TOWARDS ADAPTING COGNITIVE ARCHITECTURES FOR KNOWLEDGEABLE & PERSONALIZED DIALOGUE SYSTEMS

by

Sashank Santhanam

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2021

Approved by:

Dr. Samira Shaikh

Dr. Wlodek Zadrozny

Dr. Nicholas Davis

Dr. Minwoo Lee

Dr. Mohamed Shehab

ABSTRACT

SASHANK SANTHANAM. Towards Adapting Cognitive Architectures for Knowledgeable & Personalized Dialogue Systems. (Under the direction of DR. SAMIRA SHAIKH)

State-of-the-art conversational agents have advanced significantly in conjunction with the use of large transformer-based language models. However, even with these rapid advancements, the current generation of conversational agents suffers from three major problems: (i) long-term context modeling; (ii) producing informative and factually accurate responses; (iii) robust evaluation of the NLG systems. Our work tackles these three gaps: (i) To address the issue of long-term context modeling, we present a novel end-to-end approach inspired by neurocognitive memory processes. We also implement a novel action selection mechanism that helps identify the relevant utterances containing salient information from long-term memory to working memory to better incorporate the context of the conversation during the generation process than state-of-the-art systems. (ii) To integrate knowledge into conversational agents, we also propose a dialog framework that incorporates both local knowledge as well as users' past dialogues to generate high-quality personalized conversations. Using our framework, we demonstrate that incorporating local knowledge can largely improve *informativeness*, *coherency* and *realisticness* measures using human evaluations. However, even with these advancements, we find that knowledge grounded conversation models are prone to hallucinations. We address this issue by proposing a new dataset called "CONV-FEVER" to build a fact consistency detector. We show that our detector outperforms the current SOTA and can be integrated with existing models to increase the factual consistency of the knowledge grounded models. (iii) In the last part of this thesis, we focus on the aspect of the impact of experiment design in conversational AI systems by conducting two large-scale studies. In the first study, we compare four different experimental designs and study how each experiment design affects the quality of outputs obtained from the human evaluation. In the second study, we study the impact of cognitive biases particularly anchoring bias, and demonstrate its impact on human evaluation of NLG systems.

ACKNOWLEDGMENTS

First and foremost, I thank my advisor, Dr. Samira Shaikh, for constantly motivating and encouraging me through the Ph.D. journey and for the countless conversations that helped me shape my research and life in general. I am also grateful to my fellow committee members Dr. Minwoo Lee, Dr. Nicholas Davis, Dr. Wlodek Zadrozny, and Dr. Mohamed Shehab for the guidance and support by being involved in various projects during my Ph.D.

My journey would not have been possible without the support of my parents, my brother, and my grandparents. They have been a constant source of support and have helped me a lot during the journey for which I am extremely grateful.

I thank my collaborators at IHMC, RPI & Suny Albany, namely, Bonnie Dorr, Archna Bhatia, Adam Dalton, Brodie Mather, Alan Zemel, Amir Masoumzadeh, Tomek Strzalkowski, Younes Karimi, and Sreekar Dhaduvai for the work done during the PANACEA project and for the interesting ideas that contributed in taking me research forward. I am also extremely grateful to my colleagues at the Charlotte Visualization Center Lab members, especially, Alireza Karduni, Ryan Wesslem, Wenwen Dou, and Isaac Cho for collaborating with me in the early stages of my Ph.D. journey. I would also like to thank my collaborators at NVIDIA and Amazon for providing me with interesting ideas and guidance during my internships.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Motivation

Language Generation is a sub-field of the field of Natural Language Processing(NLP), Artificial Intelligence(AI), and Cognitive Science(CS) that has been studied since the 1960s. Yet, it is still one of the major challenges that researchers in Natural Language Processing and Computational Linguistics face.

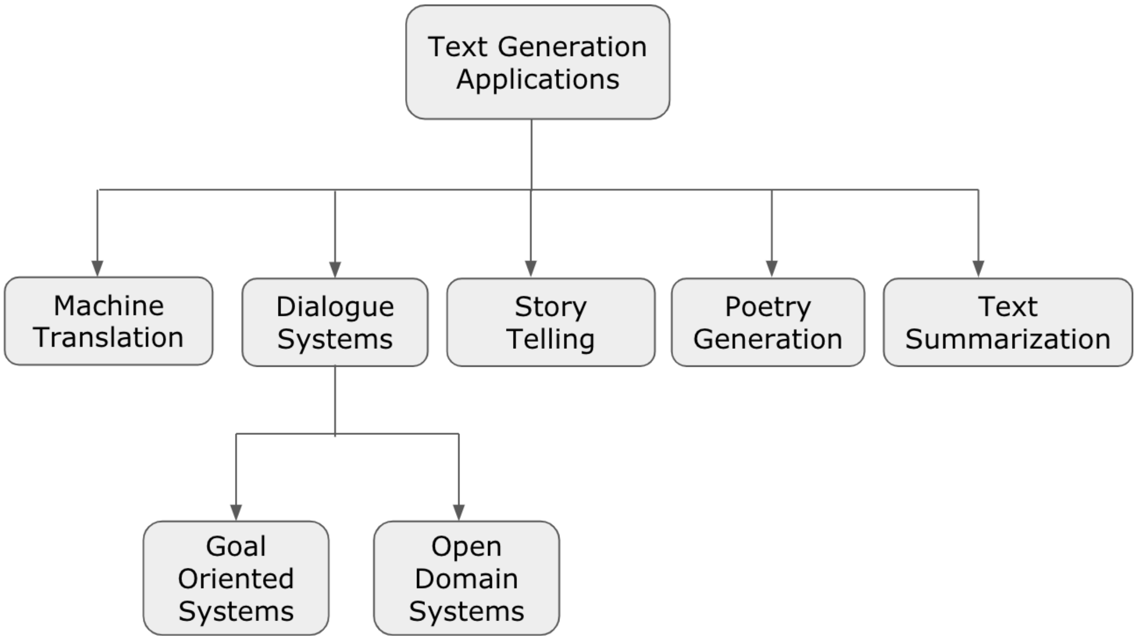


Figure 1: Overview of the different applications belong to sub field of language generation.

Some of the early success in the field of text generation was through the domain of dialogue systems by the building of systems like Eliza [217] and PARRY[1]. These

[1] https://en.wikipedia.org/wiki/PARRY

systems generated language through a set of rules. However, such rule-based systems were too constrained and brittle and cannot be generalized to produce a diverse set of responses.

The field of text generation systems shifted from traditional approaches to statistical approaches where the focus was on exploiting the patterns in data and building models to make a prediction based on the data it has seen. However, Mikolov *et al.* [144] argued that there has not been any significant progress in using statistical approaches as the means for the modeling language which led to his experimentation on using recurrent neural networks. This experimentation achieved state-of-the-art results which set the wheels in motion for neural networks becoming a model of choice for modeling sequential data like text. Neural Networks belong to a class of machine learning models that are capable of identifying patterns in text and identify features that help solve different problems related to computer vision, object recognition, image captioning, and speech recognition [201]. Another phenomenon that suited the rise of neural networks is the large number of corpora and significant computational resources that were available to be used. In the applications of language generation, neural networks have helped achieve state-of-the-art results in problems related to parsing machine translation [7], storytelling [73], dialogue systems [223, 225, 43] and poetry generation [238].

More recently the field of NLG has rapidly advanced with the development of transformer-based models [208]. Transformer models [208] and large transformer-based language models such as GPT, GPT-2, XLNet, BERT [163, 164, 229, 40] have helped achieve the SOTA performance across several natural language tasks. With these rapid advancements, transformers models are widely used across a wide range of NLG tasks including conversational agents.

1.2 Motivation

An analysis of the recent progress on open-domain conversational agents built using transformer models and LSTM models has shown these agents to be incapable of holding engaging, informative, and consistent conversations [173, 79]. In this section, I present various challenges that motivate my work in this dissertation:

1. One of the main challenges in conversational AI is to produce engaging responses and overcome the issue of producing **dull and generic responses**. Prior research has shown one possible cause for dull and generic responses might be due to their reliance on the last utterance in the dialogue history as contextual information [204]. We posit this problem as **modeling long-term context modeling**. To address this issue, we present a novel end-to-end model that is inspired by the cognitive science approach. This model provides the framework with which to conceptually and practically address both long-term memory and short-term memory (working memory) - to incorporate the longer context of conversation along with the immediate context. Besides, the model provides a novel action-selection mechanism acting as a bridge between long and short-term memory. To the best of our knowledge, our work is the first to use the Standard Model of Cognition to more closely tie the NLG system to the way human cognition works.

2. Another issue that affects conversational agents is the ability to produce **informa-**

tive responses that are grounded in knowledge. To produce informative responses, we propose a dialog framework that incorporates both local knowledge as well as users' past dialogues to generate high-quality conversations. We introduce an approach to build a dataset based on *Reddit* conversations, where outbound URL links are widely available in the conversations and the hyperlinked documents can be naturally included as local external knowledge. Using our framework and dataset, we demonstrate that incorporating local knowledge can largely improve *informativeness*, *coherency* and *realisticness* measure using human evaluations. However, qualitative analysis of responses reveals that these knowledge grounded models suffer from the issue of being faithful and factually consistent. Consistency in conversational agents is crucial to gain confidence and trust [79]. To reduce hallucinations in conversational agents, we introduce a new dataset "CONV-FEVER" that can be used to build a factual consistency detector that can be used to rerank responses and increase faithfulness in conversational AI.

3. Another possible reason for the conversation agents to be inconsistent and not coherent might be because the end-to-end approaches compress the notion of planning and generation into a single step and demanding too much from the network [147]. Traditional approaches to NLG incorporate a sequence of steps in the NLG system, including content determination, sentence planning, and surface realization [167, 169]. We investigate the impact of decoupling the generation process into separating planning and realization in open-domain dialogue and

compare it into end-to-end approaches and find that the approach produces better responses per automated metrics and detailed human evaluations.

4. An important component apart from developing models is to evaluate them. My last part of the dissertation focuses on addressing issues related to the human evaluation of conversational AI systems. Human evaluation is the primary source of evaluation since automated metrics show poor correlation with human ratings[154, 123]. We look at the evaluation of dialogue systems from the perspective of experiment design. We conduct two large-scale studies that study how different experiment designs affect the quality of the ratings obtained from the human evaluation.

1.3 Contributions

In this section, I summarize the high-level contributions in this thesis.

1. We present a novel architecture, that adapts a cognitive architecture called *Standard Model of Cognition* for augmenting traditional *seq2seq* systems. This novel architecture provides the model capable of identifying the right contextual utterances for dialogue systems to maintain context. We demonstrate this work in Section 3.

2. We present a knowledge grounded & personalized response generation framework that allows conversational AI to produce informative responses. Further, We address the issue of hallucination by releasing two new datasets that can be used to address the issue of factual inconsistency in knowledge grounded models. We demonstrate this work in Section 4.

3. We investigate the impact of separating planning and realization in open-domain dialogue and find that the approach produces better responses per automated metrics and detailed human evaluations. We demonstrate this work in Section 5.

4. From an evaluation perspective, we conduct two large-scale studies that are focused on experiment design. In the first study, we perform a systematic comparison of 3 different designs and demonstrate that continuous scales outperform discrete scales in obtaining more consistent ratings. In the second study, we study the impact of anchoring bias and show that anchoring bias is an important element that helps continuous scales outperform discrete scales. We demonstrate this work in Section 6.

1.4 Outline

This section provides an overview of the dissertation. we provide an overview of the work done in the area of language generation and traditional approaches that were used before the rise of deep learning models. In Section 2.4, we provide a summary of Dialogue Systems and identify the drawbacks of dialogue systems, that forms the backbone for the upcoming chapters (2).

In Chapter 3, we introduce a novel architecture for modeling context and generating dialogue based on the standard model of cognition. We find that our model outperforms the current state-of-the-art models in context identification which was verified through the use of human annotators and also finds a higher correlation with humans.

In Chapter 4, we introduce a dynamic personality and knowledge-powered conversational framework that is targeted towards using localized knowledge for conversational agents. Further, We also introduce a new dataset that helps tackle the issue of hallucination and factual consistency in conversational agents.

In Chapter 5, we study the effectiveness of decomposing the generation process into planning and generation phases moving away from an end-to-end approach. We study the efficacy of this approach against commonly used end-to-end models. Further, we also introduce a new novel NLU component which is used in the planning phase.

In Chapter 6, we show the impact of different experiment designs in the evaluation of NLG systems along with how cognitive biases, namely anchoring bias, impact the quality of ratings.

CHAPTER 2: RELATED WORK

2.1 Introduction

Deep Neural Networks are powerful machine learning models that have helped researchers achieve state-of-the-art performance on problems related to computer vision, object recognition, image captioning speech recognition, and other Natural Language Processing tasks [201]. One such problem that has received a lot of attention is natural language generation which is a fascinating but hard area of research [55] as it combines the fundamental aspects of artificial intelligence and cognitive science [169]. Natural Language Generation (NLG) is defined as "the sub-field of artificial intelligence and computational linguistics that is concerned with the construction of computer systems than can produce understandable texts in English or other human languages from some underlying non-linguistic representation of information that can be in the form of structured data or knowledge bases" by Reiter and Dale [169].

Some examples of NLG are domain-based systems that produce weather reports [1] and sports reports [14] using traditional approaches, including template-based methods or hand-crafted grammar rules. However, using template-based methods is not sustainable in the longer term as it creates constraints on making the systems more generalizable or produces variations in text generated without the presence of a human in the loop. With the rise in the amount of data being generated, researchers moved towards using neural network models to do language modeling and generating language as a probabilistic task of predicting the next word ignoring the cognitive aspect behind the process. Figure 2 provides an overview of the different approaches that have been used to solve the problem of language generation. In this chapter, we provide an overview of the traditional approaches to language generation (Section 2.2) before shifting towards deep learning approaches for language generation (Section 2.3). In Section 2.4, we provide a comprehensive overview in the area of dialogue systems and we conclude this chapter with an overview of the cognitive architectures (Section 2.5) that provides the base for upcoming chapters.

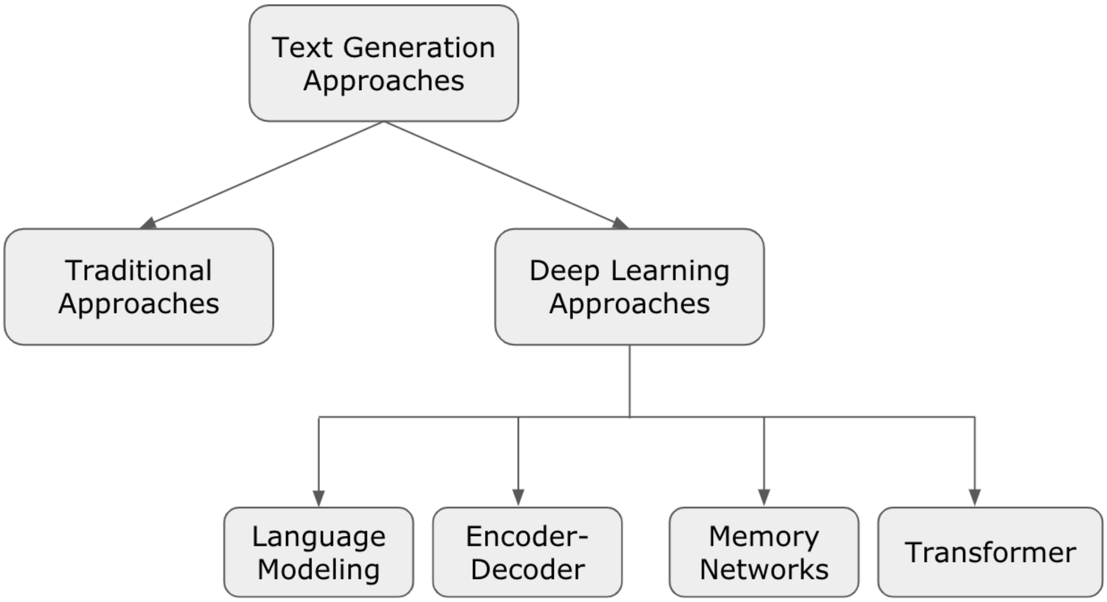


Figure 2: Taxonomy of approaches used in text generation process

2.2 Traditional Approaches to Text Generation

Traditional approaches to NLG followed the standard architecture (Figure 3[2]) that comprised of six components each performing an important task to generate a coherent

[2] https://tinyurl.com/ydgyawvw

output. Their architecture was motivated by the fact that there were many NLG systems for different applications but no well-defined architecture. Before the six-stage pipeline, Reiter introduced a simple three-stage pipeline of 1) content determination; 2) sentence planning, and 3) surface realization and named it the "consensus" architecture [167]. Cahill *et al.* conducted experiments and argued that the pipeline process is not detailed or generic and the architecture was too constrained. To overcome the issues, the authors suggested a finer architecture based on linguistic operations such as 1) lexicalization; 2) referring expression generation, and 3) aggregation [26]. One drawback of the architecture suggested by Cahill *et al.* was that no details were provided about how the systems get input and in what form. Reiter and Dale iterated on the initial architecture and suggested a new standard architecture for the NLG systems comprising of 4 tuples $\langle k, c, u, d \rangle$ where k is the knowledge source, c is the communicative model, u is the user model and d is the discourse theory [55] and the iterated model also implemented some of the aspects of Cahill's work into the architecture.

