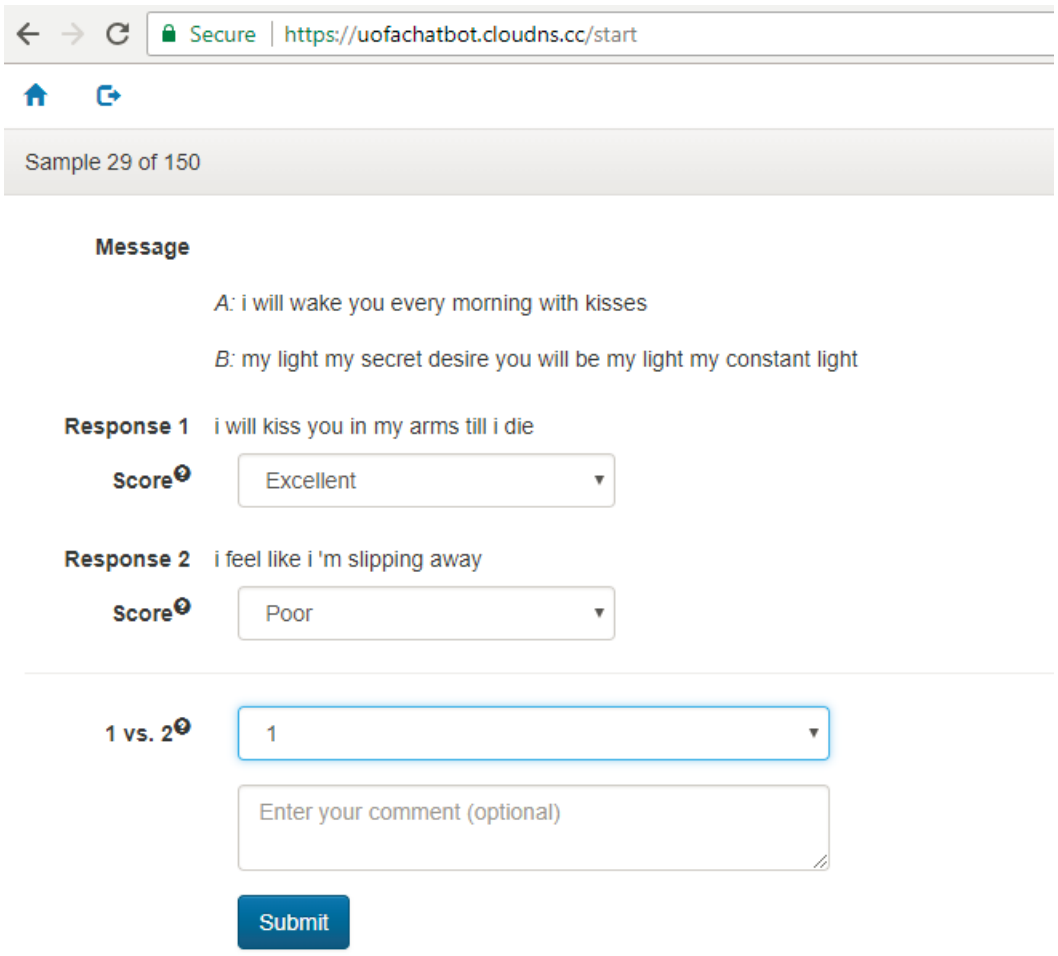


Figure 5.3: Screenshot of one dialogue context (A and B) with two candidate responses

For the 4-scale human evaluation, judges were asked to judge the responses from Bad (0) to Excellent (3). *Excellent (score 3)*: The response is very appropriate, on topic, fluent, interesting and shows understanding of the context. *Good (score 2)*: The response is coherent with the context but it is not diverse and informative. It may imply the answer. *Poor (score 1)*: The response is interpretable and grammatically correct but completely off-topic. *Bad (score 0)*: The response is grammatically broken and it does not provide an answer. The results of this experiment are detailed in Table 5.4.

The lablers with a high consensus degree rated 32.9% and 36.9% of the THRED responses in OpenSubtitles and Reddit respectively as Excellent, which is greatly larger than all baselines (up to 11.6% and 22.7% respec-

Method	Excellent	Good	Poor	Bad	Kappa
Reddit					
Seq2Seq	22.7±2.6	47.2±3.5	22.5±3.5	7.6±2.7	0.80
HRED	14.5±2.8	46.7±3.8	31.3±3.8	7.5±2.5	0.84
TA-Seq2Seq	17.1±2.4	44.8±3.5	30.1±3.2	8.0±2.3	0.72
THRED	**36.9±3.0**	**51.1±2.9**	10.3±2.4	1.7±1.5	0.84
OpenSubtitles					
Seq2Seq	8.4 ±2.2	**48.9±3.9**	33.2±3.7	9.5±3.1	0.89
HRED	11.6±2.4	41.5±3.4	36.9±3.9	10.0±2.8	0.79
TA-Seq2Seq	9.5±2.1	42.3±3.7	34.7±3.9	13.6±3.7	0.92
THRED	**32.9±3.6**	**49.2±3.3**	16.8±3.0	1.1±0.9	0.83

Table 5.4: 4-scale Human Evaluation (in %) of dialogue utterance prediction (mean preferences ±90% confidence intervals).

tively).