In the upcoming subsections, we explain the functionality of the six components and extensive research work that has been carried out to address that component.

2.2.1 Content Determination

Content Determination is the problem of deciding the domain of the information that is needed to generate text for a given input. Content determination is affected by communicative goals i.e Different communicative goals from different kinds of people from the same data may require different contents to be expressed by the system that

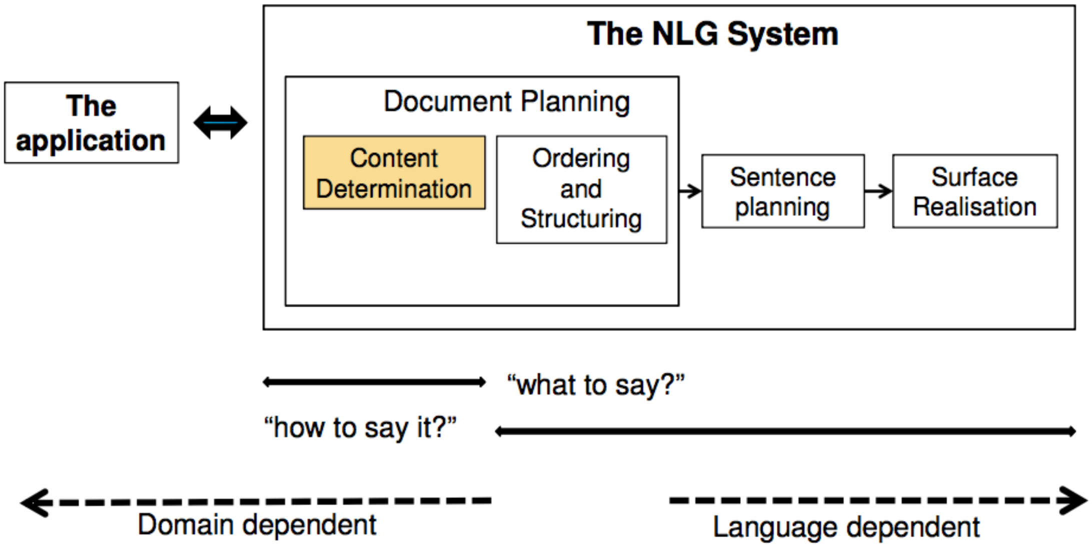


Figure 3: Three stage pipeline architecture proposed by Reiter and Dale

satisfies the parties involved. Content determination is affected by the expertise of the end-user and also by the content of the information source present in the system [169].

The problem of content determination has been approached from two different perspectives.

1. Schemas or templates

2. Statistical data-driven approaches

Schemas or Template-based content determination methods focused on generating content by an analysis of the corpora and they are prominent in tasks while are standardized like weather forecast systems like FOG [64] where rhetorical relations can be encoded as schemas or schemata [138]. The schemata are made up of *identification, constituency, attributive and contrastive.* Each component of the schemata is used to describe the different predicate patterns [138]. Schemas or Templates can be improved upon by using rule-based approaches. Rule-based approaches for the task of content determination have been used for the domain-specific systems where the implicit knowledge of the domain expert is used for more knowledge acquisition [170, 51]. Reiter *et al.* list the different techniques such as sorting, thinking aloud, expert revision for knowledge acquisition in the STOP system that generated personalized smoking-cessation leaflets [170].

With the availability of more data, the process of content determination became a data-driven process. Duboue and McKeown [46] developed a system that automated the process of producing constraints on every input and deciding if it should appear as a part of the output with the help of a two-stage process of exact matching and statistical selection, where the semantic data is clustered and text corresponding to each cluster is used to measure its degree of influence with regards to the other clusters. An alternative method was suggested by Barzilay and Lee [14] where content selection can be applied to domains where the knowledge base has not been provided by using a novel adaptation of Hidden Markov Models, in which the states correspond to the type of information characteristic to the domain of interest. Barzilay and Lapata [12] suggested another method along similar lines to the method suggested by Duboue and Mckeown, in which the content selection is treated as a collective classification problem by capturing the contextual dependencies between the input items.

Liang *et al.* [118] extended the work done by Barzilay and Lapata, by describing a probabilistic generative model that combines text segmentation and fact identification in a single unified framework using Hidden Markov Models. The generative model proposed is of three stages of selecting a set of records, identifying the fields from the records, and choosing a sequence of words from the fields and each stage is optimized using Expectation-Maximization(EM). The work done by Liang *et al.* proved instrumental in combining the process of content determination and linguistic realization into a unified framework. An example of this is the work done Angeli *et al.*[1] where the process of generation is broken down into a sequence of local decision and using a classifier on three types of decisions that include choosing the records from the database, choosing a subset of fields from records and choosing a template to render the generated text. However, Kim and Mooney identified a drawback with the method suggested by Liang *et al.* of just using a bag of words and a simple Hidden Markov Model and not considering the context-free linguistic syntax. To address this issue, Kim and Mooney used a generative model with hybrid trees which expresses correspondence between the word in natural language and grammatical structure (meaning representation) and Iterative generation strategy learning, a method similar to EM that iteratively improves probability to determine which event likely to be received as input from the human [87]. Another example of end to end generation is the work done by Konstas and Lapata, where the set of records are converted into probabilistic context-free grammar (PCFG) that describes the structure of the database and encode the grammar as a weighted hypergraph and generation process is based on process find the best derivation of the hypergraph [93]. In the next section, we will be covering document structuring, the next sub-problem of language generation.

2.2.2 Document Structuring

The second sub-problem specified by Reiter and Dale is a document or text structuring that is aimed at the process of determining the order in which the text is to be conveyed back to the user once the content is determined. Document Structuring and Content Determination are closely linked. A method that had a significant impact on the understanding of discourse relations was with the help of Rhetorical Structure Theory (RST) [132].

RST has four elements consisting of "relations" which identifies relationships between different parts of the text in the form of satellite and nuclei where nuclei represent the important part of the text and satellite represents the supplementary part of the text, "schemas" defines patterns in a part of the text can be analyzed with regards to other spans (nodes of a tree), "schemas application" and "structures" and helps in creating coherent texts [133].

Moore and Paris found issues with the RST when they tried to use the individual segments and rhetorical relations between segments to construct a text plan for their dialogue system and RST was not able to generate proper responses for follow-up questions [145]. Due to the problems with RST, Moore and Pollack suggested a two-level discourse analysis process. The first level is called "information level" which involves the relation conveyed between two sentences in discourse and the second level is called "intentional level" which deals with the discourse produced to effect change in the mental state of the participants [146]. Dimitromanolaki and Androutsopoulos use supervised machine learning that learns a new representation of document structuring tasks and apply this approach to the task of document structuring for a specific domain [42]. A lot of other researchers have interlinked the process of text structuring and content determination into a single one which has been described in the previous subsection.

2.2.3 Lexicalization

Lexicalization or the task of choosing the right words to express the contents of the message is the third sub-problem defined by Reiter and Dale. They broke down the task of lexicalization into two categories, namely, Conceptual Lexicalization and Expressive Lexicalization. Conceptual Lexicalization is defined as converting data into linguistically expressible concepts and Expressive Lexicalization is how lexemes available in a language can be used to represent a conceptual meaning [169]. Bangalore and Rambow [10] characterized the process of choosing the best lexeme to realize the meaning as a very hard task and they called this phase the syntactic phase. To solve the problem, Bangalore and Rambow suggested using a tree representation of the syntactic structure and independently hand-crafted grammar. One of the drawbacks of this method was not using a part of speech tagger and using a casual mechanism of making a union of all the synonyms from the synset.

Another issue that makes the task of lexicalization difficult is figuring out the vagueness, in terms of a crisp word meaning. The issue of vagueness with regards to adjectives was investigated by Kennedy and McNally [84], who did a semantic analysis of the predicates with the help of degree modifiers and parameterized along with two core features.

While the traditional approaches to NLG view the process of lexicalization as belonging to the sentence planning phase along with the process of sentence aggregation and Referring Expression Generation, however, recent research in NLG views lexicalization as the part of the linguistic realization phase [60].

2.2.4 Referring Expression Generation

Referring Expression Generation (REG) is the fourth sub-problem defined by Reiter and Dale and it is aggregated with the sentence planning phase of the architecture. REG is characterized by Reiter and Dale as the ability to produce a description of an entity and distinguish itself from the other domain entities [169]. An entity might be referred to in many different ways. For example, consider the following sentence, “Adrian arrived late to an event and he missed a majority of it”. There can be two ways in which an entity can be referred to. The first one is the “initial reference” (“Adrian” in the example) when an entity is brought into the discourse and the other is “subsequent reference” (“he” in the example) which refers to the entity after it is in the discourse [169]. The first step of the solution suggested by Reiter and Dale is to identify the type of reference for the target, such as pronoun or description or proper name. The identification of proper names is the easiest, while identification of pronoun can be based on a rule which depends on “if the target is referred to in the previous sentence and if the sentence contained no other entity of the same gender” [96].

Reiter and Dale evaluated the Gricean maxims in terms of conversational implicature and how efficiently the properties to be involved in the description can be computed with the help of the target and distractors [36].

There are multiple existing algorithms for the task of REG. The first one is an algorithm called **Full Brevity** that generates very short descriptions referring expressions by checking one different referring expression component one at a time. A major drawback of this method is that it is computationally expensive. An improvement over the Full Brevity was the **Greedy Heuristic** algorithm, which picks a property of target that rules out most of the distractors (they don't co-reference with the target) and adding that property to the description. This algorithm was later eclipsed in terms of performance by the **Incremental Algorithm** (IA). The Incremental Algorithm sequentially picks the properties and then rules out the distractors until a distinguishable expression is generated [36]. The drawback of all these algorithms was that the target was one object and the properties were relevant and not ambiguous or vague.

To address these drawbacks, Kees and Van Deemter explored how the incompleteness of the IA could be overcome with the help of a two-stage algorithm to generate boolean descriptions [206]. The first stage is the process of generalization of the IA by taking a union of the properties that help in singling out the target set and the next stage was to optimize the expressions produced [206, 96]. One of the issues was that Van Deemter failed to address the notion of vagueness which was addressed in the work done by Horacek[76]. Horacek introduced two complementary measures to increase the likelihood of object identification and representing the uncertainty in terms of probabilities and a triple of the object, attribute of the object, and value to the pair. The uncertainties introduced were p_k - the user is acquainted with the terms mentioned, p_p- the user can perceive the properties uttered, p_A - the user agrees with the applicability of the terms used, and with the help of the three probabilities, the probability of recognition p is calculated as the product of the three probabilities and this helps in distinguishing vagueness along with misinterpretation and ambiguity [76]. Later, Khan *et al.* addressed the issue of structural ambiguity in the coordinated phrases in the form of “Adjective Noun1 Noun1” where the issue was whether the adjective was associated with noun1 or noun2. Khan *et al.* conducted user studies and suggested how the generator can avoid these issues [86]. However, Engonopoulos and Koller had some issues with the idea of the algorithms generating distinguishing unique expressions and argues that there is a chance that the listeners might misunderstand the generated expression. To address their concerns, Engonopoulos and Koller proposed an algorithm to maximize the likelihood that a referring expression is understood by the user with the help of a probabilistic referring expression model $P(a|t)$, t refers to the expression and a to the object in the domain [53].

2.2.5 Sentence Aggregation

Sentence aggregation is characterized as the process of removing redundant information during the generation of discourse without losing any information and to produce text in a concise, fluid and readable manner [37]. Dalianis, in his survey, suggested that aggregation can be done in all the stages of the NLG process except during content determination and surface realization. Reiter and Dale marked this subproblem as belonging to the sentence planning or microplanning phase [169]. Reiter and Dale characterized the problem of aggregation to be closely interlinked with lexicalization as both deals with understanding the knowledge source and linguistic elements of words, phrases and sentences [169].

One of the initial approaches to tackle the problem of sentence aggregation was put forward by Cheng and Mellish by using Genetic Algorithms, where they used a constraint-based program with a preference function to evaluate the coherence of a text [30]. Walker *et al.*. [213] used a data-driven approach to overcome the issue of using a hand-crafted preference function used by Cheng *et al.* [30]. In their work, they used two phases; the first phase generated a large sample of sentences for input and the next phase ranked the sentences with the help of rules generated from training data. Barzilay and Lapata (2006) [13] presented an automatic method to learn the grouping constraints with the help of a parallel corpus of sentences and their corresponding database entries by looking at the number of attributes shared by the entries.

2.2.6 Linguistic Realization

Linguistic Realization was characterized by Reiter and Dale as the task of ordering different parts of a sentence and using the right morphology along with punctuation marks which are governed by rules of grammar to produce a syntactically and orthographically correct text [169]. Different NLG systems adopt different methods like template-based or abstract syntactic structures to represent the sentences internally which captures the grammatical knowledge within the linguistic realizer. In this section, we will be covering the three different approaches namely template-based, hand-coded grammar-based systems, and statistical approaches.

2.2.6.1 Hand coded grammar-based systems

Grammar-based NLG systems are systems that make their choice depending on the grammar of the language which can be manually written with the help of multilingual realizers like KPML, developed by Bateman in 1997 that depended on the systemic grammar that helps us understand the syntactic characteristic of a sentence [60, 15]. Another popular realizer is SURGE developed by Elhadad and Robin [51], which is based on functional unification formalism. Another popular realizer was called Halogen, which was introduced by Langkilde in 2002. This system uses a small set of hand-crafted grammar as a feature to generate alternative representations [104]. A downside of using these realizers is that they are complicated to use and have a steep learning curve for the users which made the researchers and the NLG community move towards simple realization engines.

2.2.6.2 Templates

Template-based NLG systems are systems that represent sentences as placeholder text and values are used to replace the placeholder and the basic template-based systems just replace the placeholder without further processing. Templates are often used in systems that require limited syntactic variability in their output. [168]. For example, consider the template "[$Person$] is leaving [*country*] " and in this scenario person and country act as templates whose values are then replaced by the system in the output. One of the issues with template-based NLG systems was the low quality of text generated.McRoy *et al.* suggested a method to overcome these issues with the help of declarative control expressions to augment the traditional templates. McRoy *et al.* addressed the issue of producing low-quality text by incorporating Attribute Grammar without slowing down the system. Van Deemter *et al.* argue that as new NLG systems have been developed, the difference between standard NLG systems and template-based systems have blurred as the modern systems use handcrafted grammars to help with realization [207].