Apart from the 4-scale rating, we conducted the evaluations side-by-side to measure the gain in THRED over the strong baselines. Humans were asked to favor response 1 over response 2 if: (1) response 1 is relevant, logically consistent to the context, fluent and on topic; or (2) Both responses 1 and 2 are relevant, consistent and fluent but response 1 is more informative than response 2. If judges cannot tell which one is better, they can rate the responses as "Equally good" or "Equally Bad". The results, illustrated in Table 5.5, suggest that THRED is substantially superior to all baselines in producing informative and plausible responses from human's perspective. The high Kappa scores imply that a major agreement prevails among the lablers.

THRED beats the strong baselines in 52% of the test data in Reddit and 56.5% in OpenSubtitles (the numbers are achieved by averaging the win ratio). However, for the rest of the cases, THRED is equally good with the baselines (25% in Reddit and 16.5% in OpenSubtitles, calculated similarly based on Table 5.5). Hence, the ratio of cases where THRED is better than or equal with the baselines in terms of quality is 77% in Reddit and 73% in OpenSubtitles. These results are also corroborated by 4-scale evaluation reported in Table 5.4.

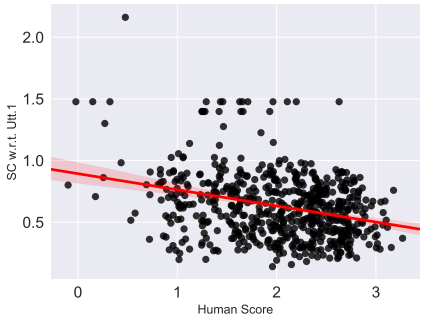
Additionally, we carried out an analysis on the correlation between the

Opponent	Wins	Losses	Ties (G)	Ties (B)	Kappa
Reddit					
THRED vs Seq2Seq	**47.5±4.4**	19.1±3.3	28.5±3.1	4.9±1.8	0.80
THRED vs HRED	**51.7±4.6**	20.1±3.4	20.9±3.1	7.2±2.3	0.75
THRED vs TA-Seq2Seq	**55.7±4.1**	13.5±2.6	24.7±3.0	6.1±1.8	0.77
OpenSubtitles					
THRED vs Seq2Seq	**54.0±4.2**	18.4±3.4	17.2±3.0	10.4±2.3	0.75
THRED vs HRED	**51.6±4.4**	19.5±3.5	18.4±2.9	10.5±2.4	0.72
THRED vs TA-Seq2Seq	**64.0±4.3**	14.4±3.1	14.1±2.5	7.5±2.1	0.90

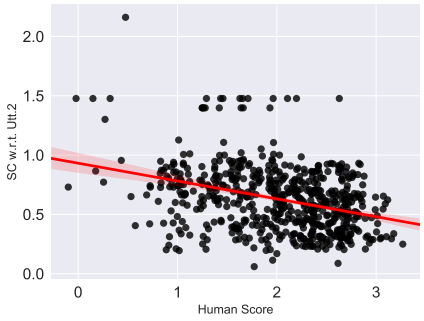
Table 5.5: Side-by-Side Human Evaluation (in %) of dialogue utterance prediction against the baselines (mean preferences ±90% confidence intervals).

human evaluator ratings and our quantitative scores. We present Pearson correlation which measures a linear relationship between the human scores and the proposed metrics.

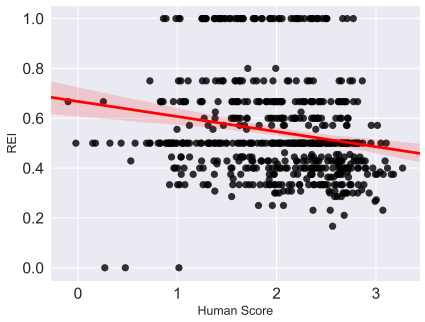
The Pearson correlation coefficient between human 4-scale ratings and the automated metrics, including SC (w.r.t. utterance 1 and w.r.t. utterance 2) and REI, are achieved as -0.312, -0.344, and -0.196 respectively on Reddit (p-value < 0.001). These results depict that SC reasonably correlates with human judgment. Pearson correlation (ρ) in SC is marginally lower than [111] ($\rho = 0.351$) and ADEM [65] ($\rho = 0.436$). Nonetheless, drawing a comparison here is difficult as the number of human subjects depends on subjective opinions. Additional work is necessary for reliability and intersubject agreement. The REI metric does not correlate well since it is tailored to measure dissimilarity between training data and the generated responses, differing from human perception of acceptable responses. The correlations are further visualized as scatter plots in the Figure 5.4. In order to better visualize the density of the points, we added stochastic noise generated by Gaussian distribution $\mathcal{N}(0, 0.1)$ to the human ratings (i.e., horizontal axis) at the cost of lowering the correlation, as done in [65]. A negative correlation is anticipated since the higher human ratings correspond to the lower semantic distance and REI.