Another disadvantage of using templates is the need for a knowledge expert to construct templates for the system [139, 60]. Angeli *et al.* used a probabilistic approach to learn temporal utterance and use compositional grammar to learn the rules for parsing time expressions [2] and Kondadadi *et al.*. used k means clustering to create template banks derived using named entity tagging and semantic analysis [92]. Despite the advantage of using template-based methods, most of the recent NLG systems have moved to a statistical-based approach.

2.2.6.3 Statistical Approaches

Statistical approaches have been used in NLG systems to reduce the manual effort of using handwritten grammar rules and to deal with large corpora to acquire probabilistic grammar to get better realizations of text. Langkilde (2000) was one of the seminal works and early papers to use a statistical approach to sentence generation. In his approach, Langkilde used corpus-based statistical knowledge, and with the help of small hand-crafted grammar, he was able to generate many different representations of sentences that were packed in the form of a forest of trees. Langkilde ranked each of the phrases by calculating a score which was decomposed into an internal and external score, former known to be context-independent and latter is context dependent[103].

The method introduced by Langkilde served as the base for subsequent research in this field.

Bangalore and Rambow suggested improvements to work done by Langkilde by introducing a tree-based model to improve to performance of the syntactic choice or the ranker method of Langkilde [10]. Cahill *et al.* presented a different method to rank and suggested using a log-linear ranking system, and they show that log-linear ranking ranks the correct solution considerably higher than the existing systems [25]. One major downside of the approaches specified by Langkilde; Banglore and Rambow; Cahill *et al.* is that they are computationally expensive as they generate a lot of possible sentences and then do the filtering with the help of the ranking mechanism. To overcome this drawback, Belz and Anja, introduce the Probabilistic Context-free Representationally Underspecified (pCRU) which uses probabilistic choice to inform generation instead of going through all the choices and then selecting a phrase [16].

The approaches described above all use a set of hand-crafted rules as the base generation and only use statistical methods for filtering the output generation. The alternative approach would be to use statistical on the base-generation systems and there have been approaches where grammatical rules have been derived from treebanks [60]. Hockenmaier and Steedman presented a method to extract dependencies and combinatory categorical grammar(CCG) from the Penn Treebank which consists of one million word sub-corpus from the Wall street journal. The algorithm method presented had four main functionalities: 1.) determining the constituent types of heads, complements, and adjuncts, 2.) binarizing the tree, 3.) assigning categories, and 4.) assigning the dependency structure [70]. The grammar and dependencies extracted by Hockenmaier and Steedman were used to do a more precise analysis of punctuation and also to help the parsers to arrive at the correct parse [221]. In the next subsection, we would be covering the deep neural networks and the recent surge in interest in these architectures towards solving natural language processing problems.

2.3 Deep Learning Approaches to Text Generation

Applying deep neural networks to Natural Language Processing has helped achieve state-of-the-art performance across different tasks, including the task of language generation due to the capability of neural networks to learn representations with different levels of abstraction [106, 65]. The simplest and most widely used type of neural network is the feed-forward neural network or multilayer perceptron [174] in which the data flow is in one direction and feed-forward neural networks are acyclic graph structures. Bengio *et al.* demonstrated the ability to feed forward neural networks on language modeling tasks [19].

Another type of neural network architecture that is more suited for dealing with sequential data $x^{(1)}, ..., x^{(n)}$ is the Recurrent Neural Networks (RNN) architecture. RNN has the capability to handle long sequences using the knowledge gained (**"memory"**) from previous sequence computations, unlike networks without sequence-based specialization. Application of memory to neural networks was demonstrated as early as 1982 through the Hopfield Network that was used to store and retrieve memory from a pre-trained set of patterns or memories, similar to a human brain. The network relied on neurons each producing a value of +1 or -1 depending on the input from the previous layer [75].

Hopfield's network was the inspiration behind Jordan's network [80], represented in Figure 4A, for doing supervised learning on sequences with the help of a single hidden layer and special units which receive input from the output unit which then forwards the values to the hidden nodes [121]. Elman simplified Jordan's architecture represented in Figure 4B, by adding a context unit with each hidden unit receiving its input from the units at the previous time step with a fixed weight of 1 and Elman showed that the network can learn dependencies by training the network on the sequence of 3000 bits. The model achieved an accuracy rate of 66.7% on predicting the third bit in the sequence [52, 121]. The Elman architecture played a substantial role in the discovering of the long short-term memory networks (LSTM) [69] which helped in tackling a very important problem of vanishing and exploding gradients caused by

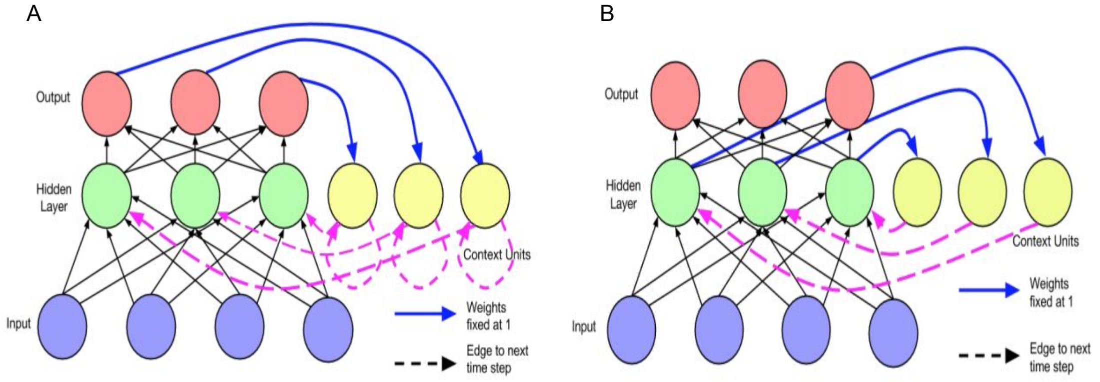


Figure 4: A. Represents the Jordan Architecture. B. Represents the Elman Architecture [121]

backpropagation while training the neural networks [183]. During backpropagation, the neural network weights receive an update proportional to the gradient of the error function. These gradients are multiplied across layers and sometimes the gradients become too small or vanish and in certain cases, the gradients grow exponentially and explode.

In the next four subsections, we list the different approaches such as language modeling, encoder-decoder, memory networks, and transformer models-based approaches for the task of language generation.

2.3.1 Language Modelling

Language models are probabilistic models that are capable of predicting the next word given the preceding words in a sequence and are widely used in the generative modeling tasks in the field of NLP. The ability of language models to model sequential data of fixed length context using feed-forward neural networks was first demonstrated in the work done by Bengio *et al.* [19]. However, the usage of fixed length context was a major drawback in the approach that was overcome in the seminal work done by Mikolov *et al.* [144] demonstrating the efficiency of RNN based language models. Similarly, another seminal work in the area of language models is the work done by Sutskever *et al.* [200] demonstrating the effectiveness of LSTM in predicting the next character of a sequence. Conditional language models are also used as a variant of language models where the language models are conditioned on a different variable apart from the preceding words like the work done by generating product reviews based on sentiment, author, item or category[122] or generating text with emotional context [63].

2.3.2 Encoder-Decoder

Another important architecture that enhanced the task of language generation was the usage of two RNNs in an end-to-end model (Figure 5) [32] that overcame a significant limitation where the neural networks can only be applied to problems where input and target can be encoded with fixed dimensionality. The **encoder** converts the input sequence into a fixed vector representation ***"c"*** by Equation 1 where "h_t" refers to hidden state at time step t, "f" represents any non-linear function and "x" represents the input sequence. The **decoder** tries to predict sequence of symbols with the help of the context vector "c". The hidden state of the **decoder** depends on the context vector "c" and is represented by Equation 2 and next symbol to be predicted is based on a condition probability 3 where g is a softmax function.

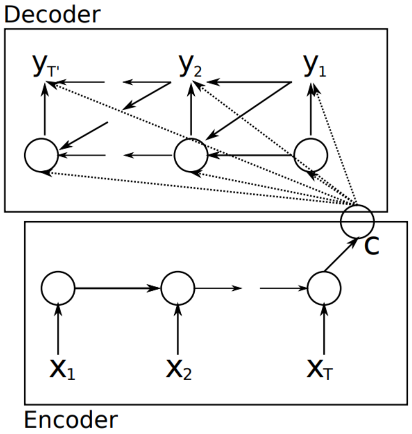


Figure 5: Encoder-Decoder architecture proposed by Cho *et al.*[32].

$$h_{(t)} = f(h_{(t-1)}, x_t) \tag{1}$$

$$s_i = f(s_{i-1}, y_{i-1}, c) \tag{2}$$

$$P(y_t|y_{t-1}, y_{t-2}, ..., y_1, c) = g(s_{(t)}, y_{t-1}, c) \tag{3}$$

Along similar lines to the work done by Cho *et al.* [32], ***seq2seq*** was introduced by Sutskever *et al.* [201] which uses two LSTMs, one to map the input sequence to a fixed vector and the other RNN to decode the fixed vector into a sequence of target symbols of varying lengths. A key difference between the work done by Cho *et al.*[32] and Sutskever *et al.*[201] was discovering that reversing the order of the input sequence improves the performance of the model and this also helps with creating short term dependencies between input and target sequence. Bahdanau *et al.* [7] identified the bottleneck caused by encoding the entire sequence into a fixed vector in the simple encoder-decoder architecture and proposed a modification that allows the decoder to "attend" to different parts of the source sentence that are relevant for predicting the next word/character of the sequence. In the attention mechanism, the context vector "c_i" is calculated as the weighted combination of all the encoder hidden states (see Eq.19), and "α" refers to how much importance should be given to respective input states.

$$
\begin{aligned}
c_i &= \sum_{j=1}^{T_x} \alpha_{ij} h_j \\
\alpha_{ij} &= \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})} \\
e_{ij} &= a(s_{i-1}, h_j)
\end{aligned}
\tag{4}
$$

A majority of the work done for the task of language generation was done using the encoder-decoder architecture. Zhang and Lapata [238] proposed a model for Chinese poetry generation with the help of RNN and the process of content determination and realization was jointly combined as one by the generator. Another example would be the NLG system developed by Wen *et al.* [218] who modified the architecture of the LSTM to constrain it semantically and be able to predict the next utterance in a dialogue context. The architecture of the modified cell the traditional LSTM cell is used for surface realization and the dialogue act cell which acts similar to the memory cell is used for the sentence planning phase. On similar lines to the work done by Wen *et al.* for language generation, Goyal *et al.* [67] presented a character level RNN for dialogue generation and addressed the issue of delexicalization and sparsity issues. Mei *et al.* [141] used the encoder-decoder architecture proposed by Cho *et al.* [32] to perform the task of content selection and realization on a set of weather database event records as a joint task by using an encoder-aligner-decoder model. The aligner is based on the attention mechanism[228, 7]. The encoder-decoder architecture was also used to generate emotional text as demonstrated by the work done by Asghar *et al*[3], Zhou *et al.*[241] and Ke *et al.* [83] .

2.3.3 Memory Networks

Memory networks, a type of learning model, were introduced by Weston *et al.* [219] to overcome to short memory encoded in the hidden states. These networks were used for a variety of question-answering tasks where the answer is produced from a set of facts into the model. The answer produced by the model can be a one-word answer or paragraph of text. The memory networks introduced by Weston *et al.* [219] had four major components input feature map, generalization, output feature map, and a response. Kumar *et al.* [98] introduced a different type of memory networks based on episodic memory and were able to solve a wider range of question-answering tasks and also on questions related to the part of speech and sentiment analysis. The work done by Kumar *et al.* [98] was extended for visual question answering by Xiong *et al.* [227]. Other works on visual question answering included the work done on using hierarchical attention on question-image pairs [128], using relational networks for visual question answering[181], using facts for visual question answering [214].

2.3.4 Transformer Models

The Transformer models (Figure 6) introduced Vaswani *et al.* [208] has helped achieve improvement over a wide range of NLP tasks. The transformer models are purely based on attention mechanisms that draw global dependencies between the input and output. The transformer is made of the encoder-decoder architecture but each encoder is a stack of six encoders with each encoder containing a self-attention and pointwise fully connected feed-forward neural networks. The decoder is also a stack of six decoders with each decoder containing the same components as the encoder,

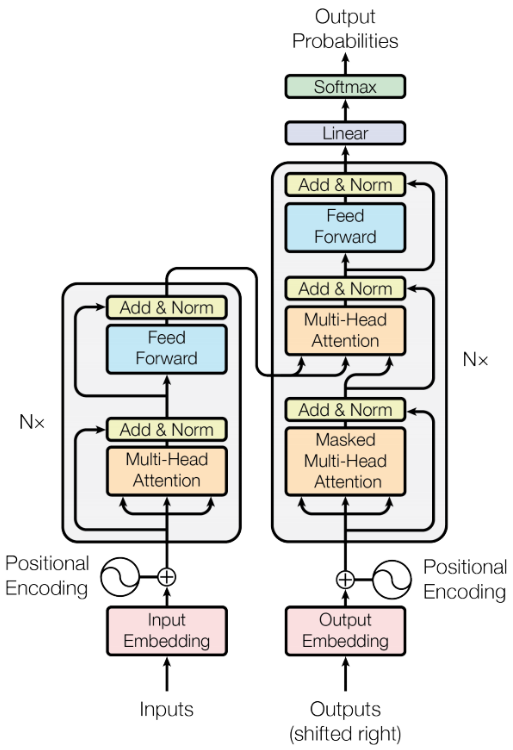


Figure 6: Transformer architecture as represented in Vaswani *et al.* [208]

but with an additional attention layer that helps the decoder focus on relevant parts of the input sentence. The work done in the space of the transformer models is still in its infancy. Some of the other works along this line include the work done by Radford *et al.* [163] and Devlin *et al.* [39] that achieve impressive results along with several NLP tasks such as Natural language inference, question-answering. Radford *et al.* [164] modified the base transformer architecture by (i) shifting the layer normalization as input to each sub-block; (ii) including a layer normalization at the final self-attention block. This work improved the existing state across a wide range of tasks such as language modeling, children's book test, reading comprehension, machine translation, question answering, modeling long-range dependencies (LAMBADA), Winograd Schema challenge, summarization.

2.4 Dialogue Systems

Dialogue systems or conversational agents (CA) are designed to generate meaningful and coherent responses that are easy to respond to, informative, and coherent when engaged in a conversation with humans. A good dialogue model incorporated in conversational agents should be able to generate dialogues with high similarity to how humans converse [115]. Conversational agents are of great importance to a large variety of applications and can be grouped under two major categories, namely, (1) Closed Domain goal-oriented systems that help users with a particular goal, (2) Open Domain Conversational agents engage in a conversation with a human and are also referred to as chit-chat models. Research in the area of dialogue systems has been pursued for a long period of time, starting from the mid-60s. “ELIZA” is one of the early, well-known dialogue systems [217] developed by Weizenbaum that used hand-crafted rules and pattern matching to mimic a psychotherapist. With the availability of large corpora [186] for training, researchers have shifted from traditional approaches such as template-based to statistically data-driven approaches [171] and more recently, building end-to-end systems using neural networks [211, 189] which is our primary focus of this section.

We provide a summarized version (Table 1) of the research done in this area, focusing on the key aspects of the dataset, architecture, optimization, evaluation metrics and we also identify potential research gaps and explain which gaps in existing research are key components that should be focused on in future work on dialogue systems.

Table 1: Survey of Deep Learning based Open Domain Dialogue Systems

Authors	Corpora	Architecture	Optimization	Evaluation Metrics	Research Gaps
Vinyals & Le [211]	1. Open subtitles 2. IT Help Desk	Seq2Seq	Cross entropy	Human evaluation	1. Lack of consistent persona. 2. Dull & generic responses.
Sordoni *et al.* [195]	1. Twitter Conversation Triples	Language model	MMI	1. BLEU 2. METEOR 3. Human evaluation	1. Incorporating context & word order.
Li *et al.* [109]	1. Twitter Conversation Triples 2. Open subtitles	Seq2Seq	SGD	1. BLEU 2. Distinct-1 3. Distinct-2 4. Human evaluation	1. Lack of consistent persona.
Shang *et al.* [189]	1. Weibo Conversation	SeqSseq + attention	N/A	Human evaluation	1. Lack of context.

Yao *et al.* [230]	1. Helpdesk chat service	Seq2Seq + Attention & Intention Network	N/A	1. Perplexity	1. Lack of context. 2. Dull & generic responses.
Serban *et al.* [187]	1. Movie Triples	Hierarchical Encoder Decoder	Adam	1. Perplexity	1. Equal importance to contexts
Li *et al.* [112]	1. Twitter Persona Dataset 2. Twitter Conversation Dataset 3. TV series transcripts	Seq2Seq + Persona Embeddings	MMI	1. Perplexity 2. BLEU 3. Human Evaluation	1. Limited persona information.
Luan *et al.* [129]	1. Ubuntu Dialogues	LSTM Language Model Learning with LDA	SGD	1. Perplexity 2. Response Ranking	1. Lack of consistent persona. 2. Dull & generic responses.

Li *et al.* [113]	1. Open Subtitles	Seq2Seq + RL	MMI + Policy Gradient	1. BLEU 2. Dialogue Length 3. Diversity 4. Human Evaluation	1. Lack of consistent persona. 2. Rewards may lead to suboptimal generation.
Dušek *et al.* [47]	1. Public Transport Information	Seq2Seq + Attention & Context Encoder	Cross Entropy	1. BLEU 2. NIST 3. Human Evaluation	1. Lack of incorporating context.
Mou *et al.* [149]	1. Baidu Teiba forum	Seq2Seq + PMI	SGD	1. Human Evaluation 2. Length 3. Entropy	1. Lack of incorporating context.
Serban *et al.* [188]	1. Twitter Dialogue 2. Ubuntu Dialogue	Latent Variable Hierarchical Encoder Decoder	Adam	1. Human Evaluation 2. Length 3. Entropy	1. Equal importance to contexts.

Mei *et al.* [142]	1. Movie Triples 2. Ubuntu Dialogue	Language Models + Attention + LDA Reranking	Adam	1. Human Evaluation i. Grammar & Fluency ii. Logic Consistency iii. Semantic Relevance iv. Scenario Dependence v. Generality	1. Lack of consistent persona.
Xing *et al.* [225]	1. Baidu Teiba forum	Seq2Seq + LDA + Joint Attention	Ada Delta	1. Perplexity 2. Distinct-1 3. Distinct-2 4. Human Evaluation	1. Lack of consistent persona
Cao & Clark [27]	1. Open Subtitles	Variational autoencoder	MMI	1. Human Evaluation	1. Lack of context. 2. Lack of consistent persona.

Ghazvininejad *et al.*[62]	1. Four Square 2. Twitter Conversation	Seq2Seq + World Facts + Contextual Facts	Adam	1. Perplexity 2. BLEU 3. Diversity 4. Human Evaluation	1. Equal importance to contexts. 2. Facts may not be available for all contexts.
Young *et al.* [232]	1. Twitter Conversation	Tri-LSTM Encoder	SGD	1. Recallk	1. Lack of consistent persona. 2. Fixed Concept net.
Asghar *et al.* [5]	1. Cornell Movie Dialog	Seq2Seq + Online Active Learning	Cross Entropy	1. Human Evaluation 2. Syntactic coherence 3. Relevance 4. Interestingness 5. Relevance	1. Manual user feedback during training.

Lewis *et al.* [107]	1. Negotiation dataset	Seq2Seq + self play + RL	SGD	1. Score 2. Agreement 3. Pareto Optimality 4. Perplexity	1. Lack of consistent persona in negotiation.
Li *et al.* [115]	1. Open Subtitles	GAN	N/A	1. Human Evaluation 2. Adversarial Evaluation	1. Reward Sparsity. 2. Mode Collapse and word order.

Qian *et al.* [162]	1. Weibo Dataset 2. Profile Binary Subset 3. Profile Related subset 4. Manual Dataset	Encoder Decoder + Profile Detector	SGD	1. Human Evaluation i. Naturalness ii. Logic iii. Correctness iii. Semantic Relevance iv. Consistency v. Variety 2. Profile Detection 3. Position Detection	1. Limited profile information as persona.
Qiu *et al.* [162]	1. Chat log of online customer service	Attentive Seq2Seq + IR + Rerank	N/A	1. Human Evaluation 2. Precision 3. Recall 4. F1	1. Lack of consistent persona.

Serban *et al.* [185]	1. Ubuntu Dialogue 2. Twitter Dialogue	MrRNN	Adam	1. Human Evaluation	1. Lack of incorporating context. 2. Lack of consistent persona.
Shen *et al.* [191]	1. Ubuntu Dialogue	Hierarchical Encoder Decoder + context	KL Divergence	1. Embedding Evaluation i. Greedy ii. Average iii. Extrema 2. Human Evaluation	1. Lack of consistent persona. 2. Hand crafted rules.
Tian *et al.* [204]	1. Baidu Tieba	Hierarchical Encoder Decoder	Ada Delta	1. BLEU 2. Length 3. Entropy 4. Diversity	1. Order of sequence not taken into consideration during weight calculation.
Xing *et al.* [226]	1. Dataset from Douban Group.	Hierarchical Recurrent Attention Network	N/A	1. Perplexity 2. Human annotation	1. Lack of consistent persona.

Zhou *et al.* [241]	1. NLPCC dataset 2. STC dataset 3. Weibo Emotion dataset	Encoder Decoder + External Memory + Internal Memory + Emotion Embedding	Cross Entropy	1. Perplexity 2. Accuracy 3.Human Evaluation	1. Need extrinsic input of desired emotion.
Ghosh *et al.*[162]	1. Fisher English Training Speech Parts 2. Distress Assessment Interview Corpus 3. SEMAINE dataset 4. CMU MOSI	Language Model	N/A	1. Perplexity 2. Human Evaluation	1. Depends heavily on external linguistic information.

Asghar *et al.* [4]	1. Cornell Movie Dialogs	Seq2Seq + Affective Embeddings	1. Cross Entropy 2. Min Affective Dissonance 3. Max Affective Dissonance 4. Max Affective Content	1. Human Evaluation i. Syntactic Coherence ii. Natural iii. Emotion Appropriateness	1. Lack of consistent persona.
Zhang *et al.* [234]	1. STC dataset	Specificity Controlled Seq2Seq	Adam	1. BLEU-1 2. BLEU-2 3. Distinct-1 4.Distinct-2 5. Average embedding 6.Extrema embedding	1. Lack of incorporating context. 2. Lack of consistent persona.

Zhang *et al.* [235]	1. PERSONA-CHAT dataset	1. Baseline Ranking Models 2. Ranking Profile Memory Network 3.Key-Value Memory Network 4.Seq2Seq 5.Generative Profile Memory Network	N/A	1. Perplexity 2. Hits@1 3.Human Evaluation i.Fluency ii. Engagingness iii. Consistency iv. Persona Detection	1. Artificial persona assignment.
Mazare *et al.* [137]	1. Reddit dataset	Transformer + Persona + Context + Response Encoder	Adamax	1. hits@1	1. Limited persona information.

Rashkin *et al.* [166]	1. Empathetic Dialogue	Transformer Model	Adamax	1. Perplexity 2. Avg. BLEU 3. P@1,100 4. Human Evaluation i. Empathy ii. Relevance iii. Fluency	1. Lack of usage of persona information available in dataset.
Huang *et al.* [78]	1. Open Subtitles 2. CBET	Seq2Seq	Adam	1. Accuracy	1. Lack of longer context to understand emotions generated.
Kottur *et al.* [95]	1. Movies-DiC dataset 2. TV-Series 3. Open Subtitles dataset	Context-aware Persona based Hierarchical Encoder Decoder	Adam	1. Perplexity 2. Recall@1 3.Recall@5	1. Needs a lot of data per speaker.

Niu & Bansal [151]	1. Stanford Politeness Corpus 2. Stack Exchange	1. Seq2Seq 2. Fusion model (Seq2Seq + polite-LM) 3. Label fine tune Model 4. Polite-RL Model	Adam	1. Perplexity 2. Perplexity @L 3.Word Error Rate 4.Word Error Rate@L 5. BLEU-4 6. Human Evaluation i. Politeness ii. Quality	1. Lack of longer context.
Chen *et al.* [29]	1. Ubuntu Dialogue 2. Douban Conversation 3. JD customer service	Hierarchical Variational Memory Network	Adam	1. Avg Embedding 2. Greedy Embedding 3. Extrema Embedding 4. Human Evaluation i. Appropriateness ii. Informativeness	1. Lack of consistent persona.

Bhatia *et al.* [20]	Yik Yak dataset	1. Seq2Seq + Locations 2. Seq2Seq + User model	N/A	1. Perplexity 2.ROUGE	1. Lack of encoding longer context.
Dinan *et al.* [43]	1. Wizards of Wikipedia	1. Retrieval Transformer Memory Network 2.Generative Transformer Memory Network	NLL	1. Recall@1 2. Perplexity 3.Human Evaluation	1. Lack of consistent persona.
Wolf *et al.* [223]	1. PERSONA-chat dataset	Transformer Model	Adam	1. Perplexity 2. Hits@1 3. F1 metrics	1. Limited persona information.
Zheng *et al.* [240]	1. PERSONALDIALOG dataset	Seq2Seq + Personality Fusion	Adam	1. Perplexity 2.Distinct-1 3. Distinct-2 4. Accuracy 5. Human Evaluation	1. Limited persona information. 2. Explicit persona.

We identify three main issues found in prior works done in the area:

1. **Dull and generic responses** - One problem with building end-to-end conversational agents based on vanilla seq2seq is that they are prone to generating dull

and generic responses such as “I don’t know”, “I am not sure.” etc. [211, 109]. These trivial responses make the conversational agents unable to hold long conversations with a human. Li *et al.*[109] suggested a mechanism to overcome this issue with an optimization function (see equation 5, T is the target and S is the source) as the authors only considered the likelihood of the responses when given input and proposed using Maximum Mutual Information (MMI) as the optimization objective function (see equation 6) where λ is a hyperparameter to penalize generic responses.

$$\hat{T} = \arg\max_{T}\{logp(T|S)\} \tag{5}$$

$$\hat{T} = \arg\max_{T}\{logp(T|S) - \lambda logp(T)\} \tag{6}$$

Recent approaches to the task of conversational modeling have all tackled this issue through the use of the previous utterance as contextual information or with the help of an attention mechanism that helps of a particular part of the input utterance or using reinforcement learning that penalizes the agent when it produces trivial responses or repetitive utterance, [113, 115, 125].

2. **Personality** - Endowing conversational agents with a coherent persona is a key to building an engaging and convincing conversational agent [151]. The concept of personality has been well studied in psychology. Traditionally, research on using personality traits has been based on the standard ***Big Five*** model (extraversion, neuroticism, agreeableness, conscientiousness, and openness to

experience) and some of the early works on building personalized dialogue systems have been based on this [131].

However, identifying personality traits through ***Big Five*** model is difficult and expensive to obtain [240, 235]. Alternative approaches to take advantage of the psycholinguistic findings are still in their infancy. Some of the proposed approaches to solve this problem have been through explicit or implicit modeling of personality [240]. Explicit modeling involves creating profiles of users with features such as age, gender [240] or assigning artificial persona to users and asking them to interact about it [235]. Implicit Modeling of persona involves creating vectors about the users based on similar features such as age, gender and other personal information [112, 95].

3. **Encoding Context** - Encoding contextual information such as world facts, knowledge base or previous turns of the conversation are important issues to ensure that the conversational agent has enough information to produce a coherent, informative, and novel response that is in tune with the context of the conversation. A lot of prior research in this field demonstrated the working of the system through a one-to-one mapping between an input utterance and the generated response. This makes it hard to judge the quality of the response generated or the performance of the model with regards to the context of the conversation or how the model performs when it comes to multi-turn conversations.

   To overcome this issue, researchers have focused on including the previous turn of the conversation as a contextual piece of information to the model and this mechanism has been done in two different ways. Prior research has used both sequential [195] and hierarchical models [187] as the mechanism of encoding contexts into the conversation. In the sequential encoding of the context, the previous turn of the conversation is concatenated to the current input utterance. In the hierarchical encoding of the context, a two-step approach is followed by performing an utterance level encoding and then followed by an inter-utterance encoding. Tian *et al* conducted an empirical study that evaluates the advantages and disadvantages of sequential and hierarchical and show evidence that hierarchical model outperforms sequential models when encoding contextual information [204].

The transformer model by Vaswani *et al.*[208] is another mechanism that provides a way of encoding context through the self-attention mechanism incorporated in the model. The transformer architectures have demonstrated the capabilities to outperform other architectures for the task of machine translation and parsing. More recently, the transformer models have been used for conversational agents [223, 43, 166]. Wolf *et al.* [223] demonstrate the usage of the transformer model for personalized response generation on the PERSONA-CHAT dataset where the model concatenates each artificial persona provided along with the utterances of the conversation. Encoding factual knowledge to augment the model was demonstrated by Dinan *et al.* [43] and Young *et al.*[232] using the transformer models and Tri-LSTM encoder approach respectively.

2.5 Cognitive Architectures

Cognitive architectures provide a blueprint for intelligence. These architectures identify the structures and processes in the brain and facilitate understanding the interactions between them [150, 199]. More concretely, while building intelligent artificial agents, cognitive architectures help to understand how perception, vision, action selection along with the ability to store knowledge using memories (short-term and long-term) make agents function with human-level intelligence [199, 105, 94]. Cognitive Architectures are also able to simulate human's cognitive and behavioral characteristics such that these architectures can be run in both virtual and physical intelligent agents[231]. Prior research in the area of cognitive architecture can be grouped into three main categories of ***Symbolic, Emergent, and Hybrid*** as represented in Figure 7.

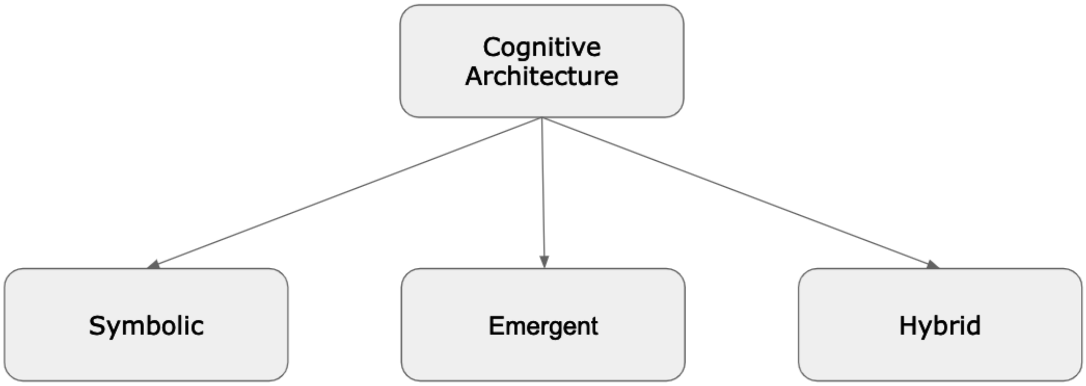


Figure 7: Types of Cognitive Architecture

Symbolic architectures also known as cognitivist architectures, use symbols to represent concepts and these symbols can be manipulated using if-then rules. These architectures maintain a consistent knowledge base for symbols and formal logic is used reasoning about the facts known about the world [94, 231]. SOAR [99, 101] is a classic symbolic architecture, developed by John Laird, Allen Newell, and Paul Rosenbloom and has been primarily used in the area of robotics.

Emergent Architectures differ from symbolic architectures by building parallel models, similar to neural networks, and represent the process of human cognition from a bottom-up approach. These architectures are easy to design and must be trained carefully to achieve optimal behavior, a property that is shared by neural networks. A major difference between symbolic architecture and emergent architecture is the lack of knowledge base and this lack of knowledge may inhibit emergent architectures from picking up new behaviors.