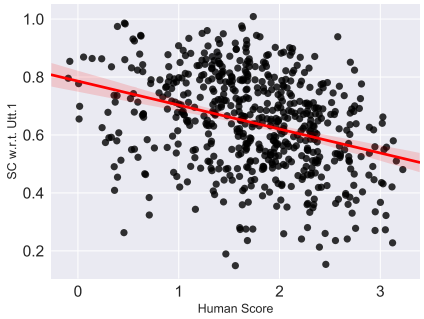
(a) **Reddit:** SC w.r.t. Utt.1 ($\rho = -0.312$)



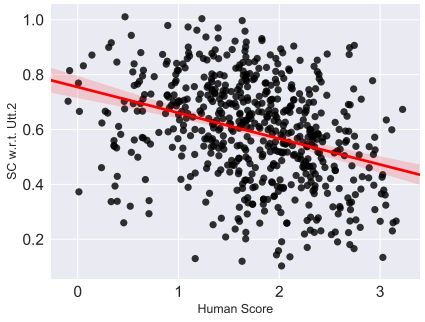
(b) **Reddit:** SC w.r.t. Utt.2 ($\rho = -0.344$)



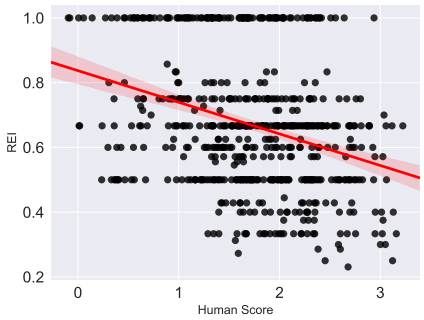
(c) **Reddit:** REI ($\rho = -0.196$)



(d) **OpenSubtitles:** SC w.r.t. Utt.1 ($\rho = -0.330$)



(e) **OpenSubtitles:** SC w.r.t. Utt.2 ($\rho = -0.341$)



(f) **OpenSubtitles:** REI ($\rho = -0.319$)

Figure 5.4: Scatter plots illustrating correlation between automated metrics and human judgment (Pearson correlation coefficient is reported in the brackets).

5.4 Comparing Datasets

Finally, we investigate the impact of the training datasets on the quality of the responses generated by THRED and all baselines. In particular, we contrast the two datasets in terms of human judgment and the automated metrics

Method	OpenSubtitles	Reddit
Human MER	1.74	**2.00**
Coherence in Utt.1	0.642	**0.631**
Coherence in Utt.2	0.629	**0.601**
REI	0.667	**0.546**

Table 5.6: Mean over metrics per dataset to fare Reddit against OpenSubtitles. According to t-test (p-value < 0.001), the models elicit more informative and diverse responses when trained on Reddit, compared to OpenSubtitles.

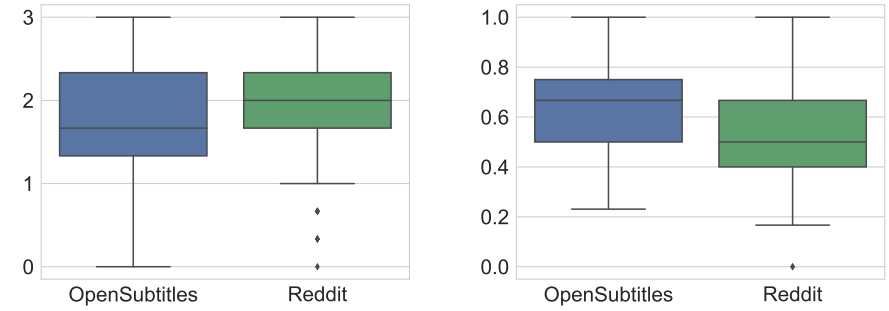
among all the models. Regarding human assessment, we took the Mean Evaluation Rating (MER) per response in the test data to draw the comparison between the datasets.

As demonstrated in Table 5.6 and Figure 5.5, the human evaluators scored generated responses from the Reddit dataset higher than utterances generated from the OpenSubtitles dataset, which is true not only in THRED, but in all models. The results in Figure 5.5 complement what we found in Table 5.6 in which only the mean is reported per metric.

Consequently, we can infer that the response quality depends also on the quality of the input data. This largely stems from the weak assumption, as stated in Chapter 3, for spotting a conversation in OpenSubtitles. In dealing with two-turn dialogues, such presumption may seem valid, whereas in multi-turn dialogues, it can aggravate the quality of conversations.

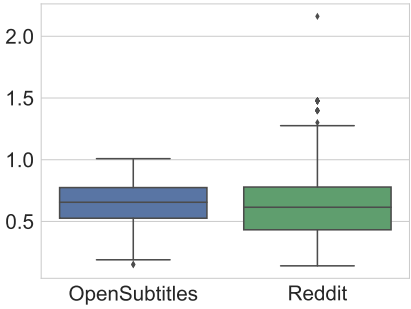
5.5 Conclusion

In this chapter, we evaluated the effectiveness of THRED method when tested on different datasets. Further, we presented two automated metrics to evaluate the response based on the conversation context: Semantic Coherence (SC) and Response Echo Index (REI). Our analysis suggests that the proposed model significantly outperforms the baselines in terms of human judgment and the proposed metrics. We demonstrated that human evaluation conforms reasonably with these metrics. Moreover, the results showcase that the Reddit dataset can be a better alternative than the OpenSubtitles dataset for training future dialogue systems.

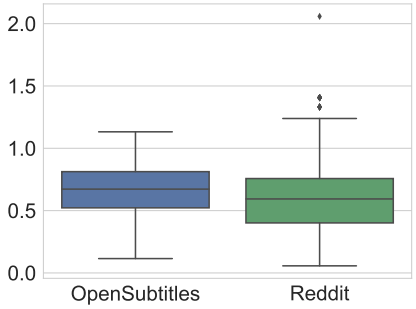


(a) Human MER

(b) REI



(c) SC w.r.t. Utt.1



(d) SC w.r.t. Utt.2

Figure 5.5: Box plots demonstrating the comparison between OpenSubtitles and Reddit. The metrics are calculated for all models in the cherry-picked data (150 samples for OpenSubtitles and 150 samples for Reddit).

As future work in the direction of evaluating open-ended dialogue systems, we seek a model that is capable of assessing the long-term influence of multiple responses in an ongoing dialogue. Such evaluation could be very challenging compared to the task of evaluating a single response. Nonetheless, we view our evaluation procedure as a good step toward this goal but more significant progress is expected to be made in this path.

Context	Generated Responses
(Reddit) sanctions are an act of war → why do you think that ?	**THRED:** because it's really a theory that supports terrorism . and this has an effect on the idea of a regime that isn't the same as a government (*Excellent, Excellent, Good*) **HRED:** because the war is n't a war . it 's a war . (*Poor, Poor, Poor*) **Seq2Seq:** because it 's an unpopular opinion , and that 's why it 's a bad thing to say . (*Good, Excellent, Poor*) **TA-Seq2Seq:** because it's a war . (*Good, Excellent, Poor*)
(Reddit) no relation to bitcoin . → actually , major relation to bitcoin . woman was ceo of a large company that does business with bitcoin .	**THRED:** bitcoin is not an economic system . if bitcoins were a direct source of income they would be worth going to . (*Excellent, Excellent, Good*) **HRED:** no , no , no . no . no . no . no . (*Poor, Good, Poor*) **Seq2Seq:** do you think i should have said " what is wrong with you " ? (*Poor, Good, Poor*) **TA-Seq2Seq:** it 's not a big deal . it 's a business . (*Good, Excellent, Good*)
(OpenSubtitles) the giant observatory in hawaii are like time machines capturing the faint light that has travelled towards us through all of cosmic history → the deeper they look into space the farther back they see in time	**THRED:** as the stars move into space they see the rising sun and the stars rise above the clouds (*Excellent, Good, Excellent*) **HRED:** this is the first time we have ever seen (*Poor, Poor, Bad*) **Seq2Seq:**the stars and stars of the moon are the stars and stars of the stars and stars and stars and stars and stars and stars (*Poor, Bad, Bad*) **TA-Seq2Seq:** they have been searching for a long time for a long time (*Good, Poor, Good*)
(OpenSubtitles) and closer to home in baltimore prominent psychiatrist and the author of a newly released book on insanity disappeared mysteriously last night after a lecture at the university → former mental patient james cole is wanted for questioning regarding dr cole 's disappearance	**THRED:** authorities warn that cole has a history of violence and advise anyone to notify authorities (*Excellent, Excellent, Excellent*) **HRED:** they have been unable to identify him (*Good, Poor, Poor*) **Seq2Seq:**cole cole is cole is cole(*Poor, Bad, Bad*) **TA-Seq2Seq:** authorities warn that cole has been kidnapped (*Poor, Good, Good*)

Table 5.7: 4 cherry-picked responses out of 300 conversations generated by all models along with human judgments in the brackets. The blue arrow specifies a dialogue turn exchange and the highlighted words in red represent the topic words acquired from the pre-trained LDA model.

Chapter 6

Conclusion

Dialogue systems have exploded in popularity in the recent years by showing a great promise in interacting with humans via natural language text. In particular, data-driven dialogue systems have attracted more attention thanks to the advances made in deep learning techniques. More specifically, neural Seq2Seq models have gained a big success in generating grammatically correct sentences by sampling words sequentially conditioned on previous words. However, despite significant progress in these models, they still suffer from several problems: (1) they tend to produce generic responses that carry little information; (2) they do not account for long conversation history as they lack explicit long-term memory and (3) they lack a consistent personality [60] as dialogue systems are usually trained over dialogues with different speakers [135].

In this dissertation, we have addressed the problem of improving the quality of responses by making the dialogue more diverse, contextual and fluent. We have introduced Topical Hierarchical Recurrent Encoder Decoder THRED devoted to generating topically consistent responses in a multi-turn general-purpose conversations. The model build upon the Seq2Seq model to condition the responses on previous utterances and on topical information. During encoding, our model maps the dialogue history utterances into fixed-length vector representations and acquires topic words from a pre-trained LDA model. Then, an utterance-level encoder is added atop the traditional word-level encoder to account for conversation history. In decoding, each word is generated

based on both the conversation history and the topical words through a hierarchical joint attention mechanism. A modified generation probability is then exploited to bias the model towards generating topic words in the responses.

Moreover, we have proposed two quantitative metrics for measuring the quality of the generated responses: Semantic Coherence and Response Echo Index. While the former measures the capability of the model to generate plausible responses which can be consistent with the context and maintain the topic of the conversation, the latter assesses how much THRED is able to generate unique responses which are measurably distant from the input dataset.

Besides, we have presented a clean and a well parsed Reddit dataset for training conversational models. The corpus is composed of 35M dialogues harvested from popular English subreddits that discuss topics like education, business and news.