Hybrid Architectures achieve a sort of middle ground between symbolic and emergent architectures and there are no restrictions in how the hybridization is done [94]. Hybrid architectures adopt hierarchical structure, just like emergent architectures, and can also perform symbolic processing. In our work, we focus on the standard model of cognition (Explained in Section 3.3), a hybrid approach-based cognitive architecture that forms a consensus from three cognitive architectures, namely SOAR, ACT-R, and Sigma [100].

Cognitive Architectures are designed to act as the blueprint for intelligent agents that can perform a multitude of different tasks such as ***Perception, Attention, Action Selection, Memory, Learning, Reasoning*** [94]. We focus on **action selection** and **memory** functionalities from established cognitive architectures. The action selection mechanism in cognitive architecture determines the type of action to take for a given situation and is involved in the decision-making process. Action selection can be performed in two ways: (i) Planning based; (ii) dynamic action selection. Planning-based action selection is commonly used in symbolic cognitive architectures where is dynamic action selection is designed to simulate the behavior of humans. We focus on performing action selection mechanisms using the Attention process. The attention process helps focus on different parts of the sensory inputs in cognitive architectures which are adopted to perform action selection on dialogue history in our work.

Memory plays a significant role in cognitive architectures and every cognitive architecture has some type of memory to store the results. There are four types of memory within cognitive architectures such as sensory memory, working memory, long-term memory, and global memory. Sensory memory acts as a pipeline that transfers inputs to the other memory structures [94]. We focus our attention on Working Memory and Long-Term Memory (refer to Section 3.3). Working memory is required for the action selection to work optimally. Long-Term Memory stores factual knowledge and other knowledge for a long period of time and can be divided into procedural and declarative which is further divided into semantic and episodic memory. Procedural memory stores routine behaviors and declarative memory stores the knowledge. Within declarative memory, semantic memory is known for storing the facts and episodic memory is known for storing past experience.

CHAPTER 3: COGNITIVE ARCHITECTURE BASED DIALOGUE SYSTEM

3.1 Introduction

Natural language generation entails not only incorporating fundamental aspects of *artificial intelligence* but also *cognitive science* [169, 199]. Extant approaches to natural language generation have typically been formulated as sequence-to-sequence (*seq2seq*) frameworks, an adaptation of machine translation systems [211, 196, 112, 110, 187, 189, 201]. Dialogue systems built using the *seq2seq* framework can be used to generate interesting, coherent dialogue [114]. However, it has been shown that engaging with these systems for longer interactions could result in dull and generic responses in open-domain situations due to their reliance on a shorter context of the conversation - namely, the last utterance in the dialogue history [211, 110, 4]. To make better use of context, researchers have used both hierarchical and non-hierarchical models [204], but these models still suffer from sub-optimal performance due to the inclusion of entire conversation history that may contain irrelevant utterances [215].

The challenge thus becomes: ***How do we encode longer context into the natural language generation system such that the algorithm can focus on the salient information in the conversation (e.g. topics, entities) while appropriately discounting parts of the conversation that may primarily serve to preserve social conventions (e.g. utterances such as "ah", "ok"***

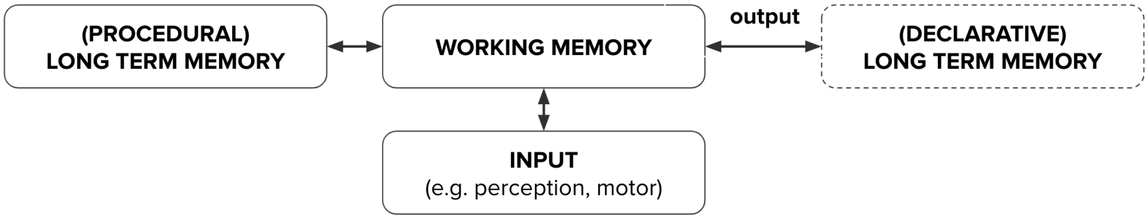


Figure 8: Standard Model of Cognitive Architecture containing two forms of long-term memory (Procedural and Declarative) and Working Memory to address given input.

etc.)?

To address this challenge, we take a cognitive science approach. Our approach relies on an adaptation of the Standard Model (8) [152, 100], an established model of memory in cognitive science. This model provides the framework with which to conceptually and practically address both long-term memory and short-term memory (also known as working memory), along with an action-selection mechanism acting as a bridge between them. According to this model of human cognition, given an input (for example, through perception), an output is generated by taking into account elements stored in the working memory as well as long-term storage (explained in detail in Section 2). Our system architecture closely mirrors the Standard Model, as described in Section 4.

The concept of memory, conceptualized as encoding contextual information in dialogue, has been explored in prior literature in question-answering systems [198, 98, 227]. For instance, Weston *et al.* [220] introduced a new class of learning models called the memory networks for question answering systems. However, the usage of memory networks for dialogue generation is still in its infancy. To the best of our knowledge, the model proposed in this article is the first to use the Standard Model of Cognition to more closely tie the natural language generation system to the way human cognition works.

In addition to building dialogue generation systems that are cognitively inspired, there is also a pressing need for more meaningful metrics to evaluate their output [123]. We present a preliminary step in this direction, with the adoption of a new metric - *the Gunning-Fog index* - that can be used to automatically evaluate the readability of generated output as a part of the conversation.

We make the following contributions as a part of this work:

1. A novel **cognitively-inspired natural language generation model** that accounts for larger contexts through *long-term* and *working memory.*

2. Novel method of identifying and computing **saliency** of the context in long-term memory.

3. Introduction of additional metrics to evaluate the readability of dialogue output, including the **Gunning-Fog index**.

3.2 Motivation

Consider the example shown in Table 2 between two interlocutors A and B. Given the query utterance, the task is to generate a response to this utterance. We note that in the query utterance, interlocutor B mentions *it* in the first sentence - presumably referring to the *novel* mentioned in the prior context. We also note the presence of interesting entities further in history (*windmill* in the first utterance by A). Some utterances do not add much salient information for response generation (socio-behavioral information notwithstanding). For example, *got it* and *so did i* by B

Table 2: Example of a conversation between A and B with the dialogue history of 7 utterances and the query utterance.

A: you know how to get back to the windmill, right?
B: got it.
A: i had a good time tonight. i really did
B: so did i.
A: see you around
B: um. did you still want to read my novel?
A: oh, yeah. sure. of course.
Query Utterance
B: hope you like it. free to stop reading at any time. i will take no offense.

are such instances. Our primary objective thus is to build a generative model that produces an appropriate response by identifying the salient contextual utterances from conversational history while appropriately discounting the non-informative (for response generation) utterances such as *got it*.

The problem can be divided into two parts:

1. **Action Selection** - Given a set of utterances as part of conversation history (U_t), the primary goal for the Action Selection mechanism is to capture the meaning of conversation history and weight the importance of each utterance in history with regards to the query utterance (Q) and inform the working memory (C_n) (Eq. 7).

$$U_{t\in 1\leq t\leq 8} \xrightarrow{ActionSelection} C_{n\in 1\leq n\leq 3}. \tag{7}$$

Our formulation stems from the observation that encoding the entire dialogue history as the context for the conversation might lead to sub-optimal performance when there is noise present in the dialogue history.

2. **Response Generation** - Our second task is to generate a response $Y = y_1, y_2, ..., y_m$ that is coherent with the context of the conversation. We create a context vector (V_{enc}) as sum of all hidden states weighted by their respective importance scores. The response generator predicts the next word y_t (Eq. 8):

$$P(y_t|y_{t-1}, y_{t-2}, ..., y_1, v_{enc}) = g(y_{t-1}, v_{enc}) \tag{8}$$

   where g is a nonlinear activation function.

3.3 Standard Model of Cognition

The standard model proposed by Laird *et al*[100] was to provide a consensus model that can be used for research and application and can also act as a coherent baseline that facilitates progress. The standard model of cognition encapsulates structures and processes found in cognitive architectures such as ***ACT-R, Sigma, Soar***.

The standard model of cognition combines different components that perform different tasks. Some of the core components of the model include perception and motor, working memory, declarative long-term memory, and procedural long-term memory. In our work, we focus on the ***working memory, procedural long-term memory, and declarative long-term memory***.

The working memory, also known as short-term memory provides a temporary space that can store the information needed to solve a particular task. Working memory can retrieve information from long-term memory through an action-selection mechanism. We use ***Attention mechanism***, a form of ***action selection*** that helps in retrieval of relevant pieces of information within cognitive architectures [8].

The declarative memory contains episodic memory [105], a form of which was introduced as a component of dynamic memory networks for question answering systems [98]. The procedural memory is responsible for storing information that helps with everyday activities [33] and has knowledge about internal or external actions.

3.4 Cognitive Memory Architecture Model

In this subsection, we introduce our end-to-end model - Cognitive Memory Architecture (CMA) - a memory augmented encoder-decoder model inspired by the work done in the field of cognitive architectures. As illustrated in Figure 20, the model comprises of the following components:

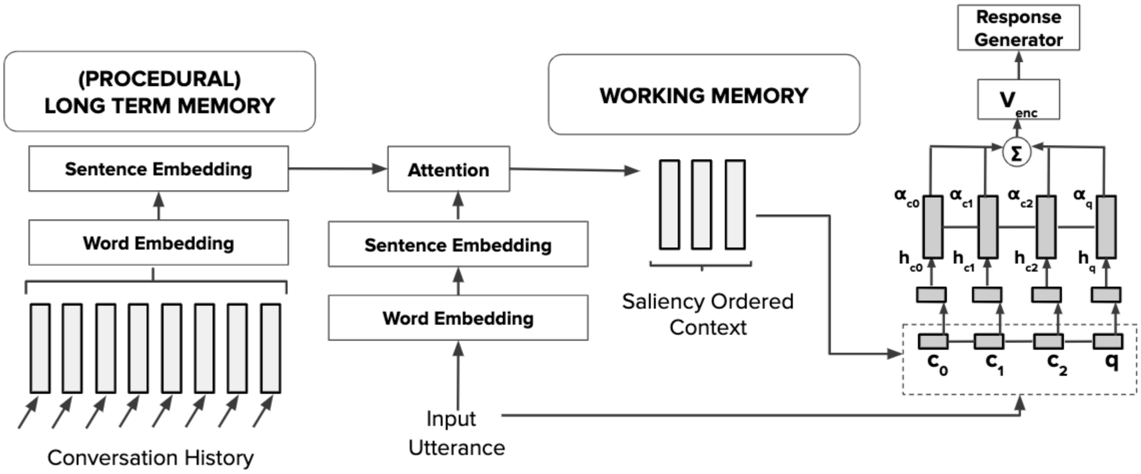


Figure 9: Architecture of CMA model with two memory components namely, long-term and working memory that hierarchically augments the input utterance.

1. **Long Term Memory** - The Long Term memory stores the history of the conversation. The input to the long term memory is the historical sequence of utterances U_t where the sequence $t \in 1 \leq t \leq 8$. We make a simplifying assumption that only a maximum of 8 utterances is stored in long-term memory (we shall discuss this assumption and its implications in the Conclusion and Discussion

section). Each utterance in history is represented by a set of words ($w_1....w_n$) and each word is represented by its respective word embedding. We adopt an approach similar to Tian *et al.* [204] and convert each utterance in the history into its respective sentence embedding as the sum of the word embeddings (Eq. 9). Similarly, we also convert the input utterance (Q) into its sentence embedding (Eq. 10). Next, we compute the cosine similarity between the sentence embeddings of each of the utterances in the dialogue history e_{u_i} and the sentence embedding of input utterance e_q (Eq. 11):

$$e_{u_i} = \sum_{w \in u_i} e_w, \tag{9}$$

$$e_q = \sum_{w \in q} e_w, \tag{10}$$

$$s_{u_i} = sim(u_i, q) = \frac{e_{u_i} \cdot e_q}{||e_{u_i}|| \cdot ||e_q||}. \tag{11}$$

We improve Eq. 11 by introducing two additional parameters λ and τ that incorporate additional constraints to consider the order of utterance in the history of the utterances. λ represents the order of the sequence (in the range of $0.1 \leq \lambda \leq 0.8$ in our work) and τ is the ratio between the number of words in the utterance to the maximum length of the target utterance. Our intuition behind defining these additional parameters in Eq. 12 is that (a) the most important utterances to focus on when generating a response may be present in the early part of dialogue history and (b) these utterances should be given relative importance that is meaningfully encoded when generating the response. Accordingly, we reformulate Equation 11 as follows (Eq. 12):

$$s_{u_i} = sim(u_i, q) + \lambda \cdot \tau. \tag{12}$$

In order to attain the importance scores of each utterance in the history relative to the query utterance, we perform the **Action Selection** using the attention mechanism proposed by Bahdanau *et al.* [8]:

$$\alpha_{u_i} = \frac{exp(s_{u_i})}{\sum_{j=0}^{t} exp(s_{u_j}) + exp(s_q)}, \tag{13}$$

$$\alpha_q = \frac{exp(s_q)}{\sum_{j=0}^{t} exp(s_{u_j}) + exp(s_q)}. \tag{14}$$

where s_q is 1 as the query utterance similarity is computed against the same vector and α_{u_i} represents the importance score of each historical utterance and α_q is the importance of the query utterance.

2. **Working Memory** - The working memory stores the utterances necessary for generating an appropriate response within the context of the conversation while appropriately down-weighting the not useful utterances. The working memory stores C_n utterances that have the highest score produced by the Action Selection Mechanism (Equation 13). In our model, the value of n depends on the number of utterances present in long-term memory; we choose a value between one and three i.e. $n \in 1 \leq n \leq 3$.

3. **Input Module** - The input module works with the utterances in the working memory C_n and query utterance Q. We use a two-step hierarchical model similar to Serban *et al.* [187]. The first step is the production of an utterance vector, which is computed as the hidden state produced after the last token of the particular utterance is processed through a *GRU* [31]. This process is followed for all the utterances in the working memory and the query utterance. The second step is the inter-utterance modeling that processes the utterance vector through the use of another *GRU* and produces a hidden vector that represents the dialogue until that utterance. The final vector V_{enc} is the sum of the hidden vectors produced in the inter-utterance modeling phase and each vector is weighted by the saliency score from the action-selection mechanism (Eq. 15):

$$V_{enc} = \sum_{i=0}^{N} \alpha_{u_i} h_{u_i} + \alpha_q h_q. \tag{15}$$

where h_{u_i} represents the hidden vector from inter-utterance modelling of utterances in working memory and h_q represents the hidden vector from inter-utterance modelling of the query utterance.

4. **Response Generator** - The response generator is responsible for generating an appropriate response Y within the context of the conversation. We use another GRU, which updates the hidden states and then generates the responses $Y = y_1, y_2, ..., y_t$. and it uses the V_{enc} produced by the Input Module.

3.5 Experiment

3.5.1 Dataset

We used the MovieTriples dataset made available upon request by Serban *et al.* [187]. The dataset contains triplets of dialogues, namely $D1, D2, D3$ between two interlocutors. In our experiment, we processed the dataset to leverage more contextual information about the conversation. We combined all the dialogues based on the dialogue ID (available in the dataset) of a particular movie. Next, we divided the resulting dataset into sequences S_t where *t*∈*3*≤*t*≤*10* and t indicates the number of utterances present in a sequence. Each sequence is further divided into dialogue history, query and target utterances. The historical sequence is represented U_t, where *t*∈*1*≤*t*≤*8*. The input and target sequences are represented by Q_t and A_t respectively. We limited the maximum length of an utterance in a sequence to *20*. The basic pre-processing of the text in the dataset was carried out by Serban *et al.* [187]. The statistics of our dataset after pre-processing are provided in Table 3.