Our results demonstrate that THRED outperforms significantly the baselines in terms of quantitative metrics and human judgment. By testing various models on the collected dataset, we were able to show that responses tend to be more engaging and interesting. We exhibit that Reddit dataset can be considered as a useful resource for training future conversational systems. While we show that the new metrics represent a reasonable diagnostic tool for automatically evaluating the quality of the responses, more advanced works are expected to be done in this area.

We suggest investigating the following avenues for future work:

Prior Knowledge: Typically, when conversing, we need prior knowledge about the person one is talking to (age, gender, speaking style, background infroemation, etc), about the place or about the event happening. Such background information has a great impact on the conversation flow and consistency [31], [138]. As discussed in Section **??**, most conversational datasets are derived either from social media websites or from movie scripts which usually lack detailed information about the speakers background and emotions. This presents a challenging problem for Seq2Seq models which have to account for external events and knowledge to closely imitate human behaviour in conver-

sations.

Logic and reasoning: There is a need for semantic understanding that is not necessarily present in current deep learning models. We need to resolve ambiguity in natural language and to add useful structure to the data. Consider the following conversation:

Speaker A: *Are you going to travel with us to the US?*

Speaker B: *Sorry, I have exams next week.*

Here, as humans, we can understand that Speaker B cannot travel to the US because he has exams. There is a chain of reasoning that dialogue systems have to follow to understand fully the semantic of the utterances. Speaker B has exams next week $\rightarrow$ he needs to prepare $\rightarrow$ he will be busy $\rightarrow$ he cannot travel to the US. Such reasoning can be easily done by humans but it is very difficult to transfer this capability to machines. Manually encoding all the reasoning steps is time consuming and labor-intensive and could be unfeasible. Ideally, we would like to build a dialogue system that would be capable of learning implicit reasoning chains automatically from the data.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016. 56

[2] D. Ameixa, L. Coheur, and R. A. Redol, "From subtitles to human interactions: Introducing the subtle corpus," Tech. rep., INESC-ID (November 2014), Tech. Rep., 2013. 40

[3] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2425–2433. 13

[4] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014. 13, 20, 23, 52, 56

[5] L. Bahl, P. Brown, P. De Souza, and R. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86.*, IEEE, vol. 11, 1986, pp. 49–52. 31

[6] R. E. Banchs, "Movie-dic: A movie dialogue corpus for research and development," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, Association for Computational Linguistics, 2012, pp. 203–207. 40

[7] S. Banerjee and A. 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. 35, 36

[8] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003. 14, 16

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003. 24, 50

[10] D. G. Bobrow, R. M. Kaplan, M. Kay, D. A. Norman, H. Thompson, and T. Winograd, "Gus, a frame-driven dialog system," *Artificial intelligence*, vol. 8, no. 2, pp. 155–173, 1977. 34

[11] A. Bordes, Y.-L. Boureau, and J. Weston, "Learning end-to-end goal-oriented dialog," *arXiv preprint arXiv:1605.07683*, 2016. 3

[12] ——, "Learning end-to-end goal-oriented dialog," *international conference on learning representations*, 2017. 34

[13] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *arXiv preprint arXiv:1511.06349*, 2015. 37

[14] D. Britz, A. Goldie, T. Luong, and Q. Le, "Massive exploration of neural machine translation architectures," *arXiv preprint arXiv:1703.03906*, 2017. 20

[15] P. F. Brown, "The acoustic-modeling problem in automatic speech recognition.," CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE, Tech. Rep., 1987. 31

[16] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. L. U. Limtiaco, R. S. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, C. Tar, Y.-h. Sung, B. Strope, and R. Kurzweil, "Universal sentence encoder," *arXiv preprint arXiv:1803.11175*, 2018. 60

[17] T. L. Chartrand and J. A. Bargh, "The chameleon effect: The perception–behavior link and social interaction.," *Journal of personality and social psychology*, vol. 76, no. 6, p. 893, 1999. 44

[18] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," *arXiv preprint arXiv:1601.06733*, 2016. 19

[19] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "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)*, 2014, pp. 1724–1734. 52

[20] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 93–98. 20

[21] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," *arXiv preprint arXiv:1412.1602*, 2014. 23

[22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014. 19

[23] K. M. Colby, S. Weber, and F. D. Hilf, "Artificial paranoia," *Artificial Intelligence*, vol. 2, no. 1, pp. 1–25, 1971. 26

[24] C. Danescu-Niculescu-Mizil and L. Lee, "Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs," in *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, Association for Computational Linguistics, 2011, pp. 76–87. 4, 41

[25] M. Davies, *The corpus of american soap operas: 100 million words, 2001–2012*, 2012. 41

[26] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, p. 391, 1990. 13

[27] A. H. F. Dinevari, "Towards the implementation of an intelligent software agent for the elderly," Master's thesis, University of Alberta, 2017. 2

[28] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *journal of artificial intelligence research*, vol. 22, pp. 457–479, 2004. 13

[29] D. A. Field, "Laplacian smoothing and delaunay triangulations," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 4, no. 6, pp. 709–712, 1988. 15

[30] P. Forchini, *Movie language revisited. Evidence from multi-dimensional analysis and corpora*. Peter Lang, 2012. 39, 40

[31] M. Ghazvininejad, C. Brockett, M.-W. Chang, B. Dolan, J. Gao, W.-t. Yih, and M. Galley, "A knowledge-grounded neural conversation model," *arXiv preprint arXiv:1702.01932*, 2017. 74

[32] Z. He, S. Gao, L. Xiao, D. Liu, H. He, and D. Barber, "Wider and deeper, cheaper and faster: Tensorized lstms for sequence learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 1–11. 58

[33] M. Henderson, B. Thomson, and J. D. Williams, "The second dialog state tracking challenge," in *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 2014, pp. 263–272. 3

[34] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012. 13

[35] S. Hochreiter and J. Schmidhuber, "Lstm can solve hard long time lag problems," in *Advances in neural information processing systems*, 1997, pp. 473–479. 19

[36] M. Hoffman, F. R. Bach, and D. M. Blei, "Online learning for latent dirichlet allocation," in *advances in neural information processing systems*, 2010, pp. 856–864. 53

[37] C. L. Isbell, M. Kearns, D. Kormann, S. Singh, and P. Stone, "Cobot in lambdamoo: A social statistics agent," in *AAAI/IAAI*, 2000, pp. 36–41. 27

[38] M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher, and H. Daumé III, "A neural network for factoid question answering over paragraphs," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 633–644. 13

[39] S. Jafarpour, C. J. Burges, and A. Ritter, "Filter, rank, and transfer the knowledge: Learning to chat," *Advances in Ranking*, vol. 10, pp. 2329–9290, 2010. 27

[40] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013, vol. 112. 43

[41] S. Jean, O. Firat, K. Cho, R. Memisevic, and Y. Bengio, "Montreal neural machine translation systems for wmt'15," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 2015, pp. 134–140. 23

[42] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *International Conference on Machine Learning*, 2015, pp. 2342–2350. 19

[43] D. Jurafsky, *Speech & language processing*. Pearson Education India, 2000. 12, 15, 34

[44] D. Jurafsky and J. H. Martin, *Speech and language processing*. Pearson London, 2014, vol. 3. 12

[45] F. Jurčıček, S. Keizer, M. Gašić, F. Mairesse, B. Thomson, K. Yu, and S. Young, "Real user evaluation of spoken dialogue systems using amazon mechanical turk," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011. 39

[46] J. Kalbfleisch and J. F. Lawless, "The analysis of panel data under a markov assumption," *Journal of the American Statistical Association*, vol. 80, no. 392, pp. 863–871, 1985. 15

[47] N. Kalchbrenner, I. Danihelka, and A. Graves, "Grid long short-term memory," *arXiv preprint arXiv:1507.01526*, 2015. 19

[48] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models.," in *AAAI*, 2016, pp. 2741–2749. 20

[49] D. Kinga and J. B. Adam, "A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015. 11, 57

[50] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, IEEE, vol. 1, 1995, pp. 181–184. 15

[51] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, *et al.*, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, Association for Computational Linguistics, 2007, pp. 177–180. 13

[52] I. de Kok, D. Heylen, and L.-P. Morency, "Speaker-adaptive multimodal prediction model for listener responses," in *Proceedings of the 15th ACM on International conference on multimodal interaction*, ACM, 2013, pp. 51–58. 44

[53] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.," *Psychological review*, vol. 104, no. 2, p. 211, 1997. 13

[54] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998. 13

[55] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015. 8, 43, 44

[56] H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky, "Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task," in *Proceedings of the fifteenth conference on computational natural language learning: Shared task*, Association for Computational Linguistics, 2011, pp. 28–34. 13

[57] O. Lemon, K. Georgila, J. Henderson, and M. Stuttle, "An isu dialogue system exhibiting reinforcement learning of dialogue policies: Generic slot-filling in the talk in-car system," in *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*, Association for Computational Linguistics, 2006, pp. 119–122. 34

[58] A. Leuski and D. Traum, "Npceditor: Creating virtual human dialogue using information retrieval techniques," *Ai Magazine*, vol. 32, no. 2, pp. 42–56, 2011. 27

[59] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," *arXiv preprint arXiv:1510.03055*, 2015. 3, 4, 13, 20, 31, 47, 59, 6

[60] J. Li, M. Galley, C. Brockett, G. P. Spithourakis, J. Gao, and B. Dolan, "A persona-based neural conversation model," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 994–1003. 13, 28, 33, 34, 39, 40, 73

[61] J. Li, W. Monroe, and D. Jurafsky, "A simple, fast diverse decoding algorithm for neural generation," *arXiv preprint arXiv:1611.08562*, 2016. 23

[62] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, "Deep reinforcement learning for dialogue generation," *arXiv preprint arXiv:1606.01541*, 2016. 33, 56

[63] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004. 35, 36, 59

[64] C.-W. Liu, R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, and J. Pineau, "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*, 2016, pp. 2122–2132. 36, 59

[65] R. Lowe, M. Noseworthy, I. V. Serban, N. Angelard-Gontier, Y. Bengio, and J. Pineau, "Towards an automatic turing test: Learning to evaluate dialogue responses," *arXiv preprint arXiv:1708.07149*, 2017. 4, 37, 59, 68

[66] R. Lowe, N. Pow, I. V. Serban, and J. Pineau, "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*, 2015, pp. 285–294. 13, 41, 42

[67] M.-T. Luong, Q. V. Le, I. Sutskever, O. Vinyals, and L. Kaiser, "Multi-task sequence to sequence learning," *arXiv preprint arXiv:1511.06114*, 2015. 20