Table 3: Descriptive statistics of the corpus used in our experiments

	Training	Testing
Number of sequences	42738	1000
Avg. context utterances	5.5	5.24

3.5.2 Metrics

Evaluating the quality of responses generated by the model in open domain situations where the goal is not defined is an important area. Prior work in this regard includes PARADISE [212], one of the well-known metrics for evaluating spoken dialogue systems, however, this metric relies on human-generated supervised signals like task success. Researchers have also adopted methods such as BLEU [157], METEOR [9], ROUGE [120] from machine translation and text summarization [123] tasks. Metrics like BLEU and METEOR are based on word overlaps between the proposed and ground truth responses; they do not adequately account for diverse responses that are possible for a given input utterance and show little to no correlation with human judgments [123, 210]. Crowdsourced judgments have also been used as an alternative form of evaluating the quality of the responses. Researchers have used different metrics such as *ease of answering*, *coherence*, *information flow* [114], *naturalness* [4], *fluency* [235] and *engagement* [210] for human-based judgments. However, such evaluations are expensive to obtain and infeasible when a large volume of responses needs to be evaluated.

Gunning Fog index: With the need for meaningful yet automated metrics for evaluation, we propose using the Gunning Fog Index[3] as a new metric for evaluating the quality in terms of ***readability*** of the generated text. Gunning-Fog index (Equation 16) provides a score between 0 and 20 indicating the readability of the sentence.

$$GunningFogScore = 0.4 * \left[\left(\frac{words}{sentences}\right) + 100 * \left(\frac{complexwords}{words}\right)\right] \tag{16}$$

where complex words refer to words containing three or more syllables.

Li *et al.* [114] previously proposed Ease of Answering as a metric for human evaluation of generated responses. We consider the Gunning Fog index as a complementary

[3] https://en.wikipedia.org/wiki/Gunning_fog_index

Table 4: Importance of recent utterances when compared to earlier utterances present in dialogue history.

Type of dialogue history	Recent utterances	Earlier utterances
Long (n=91)	63.95%	36.05%
Medium (n=16)	64.29%	35.71%
Short (n=13)	8.33%	91.67%

measure to the Ease of Answering metric, since it can be used to evaluate if generated response as part of the conversation is easy to comprehend and respond to.

3.6 Evaluation

3.6.1 Action Selection Approach

We conducted several experiments to evaluate the efficacy of our proposed approach on the selection of appropriate contextual utterances and the downstream natural language generation output. Experiments 1 and 2 described in the following subsection are focused on the performance of the action section mechanism.

We analyzed the ability of the action selection mechanism to pick the relevant contexts from long-term memory to working memory and compare it against human judgments of relevance. We recruited 60 annotators to annotate 120 randomly sampled conversations from the test data. We asked each annotator to rank order each utterance in order of saliency from the dialogue history U_t where $t \in 1 \leq t \leq 8$. An example ranking can be found in Table 4 in the column labeled Human Ranking. We took the majority vote for each utterance to be its rank. As seen in the example in Table 4, the utterance before the query utterance was rated most salient by the human annotators, while the second utterance (*thanks*) by B was labeled least salient.

Table 5: Example demonstrating action selection, with Human Ranking compared to CMA model and Tian *et al.*

Dialogue History	Human Ranking	CMA Model	Tian *et al.*
A: hey, they would not let up, man! they keep calling you an accomplice in that burglary murder.	7	6	**5**
B: thanks	8	**8**	8
A: by the way, what are your doing with <person>?	6	**5**	3
B: she needed a lift.	5	7	7
A: oh. okay! this calls for a beer! a lot of beer!	3	4	6
B: thanks, but i have some business to take care of.	4	3	2
A: well. take care of that later. here. i will get the beer.	2	1	1
B: how'd you know this was <person>'s place?	1	2	4
Query Utterance			
A: <person>told me all about it.			

Experiment 1: What is the relative importance of context utterances when generating a response? Are more recent utterances more important? Table 4 shows the importance of recent utterances when compared to the earlier utterances in the dialogue history as rated by the human annotators. In Table 4 we refer to dialogue histories of length 6, 7, or 8 as Long, dialogue histories of length 4 or 5 as a medium, and dialogue histories of length less than 3 as Short. *Recent Utterances* refer to the two utterances immediately before the query utterance. *Earlier Utterances* refer to the utterances in dialogue history beyond the two utterances before query utterance. We observe the following from the human annotator's rankings: 1) For long and medium dialogue histories, only around 63% , the recent utterances are labeled as the most salient in capturing the context of the conversation. This means that around 36% of the time, salient information is included in the history of dialogue

beyond the immediate one or two utterances before the query utterances. 2) For short conversations, the situation is even worse. 91.67% of the time the earlier dialogue histories are referred to as most important. The salient entities or information to respond to would not be present in the one or two utterances immediately preceding the query utterances, but earlier in dialogue history (however, we note lower sample sizes for short and medium vs. long histories).

Finding 1: Our empirical results show that salient information is indeed present earlier in dialogue history, even while varying history length.

Experiment 2: How does the action selection mechanism in our proposed CMA model address the relative importance of context utterances from the history of conversation?

To evaluate the performance of the action selection mechanism, we compare our results with an existing state-of-the-art model proposed by Tian *et al.* [204], as their work is most similar work to ours which also attempts to identify and encode context importance. Table 5 shows an example conversation along with the performance of our CMA model action selection mechanism compared with Tian *et al.* and ranking given by human annotators (1=most salient, 8=least salient). Figure 10 shows the accuracy of our CMA model and Tian *et al.* model in identifying the top three salient contexts (as judged by human annotators). We compare the performance of both models when dialogue histories are Long (length 6, 7, or 8), Medium (length 4 or 5), and Short (length 3 or below). CMA model refers to saliency scores attained by the action-selection mechanism using equation Eq. 12. We demonstrate using Figure 10 the ability of our method to identify salient utterances and outperform the existing

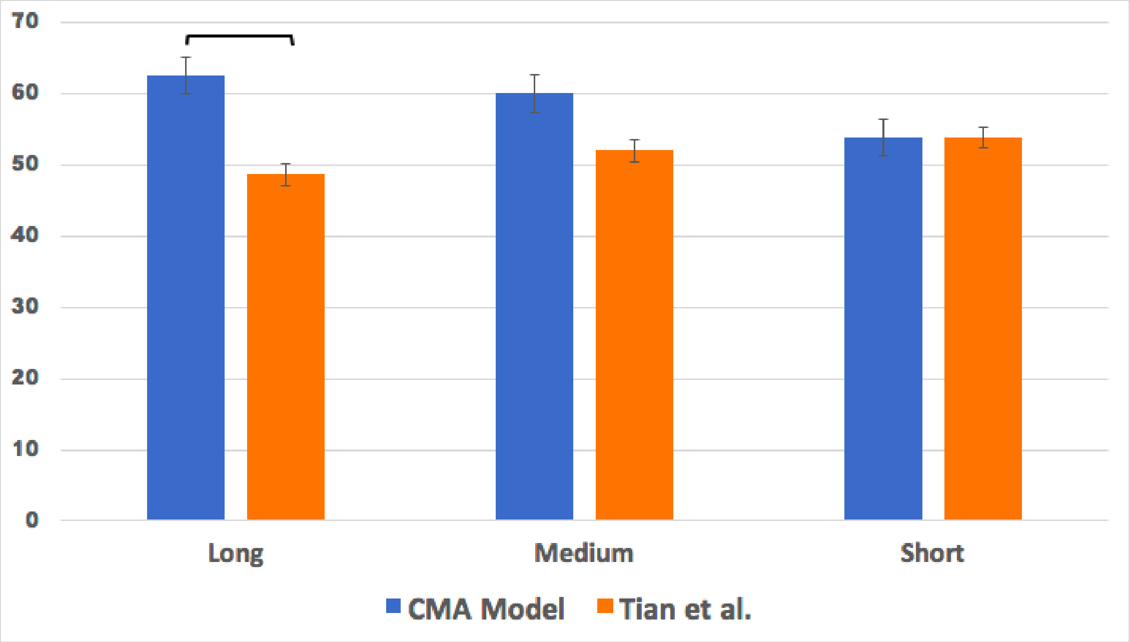


Figure 10: Accuracy of identifying salient utterances by action selection mechanism in our proposed CMA model compared to state of the art method used by Tian *et al*[204].

state-of-the-art methods in longer historical dialogue histories (statistically significant $p<0.001$ while achieving comparable performance in shorter dialogue histories.

Finding 2: CMA model can identify salient utterances in dialogue histories of varying length (compared with human judgments) and also outperform state-of-the-art models in this task.

Experiment 3: How well does the action selection mechanism in our proposed CMA model correlate with the rankings provided by humans?

We did correlation analysis to evaluate the performance of the action-selection mechanism. We compare our results with an existing state-of-art model proposed by Tian *et al.* [204]. Table 6 shows Spearman and Kendall tau correlation results of our action selection mechanism and Tian *et al.* method to the rankings provided by the

Table 6: Spearman and Kendall Tau Rank Correlation Analysis between CMA Action Selection mechanism and Tian *et al.* method to the rankings from human annotators. *** $p < 0.001$; * $p < 0.05$

		Long	**Medium**	**Short**
Spearman	CMA $\sim$ Human Ranking	**0.46*** **	**0.27***	0.13
	Tian *et al* $\sim$ Human Ranking	0.33***	0.12	**0.25**
Kendall Tau	CMA $\sim$ Human Ranking	**0.37*** **	**0.23***	0.12
	Tian *et al* $\sim$ Human Ranking	0.26***	0.10	**0.24**

human annotators. We find that CMA action selection mechanisms show a higher correlation to humans than the current state-of-the-art method. On a fine-grained analysis, we find a higher correlation on dialogue histories of **Long (length 6, 7 or 8), Medium Medium (length 4 or 5)** to ranking judgments provided by human annotators. **Finding 3:** We find that the CMA Action selection mechanism has shown a higher correlation to the humans on dialogue history histories of length ≥ 4.

3.6.2 Response Generation

We report the performance of our CMA model on the task of dialogue generation using traditional metrics such as BLEU, Diversity, and Length to be consistent with existing literature. We also report the performance on the new metric adopted in our work, the Gunning Fog index. In the tables, CMA refers to the model implemented in this paper, No Context Seq2Seq model represents the model with no context provided (only the query utterance is input to the model during the generation process) and Context Seq2Seq represents the model which takes the last utterance of the dialogue history as context (Query utterance+one prior utterance).

3.6.2.1 Dialogue Evaluation Metrics

Table 7: Performance of CMA model and baselines on BLEU score, diversity, length and Gunning-Fog Index. *** $p < 0.001$

Model	BLEU score	Diversity	Length	Gunning-Fog
CMA	**0.0910*** **	0.00089	**9.24*** **	**4.20*** **
Context seq2seq	0.0604	**0.00091**	6.76	3.94
No context seq2seq	0.0633	**0.00091**	6.56	3.96

We investigate the ability of our proposed model to generate diverse responses. For calculating Diversity, we use the Distinct-1 metric proposed by Li *et al.* [110]. Distinct-1 computes the number of distinct unigrams over the total number of generated tokens. We also report on the length of the generated responses and BLEU score in Table 7. **Finding 4:** We find that while the CMA model is less diverse than the baselines (but not statistically significant), it can generate longer, coherent sentences and significantly outperform baseline on the other metrics.

3.6.2.2 Readability Metrics

Table 7 shows the Gunning-Fog Index scores obtained between the existing conversation in the dataset and the conversation with the new response generated. **Finding 5:** The model achieves a significant increase in readability compared to baseline models. An ideal Gunning Fog index score is considered to be within the range of 7-8; hence there is room for improvement on this metric.

3.7 Discussion

We have shown how the long-term and working memory as described by cognitive architectures can be adapted to augment *seq2seq* models for dialogue generation. We find that the action selection mechanism can identify salient utterances and outperform extant methods to maintain the conversation context. We make a simplifying assumption that long-term (declarative) memory has at most 8 prior utterances made for practical reasons such as training time and compute resources. In future work, we plan to (a) incorporate more context, including world knowledge (e.g. Wikipedia) and (b) transformer models (e.g. BERT) [41].

We also proposed the adoption of a new metric to evaluate performance, the Gunning-Fog index. This index identifies the readability of the conversation and can be used alongside existing metrics for evaluation such as ease of answering to get a better understanding of the efficacy of output generated by the NLG system.

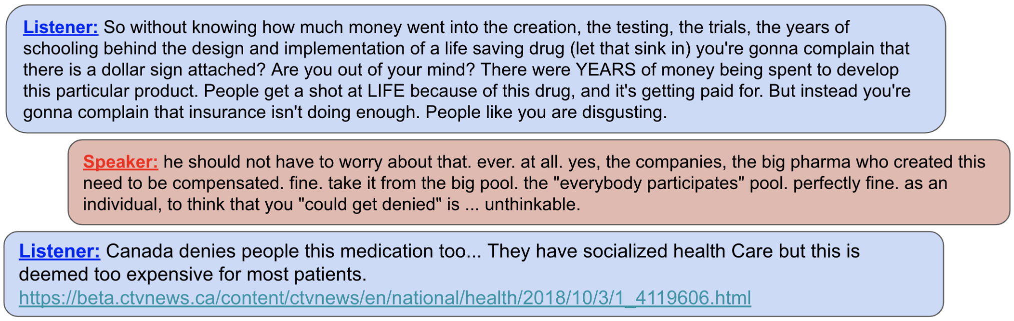
CHAPTER 4: PERSONALIZED AND KNOWLEDGE INFUSED DIALOGUE SYSTEMS

We have seen in Section 2.4 of Chapter 2 that the current state of conversational agents suffers from a myriad of issues. A couple of those issues relate to endowing a conversational agent with a personality so that agent can gain the user's confidence and trust. Apart from adding personality to conversation agents, another issue that plagues conversational agents is making responses more informative. In this chapter, we present a new model that infuses personality and knowledge into the conversational agent. Further, we also explore an important issue of factual consistency in dialog systems.

4.1 Introduction

One of the biggest challenges in conversational AI and dialog systems is building human-like conversational agents that are capable of generating *realistic*, *informative*, and *coherent* responses, so that users find them engaging and enjoy the ongoing conversation. Traditionally, conversational agents are built using RNN-based *seq2seq* models [211]. However, these models tend to generate vague and generic responses that are less engaging [111]. Recent advances in large-scale language models [164, 192, 165, 24] have pushed the state-of-the-art in Natural Language Generation (NLG), paving the way to use transformer-based models [208] in end-to-end dialog systems.

There have been several efforts [223, 66] to apply the large-scale language models to



Listener: So without knowing how much money went into the creation, the testing, the trials, the years of schooling behind the design and implementation of a life saving drug (let that sink in) you're gonna complain that there is a dollar sign attached? Are you out of your mind? There were YEARS of money being spent to develop this particular product. People get a shot at LIFE because of this drug, and it's getting paid for. But instead you're gonna complain that insurance isn't doing enough. People like you are disgusting.

Speaker: he should not have to worry about that. ever. at all. yes, the companies, the big pharma who created this need to be compensated. fine. take it from the big pool. the "everybody participates" pool. perfectly fine. as an individual, to think that you "could get denied" is ... unthinkable.

Listener: Canada denies people this medication too... They have socialized health Care but this is deemed too expensive for most patients. https://beta.ctvnews.ca/content/ctvnews/en/national/health/2018/10/3/1_4119606.html

Figure 11: An excerpt from a *Reddit* conversation between a speaker and a listener about a particular topic. As the conversation proceeds, a new piece of evidence is introduced by the listener through an URL.

build more engaging personalized conversational agents on the supervised Persona-Chat dataset [236]. These models can produce conversations that adhere to the reference profile facts but are devoid of unique personality and instead exhibit a mean average style [23]. Most recently, [23] introduced a dataset based on conversations from *Reddit* comments and built a conversational agent that conditions on a knowledge base of past reference conversations to model the speaker's persona. However, it only considers past dialogues and did not use any external knowledge to ground the generations.

In this work, we aim to improve dialogue's coherence and informativeness by incorporating local knowledge in a self-supervised framework for a large, web-scraped persona dataset. We use references to external links in the current dialog as the source for local knowledge. Indeed, local references for external knowledge widely exist in online conversations between humans. For example, we find that during conversations on platforms such as *Reddit*, users often use hyperlinked documents (e.g., by URLs) as additional pieces of evidence to ground their statements. Consider the example shown in Figure 18, a small snippet of a conversation between a speaker and a listener, where the listener posts a URL in the last turn. These hyperlinked documents usually contain relevant pieces of information that are closely related to the current conversation. Despite that, they were ignored or filtered out by previous work [239, 23].