[68] M.-T. Luong and C. D. Manning, "Achieving open vocabulary neural machine translation with hybrid word-character models," *arXiv preprint arXiv:1604.00788*, 2016. 20

[69] M.-T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba, "Addressing the rare word problem in neural machine translation," *arXiv preprint arXiv:1410.8206*, 2014. 20

[70] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1412–1421. 57

[71] C. Machinery, "Computing machinery and intelligence-am turing," *Mind*, vol. 59, no. 236, p. 433, 1950. 1

[72] K. W. Mathewson and P. Mirowski, "Improvised comedy as a turing test," *CoRR*, vol. abs/1711.08819, 2017. arXiv: `1711.08819`. [Online]. Available: `http://arxiv.org/abs/1711.08819`. 24, 49

[73] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943. 9

[74] T. Mikolov, M. Karafiát, L. Burget, J. Černock, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010. 3, 17, 29

[75] T. Mikolov, S. Kombrink, L. Burget, J. Černock, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, IEEE, 2011, pp. 5528–5531. 17

[76] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119. 14

[77] T. Mikolov and G. Zweig, "Context dependent recurrent neural network language model.," *SLT*, vol. 12, pp. 234–239, 2012. 17

[78] P. Mirowski, S. Chopra, S. Balakrishnan, and S. Bangalore, "Feature-rich continuous language models for speech recognition," in *Spoken Language Technology Workshop (SLT), 2010 IEEE*, IEEE, 2010, pp. 241–246. 24, 49

[79] V. Mnih, N. Heess, A. Graves, *et al.*, "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212. 23

[80] S. Möller, R. Englert, K. Engelbrecht, V. Hafner, A. Jameson, A. Oulasvirta, A. Raake, and N. Reithinger, "Memo: Towards automatic usability evaluation of spoken dialogue services by user error simulations," in *Ninth International Conference on Spoken Language Processing*, 2006. 35

[81] N. Mrkšić, D. O. Séaghdha, B. Thomson, M. Gašić, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, "Multi-domain dialog state tracking using recurrent neural networks," *arXiv preprint arXiv:1506.07190*, 2015. 34

[82] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang, *et al.*, "Abstractive text summarization using sequence-to-sequence rnns and beyond," *arXiv preprint arXiv:1602.06023*, 2016. 13, 20

[83] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "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. 13, 35, 37, 59

[84] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543. 14, 51

[85] I. Porteous, D. Newman, A. Ihler, A. Asuncion, P. Smyth, and M. Welling, "Fast collapsed gibbs sampling for latent dirichlet allocation," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2008, pp. 569–577. 25

[86] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *IN PROCEEDINGS OF THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS*, 2010, pp. 45–50. 58

[87] N. Reithinger, R. Engel, M. Kipp, and M. Klesen, "Predicting dialogue acts for a speech-to-speech translation system," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, IEEE, vol. 2, 1996, pp. 654–657. 34

[88] A. Ritter, C. Cherry, and W. 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. 27, 41, 42

[89] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951. 11

[90] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958. 9

[91] A. Roy, C. Guinaudeau, H. Bredin, and C. Barras, "Tvd: A reproducible and multiply aligned tv series dataset.," in *LREC*, 2014, pp. 418–425. 41

[92] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985. 12, 57

[93] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," *arXiv preprint arXiv:1509.00685*, 2015. 13, 20

[94] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015. 20

[95] I. V. Serban, R. Lowe, P. Henderson, L. Charlin, and J. Pineau, "A survey of available corpora for building data-driven dialogue systems," *arXiv preprint arXiv:1512.05742*, 2015. 28, 29, 31, 39, 42–44, 57

[96] I. V. Serban, A. Sordoni, Y. Bengio, A. C. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models.," in *AAAI*, 2016, pp. 3776–3784. 3, 20, 28, 30, 40, 56

[97] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. C. Courville, and Y. Bengio, "A hierarchical latent variable encoder-decoder model for generating dialogues.," in *AAAI*, 2017, pp. 3295–3301. 32, 44, 57, 60

[98] L. Shao, S. Gouws, D. Britz, A. Goldie, B. Strope, and R. Kurzweil, "Generating high-quality and informative conversation responses with sequence-to-sequence models," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2210–2219. 23, 33, 44, 65

[99] A. Show, "Tell: Neural image caption generation with visual attention," *Kelvin Xu et. al.. arXiv Pre-Print*, 2015. 23

[100] W. M. Soon, H. T. Ng, and D. C. Y. Lim, "A machine learning approach to coreference resolution of noun phrases," *Computational linguistics*, vol. 27, no. 4, pp. 521–544, 2001. 13

[101] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan, "A neural network approach to context-sensitive generation of conversational responses," *arXiv preprint arXiv:1506.06714*, 2015. 13, 29

[102] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting.," *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014. 57

[103] A. Stolcke, "Srilm-an extensible language modeling toolkit," in *Seventh international conference on spoken language processing*, 2002. 15

[104] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. V. Ess-Dykema, and M. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Computational linguistics*, vol. 26, no. 3, pp. 339–373, 2000. 34

[105] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112. 3, 20, 29, 30

[106] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *arXiv preprint arXiv:1503.00075*, 2015. 19

[107] L. Theis, A. van den Oord, and M. Bethge, "A note on the evaluation of generative models," *international conference on learning representations*, 2016. 63