Our primary goal here is to learn a model that can generate high-quality responses by modeling the past dialogues of the speaker as well as attending to any external document that has been referred to throughout the conversation. To do so, we present a dialog framework that combines the retrieval and generation process. We build upon [23] by using *Reddit* comments as our data source, and build an external knowledge base with the user-posted outbound links referenced throughout dialogues. We perform a K-Nearest-Neighbour (KNN) based search to retrieve relevant evidence phrases from the external documents and use them to context prime the model. Recent work by [56] also incorporates external knowledge into the conversational agents through information retrieval. Unlike their approach that uses Wikipedia, pre-defined images, and dialogue knowledge bases, our work ensures that diverse sources of knowledge are used by performing retrieval from hyperlinked documents introduced in a conversation.

We also find that limiting the search space for KNN to a local knowledge base, rather than a global knowledge base such as Wikipedia, ensures that the most relevant and informative context is retrieved when generating a response. Also, similar to [23] we incorporate persona into the responses using user's past dialogues to ensure that the generated response is consistent with the speaker's style of writing and their opinion on certain topics.

With the infusion of knowledge as a conditional parameter to the model, another issue that conversational agents need to address is the aspect of consistency. Consistency is considered an important factor to gain the trust of the user. Previous studies have evaluated a neural model's factual correctness in several fields. For example, for large pre-trained models, works such as [159, 126, 172] have shown that these generative models can memorize knowledge, but still generate factually incorrect responses. There is also active research in areas such as summarization [136, 130, 97], document-level generation [135] and NLG [48, 202]. In contrast, there is limited work within open-domain systems. [117] focused on decoding strategies to generate factually consistent responses. [143] proposed an annotation schema for the correctness and confidence of a model's response and calibrated the response such that its correctness aligns with its confidence. However, their work focused on a Q/A dataset and on the knowledge that a model has learned during training.

In this work, we conduct a thorough study to evaluate how factually consistent neural response generation models are with respect to the provided knowledge sentence retrieved from a knowledge base. We include a variety of GPT2-based models with different sizes, decoding strategies, and quality of knowledge sentences. For our annotation of factual correctness, we categorize responses into three types as shown in Figure 12: chit-chat responses that don't contain information that needs to be verified, factually consistent responses with respect to the input knowledge, and hallucinated responses that contain information not found in the input knowledge. Such annotations are different from previous work in that we are dealing with open-domain conversations, where there are chitchat responses, and we are using both dialog context and knowledge to evaluate responses. After showing that state-of-the-art neural response generation models do produce factually inconsistent responses, we propose to build a factual consistency detection model that is trained from an automatically generated corpus

...

Speaker 1: What do you think about Murray?

Speaker 2: I think Murray is a great player he just needs to stay healthy in order to compete more. Who do you like best?

Knowledge:

In tennis, the term Big Four refers to the quartet of men's singles players comprising Roger Federer, Rafael Nadal, Novak Djokovic, and Andy Murray.

Chit-Chat Response:

I like Federer and Nadal. I think Federer is the best.

Informative = 2; **F.C** = N/A; **H** = N/A

Factually Consistent Response:

Rafael Nadal is my favorite of the Big Four.

Informative = 3; **F.C** = 1; **H** = No

Hallucinated Response:

I like Djokovic. **He has played in the top ten singles players of the world.**

Informative = 4; **F.C** = 1; **H** = Yes

Figure 12: *Chit-Chat Response* does not include any information that needs to be verified and cannot be evaluated as consistent or not consistent. *Factually Consistent Response* is consistent with the provided knowledge. *Hallucinated Response* is not consistent with the knowledge but may still be correct. **F.C** = Factual Consistency. **H** = Hallucination

and demonstrates the competitive performance of the classifier. We make a distinction between factually **consistent** and factually **correct** responses. The former accurately portrays the input knowledge, and the latter is accurate with respect to the "world knowledge". Therefore factual correctness is a superset of factual consistency. Our detection model focuses specifically on factual consistency. Checking if a response is correct against "world knowledge" is an important problem that we leave for future exploration.

In summary, our contributions are as follows:

1. We propose a dialog framework that incorporates both local external knowledge and user's past dialogues to generate high-quality responses.

2. We present an approach to creating a dataset based on *Reddit* conversations,

which uses outbound links in the comments as the external knowledge.

3. We demonstrate that incorporating the local knowledge consistently improves *informativeness*, *coherency* and *realisticness* measures when compared to ground-truth human responses. Also, our model outperforms the state-of-the-art conversational agent on the *Reddit* dataset [23], as it exploits both external knowledge and the user's past dialogues.

4. We show that scaling up our model from 117M to 8.3B parameters consistently decreases the validation perplexity from 20.16 to 12.38 based on a vocabulary of 50K BPE subwords [184]. In particular, our 8.3B model generates high quality responses on par with human responses in terms of *informativeness*, *coherency* and *realisticness* evaluations.

5. A large-scale study with a thorough analysis of factual correctness for knowledge-grounded neural response generation models.

6. Release two datasets we prepared in this study: a human-annotated corpus on factual correctness from multiple neural response generation models; and the Conv-FEVER corpus that was adapted from the Wizard of Wikipedia dataset [43]

4.2 Dataset

4.2.1 Reddit Dataset

To create a large-scale dataset for self-supervised learning, we rely on the publicly available archive of *Reddit* comments that have been made available on pushshift.io

[4]. In our work, we use conversations extracted from a subset of months ranging from October 2018 to April 2019. We extract conversations as a sequence of turns by traversing through *Reddit*'s comment graph structure. To ensure that the large volume of comments is of high quality, we apply the filtering strategy proposed by [23] and add other conditions to further improve the quality of the conversations. Adding all these filtration rules together, we extract conversations based on the following conditions:

1. The conversation has a minimum of 5 turns.
2. The conversation has a maximum of 15 turns.
3. At least one turn has a minimum karma score of 4 within the conversation.
4. All turns in the path have at least 3 words.
5. The conversation shares a maximum of 2 turns with previously extracted paths.
6. No turns in the path originates from a "Not Safe For Work" subreddit.
7. No user in the conversation is marked as "Deleted".

We process each month individually in parallel. Once all the conversations were extracted from a specified month, we then extract all the URLs mentioned in each turn of a conversation to create the knowledge base of hyperlinked documents (Ext-Docs knowledge-base). The URLs are filtered out based on an undesirable list of domain

[4] `https://files.pushshift.io/reddit/comments/`

names and extensions. We use the two-block lists found in the Megatron-LM repository [5].

Overall, we extracted 48M conversations and found that 10.4% of the conversations had used a URL as a piece of evidence in the conversation. To create a more balanced dataset between conversations that use no URLs and conversations that use URLs, we downsample the conversations with no URLs. After downsampling, we ended up with a total of 1,585,875 conversations where 1,232,244 of these conversations had no URLs and 353,631 conversation had used URLs. We further split the filtered dataset with an 80-10-10 ratio to create the training, validation, and test sets.

Additionally, we precomputed all the past dialogues made by users across the time span of our dataset (2018-10 to 2019-04) and stored them. In the final dataset, we had 593,734 unique users and on average each user had around 21.13 historical comments.

4.2.2 Conv-FEVER Corpus

To create the Conv-FEVER Corpus, We leverage Wizard of Wikipedia(WoW) [43], a knowledge grounded dialog dataset generated through MTurkers who play the role of wizard and apprentice. The wizard has access to Wikipedia passages and the apprentice is given the role of learning more about a topic by engaging in a dialog with the wizard. At every turn, the wizard selects a knowledge sentence from the Wikipedia passages to generate a knowledge-grounded turn. The wizard's responses are based on knowledge, so we hypothesize that they are consistent with respect to the knowledge. To generate inconsistent responses, we leverage a few data augmentation

[5] https://github.com/NVIDIA/Megatron-LM

strategies introduced in [97], including random pairing, negation, entity swapping.

Random Pairing (R): We perform two types of random pairing: (1) Replace the response with a response from a random dialog; (2) Replace the annotated knowledge sentence with a knowledge sentence from a random dialog.

Negation (N): We perform two types of negation: (1) Negation applied on the response; (2) Negation applied to the annotated knowledge. [6]

Entity Swapping (E): We performed two types of entity swapping: (1) Entity swapping on the context if there is a common entity mentioned in the context and the response; (2) Entity swapping on the knowledge if there is a common entity mentioned in the knowledge and the response. The common entity is replaced by an entity of the same type. We tag entities using the SpaCY NER tagger [74]. Table 8 shows the statistics of the data set.

Table 8: Conv-FEVER dataset statistics

Dataset	Num. Consistent	Num. Inconsistent
WoW	68957	-
Random Pairing	-	137914
Negation	-	107845
Entity Swapping	-	73178

4.3 Architecture

4.3.1 Knowledge Integration

Consider the conversation $\{X_i\}_{i=1}^{n-1}$, where X_i is a turn in the conversation between two or more users. The task is to generate the turn X_n for user A (i.e., the speaker

[6] We apply negation for these tokens: are, is, was, were, have, has, had, do, does, did, can, ca, could, may, might, must, shall, should, will, would

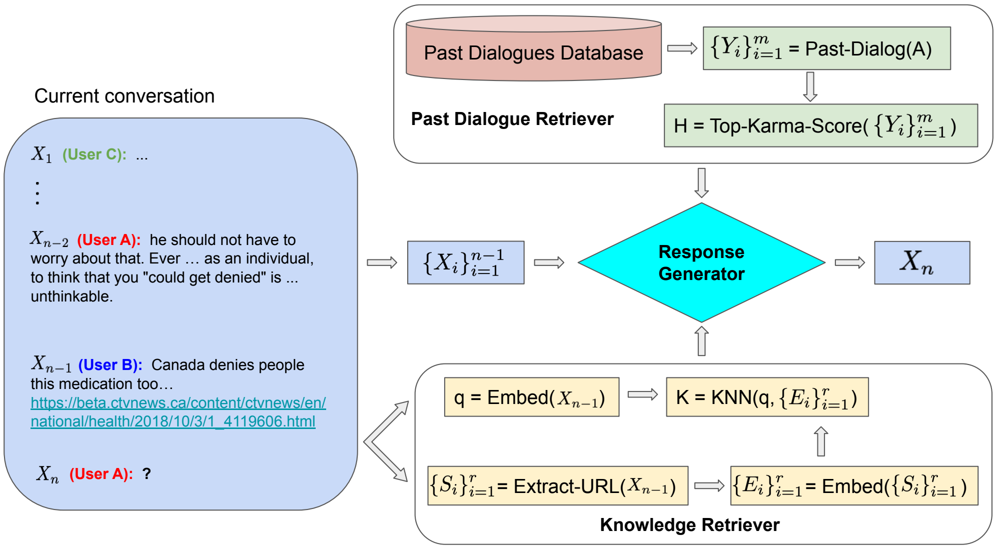

Current conversation
X_1 (User C): ...
⋮
X_{n-2} (User A): he should not have to worry about that. Ever … as an individual, to think that you "could get denied" is ... unthinkable.
X_{n-1} (User B): Canada denies people this medication too…
https://beta.ctvnews.ca/content/ctvnews/en/national/health/2018/10/3/1_4119606.html
X_n (User A): ?
Past Dialogues Database
$\{Y_i\}_{i=1}^m$ = Past-Dialog(A)
Past Dialogue Retriever
H = Top-Karma-Score($\{Y_i\}_{i=1}^m$)
$\{X_i\}_{i=1}^{n-1}$
Response Generator
X_n
q = Embed(X_{n-1})
K = KNN(q, $\{E_i\}_{i=1}^r$)
$\{S_i\}_{i=1}^r$= Extract-URL(X_{n-1})
$\{E_i\}_{i=1}^r$= Embed($\{S_i\}_{i=1}^r$)
Knowledge Retriever

Figure 13: Architecture diagram of our framework consisting of the following components: (i) Knowledge Retriever: helps retrieve relevant sentences K from the URLs; (ii) Past Dialogue Retriever: retrieves past dialogues H from user A who is generating our response; (iii) Response Generator: a GPT-2 model that is to be finetuned and take the knowledge retrived along with past dialogues and the current conversation as input.

in Figure 18) given the current conversation. It is done by using our framework illustrated in Figure 20, which consists of three components:

1. **Knowledge Retriever:** To include external knowledge, we consider X_{n-1}, the last turn in the current conversation, and extract the information referenced by the outbound URL links. The extracted knowledge is then divided into sentences $\{S_i\}_{i=1}^r$ and each sentence S_i is encoded to a fixed size vector E_i using Universal Sentence Encoder [28], denoted by USE. The knowledge retriever encodes the last turn of the conversation X_{n-1} as query q using the same USE embedding, and performs a cosine similarity search between $\{E_i\}_{i=1}^r$ and q. Then, it picks k sentences $K = \{S_i\}_{i=1}^k$ with the highest similarity scores. We simply set $k = 5$ in all experiments. In our framework, we pre-compute all the sentence embeddings

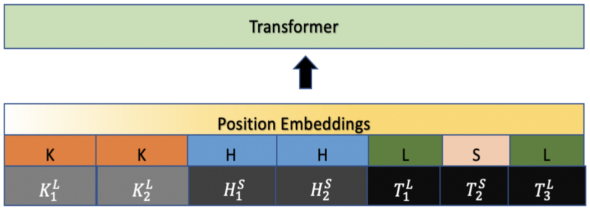


Figure 14: Illustration of input representation to the GPT-2 transformer model for a conversation between a listener (L) and speaker (S) (to be modeled). Along with the subword embeddings of current conversation (T_i), the model also receives past dialogues from the speaker's history (H_i^s) and most relevant knowledge sentences (K_i^L) introduced by listener. We also add positional embeddings and token type embeddings K, H, L, S for knowledge, past dialogues, listener, and speaker, respectively.

$\{E_i\}_{i=1}^r$ associated with all the external URLs and build a knowledge-base called Ext-Docs which includes all the documents referenced within the dataset.

2. **Past Dialogue Retriever:** We denote $Y = \{Y_i\}_{j=1}^m$ as the past dialogue turns associated with speaker A. Note that past dialogues contain all the historical comments made by each user but do not contain any utterances from the current conversation. To retrieve the relevant and high-quality past dialogues emblematic of a user's personality, we follow the heuristic strategy used by [23]. We sort the past dialogues based on their karma score in *Reddit* (which is the difference between the up-votes and down-votes of a comment) and pick the ones with the highest scores. We denote the retrieved past dialogues as H.

3. **Response Generator:** Traditionally, the goal of the response generation component has been to produce an informative response X_n conditioned on the current conversation turns $\{X_i\}_{i=1}^{n-1}$. However, just incorporating these turns might not provide enough information to produce an informative response [56, 223]. For the response generator to make use of the retrieved knowledge

> and past dialogues, we concatenate them as part of the conditional input for a left-to-right GPT-2 language model [164], in which the input context size is 1024 in our experiments. The retrieved knowledge and past dialogue sequences are truncated to a maximum of 250 tokens each, and the current conversation is allocated a minimum of 524 tokens in case there is no past dialog for a particular user or outbound URL links in the current conversation.

We illustrate the input representation to the GPT-2 response generator in Figure 14. In addition to the positional embedding, we tell the model which sequence of tokens is from the speaker of interest or listeners in the current conversation, which are retrieved external knowledge, and which are speaker's past dialogue. This is achieved by adding token type embeddings to the positional encoding and subword embeddings.

4.3.2 Consistency Evaluator

We train a factual consistency detector on the Conv-FEVER dataset. We use the BERT-base [39] model and initialize our detector by first training on the FEVER dataset [203] taken from the set of tasks presented in KILT [158]. The FEVER task is aimed at determining if a claim can be supported or refuted given a Wikipedia document. Claims that are labeled as supports can be thought of to be consistent and refutes can be thought of as being inconsistent. To create our initial training corpus, we extracted all data points in the FEVER corpus that contained a pointer to the ground truth Wikipedia documents as evidence. In total, we trained on 48,451 supports and 18,625 refutes [7]. We then finetune this model on the Conv-FEVER

[7] This number differs from the original since we dropped data points without pointer to Wikipedia documents.

dataset. The model takes in the dialog context and knowledge along with the response and determines if the response is support (consistent) or refute.