[108] J. 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. 40

[109] ——, "Parallel data, tools and interfaces in opus.," in *LREC*, vol. 2012, 2012, pp. 2214–2218. 39, 44

[110] K. Torkkola, "Linear discriminant analysis in document classification," in *IEEE ICDM Workshop on Text Mining*, 2001, pp. 800–806. 24

[111] A. Venkatesh, C. Khatri, A. Ram, F. Guo, R. Gabriel, A. Nagar, R. Prasad, M. Cheng, B. Hedayatnia, A. Metallinou, R. Goel, S. Yang, and A. Raju, "On evaluating and comparing conversational agents.," *arXiv preprint arXiv:1801.03625*, 2018. 4, 60, 68

[112] O. Vinyals, Ł. Kaiser, T. Koo, S. Petrov, I. Sutskever, and G. Hinton, "Grammar as a foreign language," in *Advances in Neural Information Processing Systems*, 2015, pp. 2773–2781. 20, 44

[113] O. Vinyals and Q. Le, "A neural conversational model," *arXiv preprint arXiv:1506.05869*, 2015. 13, 20, 28, 43, 45

[114] M. A. Walker, D. J. Litman, C. A. Kamm, and A. Abella, "Paradise: A framework for evaluating spoken dialogue agents," in *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, 1997, pp. 271–280. 35

[115] V. Warnke, R. Kompe, H. Niemann, and E. Nöth, "Integrated dialog act segmentation and classification using prosodic features and language models," in *Fifth European Conference on Speech Communication and Technology*, 1997. 34

[116] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966. 26

[117] T.-H. Wen, M. Gasic, N. Mrksic, L. M. Rojas-Barahona, P.-H. Su, S. Ultes, D. Vandyke, and S. Young, "Conditional generation and snapshot learning in neural dialogue systems," *arXiv preprint arXiv:1606.03352*, 2016. 34

[118] T.-H. Wen, M. Gasic, N. Mrksic, P.-H. Su, D. Vandyke, and S. Young, "Semantically conditioned lstm-based natural language generation for spoken dialogue systems," *arXiv preprint arXiv:1508.01745*, 2015. 34

[119] T.-H. Wen, Y. Miao, P. Blunsom, and S. Young, "Latent intention dialogue models," *arXiv preprint arXiv:1705.10229*, 2017. 34

[120] J. Weston, A. Bordes, S. Chopra, A. M. Rush, B. van Merriënboer, A. Joulin, and T. Mikolov, "Towards ai-complete question answering: A set of prerequisite toy tasks," *arXiv preprint arXiv:1502.05698*, 2015. 13

[121] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *Neurocomputing: foundations of research*, pp. 123–134, 1988. 9

[122] J. Wieting, M. Bansal, K. Gimpel, and K. Livescu, "Towards universal paraphrastic sentence embeddings," *arXiv preprint arXiv:1511.08198*, 2015. 60

[123] J. D. Williams and S. Young, "Partially observable markov decision processes for spoken dialog systems," *Computer Speech & Language*, vol. 21, no. 2, pp. 393–422, 2007. 34

[124] S. Wiseman and A. M. Rush, "Sequence-to-sequence learning as beam-search optimization," *arXiv preprint arXiv:1606.02960*, 2016. 20

[125] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016. 20, 58, 61

[126] C. Xing, W. Wu, Y. Wu, J. Liu, Y. Huang, M. Zhou, and W.-Y. Ma, "Topic aware neural response generatio.," in *AAAI*, 2017, pp. 3351–3357. 24, 32, 49, 54, 56, 57

[127] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *International Conference on Machine Learning*, 2016, pp. 2397–2406. 13

[128] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057. 19

[129] R. Yan, Y. Song, and H. Wu, "Learning to respond with deep neural networks for retrieval-based human-computer conversation system," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, ACM, 2016, pp. 55–64. 27

[130] Z. Yan, N. Duan, J. Bao, P. Chen, M. Zhou, Z. Li, and J. Zhou, "Docchat: An information retrieval approach for chatbot engines using unstructured documents," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 516–525. 27

[131] S. Young, "The statistical approach to the design of spoken dialogue systems," 2002. 34

[132] S. J. Young, "Probabilistic methods in spoken–dialogue systems," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 358, no. 1769, pp. 1389–1402, 2000. 28

[133] S. Young, M. Gašić, B. Thomson, and J. D. Williams, "Pomdp-based statistical spoken dialog systems: A review," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1160–1179, 2013. 3, 59

[134] W. Zeng, W. Luo, S. Fidler, and R. Urtasun, "Efficient summarization with read-again and copy mechanism," *arXiv preprint arXiv:1611.03382*, 2016. 20

[135] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?" *arXiv preprint arXiv:1801.07243*, 2018. 33, 73

[136] T. Zhao, R. Zhao, and M. Eskenazi, "Learning discourse-level diversity for neural dialog models using conditional variational autoencoders," *arXiv preprint arXiv:1703.10960*, 2017. 50

[137] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," in *European Conference on Information Retrieval*, Springer, 2011, pp. 338–349. 32, 54

[138] W. Zhu, K. Mo, Y. Zhang, Z. Zhu, X. Peng, and Q. Yang, "Flexible end-to-end dialogue system for knowledge grounded conversation," *arXiv preprint arXiv:1709.04264*, 2017. 74