4.4 Experiment

4.4.1 Models

We investigate four different models to demonstrate the benefits of incorporating past dialogues and local knowledge:

1. **Baseline (B):** The simplest of the four models used in our experiments, which is used to establish a baseline. In this model, only the current conversation, i.e., $\{X_i\}_{i=1}^{n-1}$, is provided as an input sequence to the response generator. Despite its simplicity, it is a strong baseline for response generation as demonstrated by [239].

2. **Baseline + Past Dialogues (B + H)**: This model is the state-of-the-art response generation approach presented by [23]. In this model, a heuristic-based approach is used to identify the retrieved past dialogues of a speaker, which is then combined with the current conversation. The retrieved past dialogue (denoted as H) is concatenated with the current conversation as the input to the response generator.

3. **Baseline + Knowledge (B + K):** This setting measures the importance of adding external knowledge for the response generation process. In this model, we combine retrieved knowledge sentences from the external URLs (denoted by K) and concatenate them as additional pieces of evidence to the current

conversation.

4. **Baseline + Knowledge + Past Dialogues (B + K + H)**: This setting measures the importance of incorporating both external knowledge and retrieved past dialogues for the response generation process. In this model, we combine retrieved knowledge sentences K from the external URLs and the retrieved past dialogues H from a user that is being modeled. We concatenate them as additional pieces of evidence to the current conversation.

4.4.2 Automated Metrics

Automatic evaluation for the quality of generated responses is still an active area of research for open-domain conversation. Previous work has used metrics such as BLEU [157], METEOR [9], ROUGE [120] from machine translation and text summarization [123] tasks, although several works have demonstrated that they don't correlate well with human judgments for open-ended tasks such as dialogue [124]. In this work, we report the BLEU score following established reporting practices. We also report the perplexity (PPL) on the validation set as a measure to compare different models, which was found to correlate with fluency in generations in a previous study.

4.4.3 Human Evaluation

Human evaluation is viewed as the most effective way of evaluating the quality of the generated text. Traditionally, human evaluation is conducted through the use of Likert scales [119] or continuous scales as the primary experiment design. However, prior research has shown that the usage of Likert scales affects the quality of ratings obtained from the human annotators [154, 180], and the usage of continuous scales such as magnitude estimation is prone to cognitive bias [177]. To avoid these issues, we provide pairs of conversations side by side with the last turn generated by either the model or the human and ask the annotator to choose between the two. We also provided a tie option. Overall, we randomly sample 100 conversations from the test set for our evaluations. The annotators are asked to evaluate the quality of the responses according to the following metrics:

1. **Informativeness** measures whether the response from the speaker is informative for listeners (i.e. contains more detailed information).

2. **Coherence** measures whether the response from the speaker matches the topic and discussion from the earlier context of the conversation.

3. **Realistic** measures whether the response from the speaker looks like a response from a real human instead of a bot.

We utilize 5 unique workers per example in our evaluations. To obtain high-quality human labels from native English speakers, the workers are required to reside in the United States and have a Human Intelligence Task (HIT) approval rate greater than or equal to 95%. We explicitly state in the instructions that payment is contingent on raters spending at least 25 seconds per assignment. We tried to filter the inexperienced raters based on their past *Reddit* use as in previous study [23], but we found this is less effective as the raters tend to select the maximum hours we provided in our survey.

To evaluate factual consistency in conversational agents, we use four GPT2 [164] based models (small, medium, large, XL) and three different decoding mechanisms:

Nucleus sampling [72] with p=0.9, Beam-Search, and Delayed Beam-Search(DBS) [135] which uses top-k sampling for n delay steps followed by beam search. We use k=10, 5 delay steps, and a beam size of 5. We use two different configurations for the knowledge sentence provided to the model: 1) the ground truth knowledge that is provided in the WoW test set and 2) Dense Passage Retrieval (DPR) model from [82] where the knowledge base consists of 21 million Wikipedia articles. We sample dialogs of length 5 turns from the WoW test sets and obtain 100 responses for each model resulting in 2400 responses in total.

We annotated these system-generated responses based on the following setup: We propose a two-stage human annotation setup for factual correctness in the context of open-domain agents. The annotators are given a dialog context along with a knowledge sentence and evaluate the system-generated response. Stage 1 involves evaluating **Appropriateness** and **Informativeness** on a Likert scale 1-5. We define appropriateness as relevant to the dialog context and informativeness as being detailed and informative. This stage is used to filter out incoherent and chit-chat responses. If a response scores low on informativeness, it is categorized as a chit-chat response.

Stage 2 of our setup involves evaluating **Factual Consistency** and **Hallucination**. We pose the Factual Consistency question: *Is the response generated factually accurate with regards to the input knowledge?* with a three-point scale: factually incorrect(0), partially correct(0.5), and completely correct(1). For Hallucination: *Is the response generated making up more information than what is provided in the conversational context and input knowledge?*, we collect a binary label whether each response contains any hallucinated information. We follow the recommendations on describing a human annotation setup outlined in [77]. Figure 12 shows an example annotation.

After the completion of stage 1, we filter out responses whose appropriateness and informativeness scores are below 3 to ensure we have responses that contain some form of knowledge and are relevant to the dialog context. We are left with 1684 responses to be annotated in stage 2 (959 responses using the ground-truth knowledge and 725 outputs using knowledge from DPR).

4.5 Results

In this section, we report the results of automatic and human evaluations detailed in the previous section.

4.5.1 Automated Metrics

Table 9 provides a comparison of the different models used in our experiments. We find that compared to the baseline model (B), the addition of knowledge or past dialogue reduces the validation perplexity. In particular, adding past dialogue information can improve the perplexity significantly. We also notice that the best perplexity is achieved by adding both retrieved knowledge and past dialogues as additional pieces of evidence. The BLEU score degrades when we add knowledge and past dialogue separately but slightly improves as we incorporate them together. As we will demonstrate in human evaluation results, these BLEU scores don't correlate well with human judgments. We don't report the BLEU score further.

We also performed ablation studies on the best performing model (B + K + H) for various model sizes. Table 10 gives the different configurations of models that were trained. We find that validation perplexity drops significantly as we increase the size

Table 9: Automated metrics results (Val PPL and BLEU) on the test set obtained by fine-tuning the 345M model with different experimental settings. **B**: stands for baseline model that only exploits current dialog context. **H**: stands for the heuristic approach for retrieving past-dialogues. **K**: stands for retrieval of knowledge. ↑ means the number is the higher the better, and ↓ means the number is the lower the better.

Models	Val PPL(↓)	BLEU(↑)
B	18.12	15.3
B + K	18.10	14.1
B + H	16.84	14.0
B + K + H	**16.83**	**15.4**

Table 10: Scaled up results for our best performing model (B + K + H). ↓ means the lower value is the better.

Model	Hidden size	Layers	Attention heads	#Parameters	Val PPL(↓)
B + K + H	768	12	12	117M	20.16
B + K + H	1024	24	16	345M	16.83
B + K + H	1536	40	16	1.2B	14.57
B + K + H	3072	72	24	8.3B	**12.38**

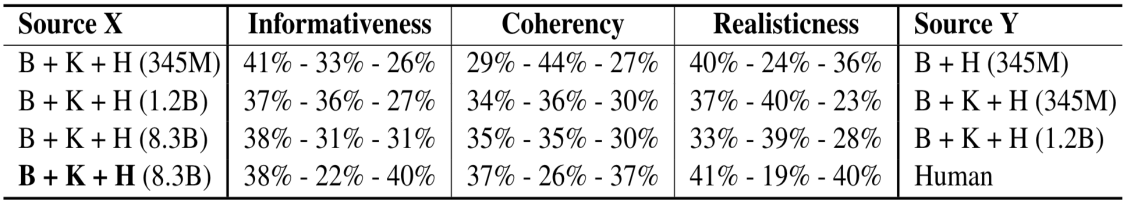
Table 11: Pairwise comparison results (X wins - Ties - Y wins) between 345M models and human-generated text using Mechanical Turk. B: stands for baseline model that only exploits current dialog context. H: stands for the heuristic approach for past dialogues. K: stands for retrieval for knowledge. To make a relative comparison between models, we highlight the last columns of the results (best viewed in color). They indicate the percentages of cases that the models are outperformed by humans, which are the lower the better.

Source X	**Informativeness**	**Coherency**	**Realisticness**	**Source Y**
B	28% - 20% - 52%	29% - 22% - 49%	31% - 33% - 36%	Human
B + K	31% - 26% - 43%	30% - 27% - 43%	26% - 36% - 38%	Human
B + H	29% - 31% - 40%	26% - 33% - 41%	29% - 21% - 50%	Human
B + K + H	34% - 29% - 37%	29% - 33% - 38%	26% - 39% - 35%	Human

of the models. These results are consistent with prior studies [192].

Human Evaluation

We report the human evaluation results for different models in Table 11. Specially, we compare the generated responses from these models to human responses. To make relative comparisons between models, we highlight the last column of the results (X



Source X	Informativeness	Coherency	Realisticness	Source Y
B + K + H (345M)	41% - 33% - 26%	29% - 44% - 27%	40% - 24% - 36%	B + H (345M)
B + K + H (1.2B)	37% - 36% - 27%	34% - 36% - 30%	37% - 40% - 23%	B + K + H (345M)
B + K + H (8.3B)	38% - 31% - 31%	35% - 35% - 30%	33% - 39% - 28%	B + K + H (1.2B)
B + K + H (8.3B)	38% - 22% - 40%	37% - 26% - 37%	41% - 19% - 40%	Human

Figure 15: Pairwise comparison results (X wins - Ties - Y wins) between our best performing model (B + K + H) and a state-of-the-art model on *Reddit* data (B + H) [23]. We also include pairwise comparisons with different model sizes. We find the larger model always outperforms the smaller one across all three metrics. In particular, Our model with 8.3B parameters can generate high-quality responses on par with human responses.

wins - Ties - Y wins); they indicate the percentages of cases where the models were outperformed by humans, thus the lower the better. We draw the following observations:

1. Adding external knowledge significantly improves the informativeness and coherency metrics for both the baseline model (B vs. B + K), and previous state-of-the-art model (B + H vs. B + K + H).
2. Incorporating past dialogues also improves both the informativeness and coherency measures for baseline models (B) and (B + K).
3. Our model (B + K + H) outperforms others, including the state-of-the-art model (B + H) on the *Reddit* dataset [23], in terms of informativeness, coherency and realistic measures.

In Table 15, we perform comparison between models, including pairwise comparison between our method (B + K + H) and previous state-of-the-art model (B + H) for this task. We also scale our model up to 8.3 billion parameters and report the human evaluations results in Table 15. We find consistent improvements of all evaluated metrics when we increase the size of the model. Noticeably, our 8.3 billion models can generate responses with quality comparable to humans in terms of informativeness, coherency, and realistic metrics.

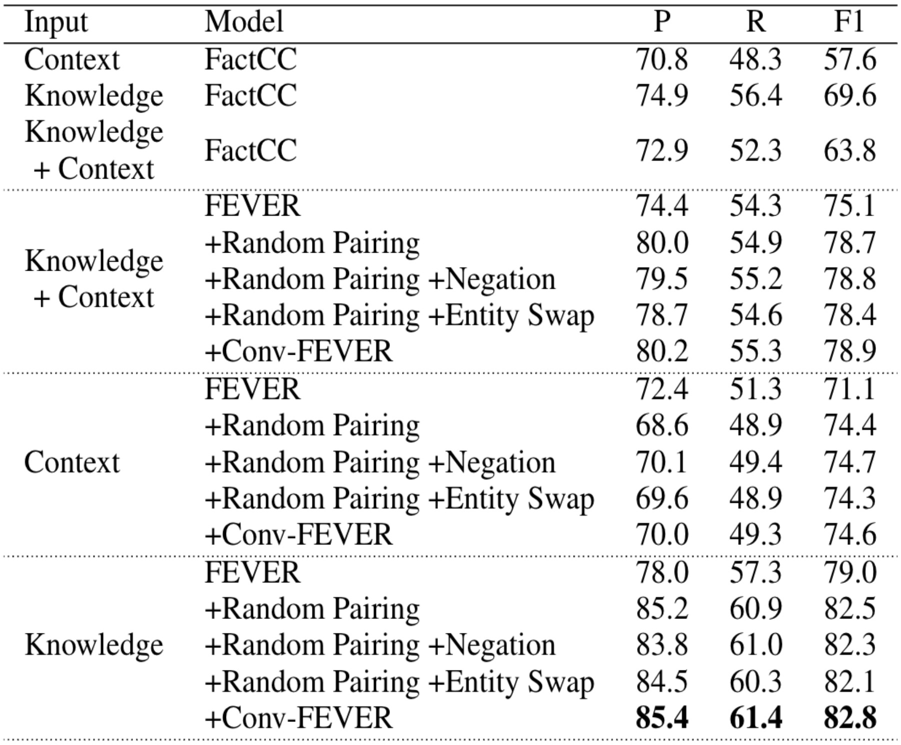
Case Study

Table 12 displays a conversation between a speaker and listener where the last turn of the conversation is generated by the model. We also show the top two retrieved sentences from the external URL that is used to generate the response. From the generated response, our model can make use of the relevant spans of knowledge such as "The men training for less than 3 months, on average, squatted 102kg (225lbs)" and "The men training for less than 3 months, on average, benched 85kg (185-190lbs)".

4.5.2 Factual Consistency Results

Table 16 shows the detector's performance on this cleaned test set. We present results for training on FEVER versus training on both the FEVER and Conv-FEVER. We compare our model against FactCC [97] which is also a BERT-based model used to predict factual consistency for neural summarization models. We see that training on a dialog dataset with synthetically generated negative examples outperforms just training on a document-level dataset. Additionally, our best-performing model requires just using knowledge as input. We believe this is because most of the information contained in the response comes from knowledge.

Table 17 shows the results of our annotations. We see that as models get larger, there is an increase in factual consistency and a decrease in hallucinated responses. In a more realistic setting where DPR is being used to retrieve knowledge from a



Input	Model	P	R	F1
Context	FactCC	70.8	48.3	57.6
Knowledge	FactCC	74.9	56.4	69.6
Knowledge + Context	FactCC	72.9	52.3	63.8
Knowledge + Context	FEVER	74.4	54.3	75.1
	+Random Pairing	80.0	54.9	78.7
	+Random Pairing +Negation	79.5	55.2	78.8
	+Random Pairing +Entity Swap	78.7	54.6	78.4
	+Conv-FEVER	80.2	55.3	78.9
Context	FEVER	72.4	51.3	71.1
	+Random Pairing	68.6	48.9	74.4
	+Random Pairing +Negation	70.1	49.4	74.7
	+Random Pairing +Entity Swap	69.6	48.9	74.3
	+Conv-FEVER	70.0	49.3	74.6
Knowledge	FEVER	78.0	57.3	79.0
	+Random Pairing	85.2	60.9	82.5
	+Random Pairing +Negation	83.8	61.0	82.3
	+Random Pairing +Entity Swap	84.5	60.3	82.1
	+Conv-FEVER	**85.4**	**61.4**	**82.8**

Figure 16: Factual detector performance comparing the state of the art model against the model trained on our Conv-FEVER dataset

larger knowledge base, the highest factual consistency score is 0.72, and at the lowest 18.0% of responses have hallucinated information, indicating room for improvement in factual consistency. Additionally, we see in the DPR setting beam search performs better for smaller models, whereas for larger models DBS is the best. Massrelli *et al.* [135] showed DBS performed the best for factual verification across all model sizes; however, that was for document-level generation.

		GPT2-Small		GPT2-Medium		GPT2-Large		GPT2-XL	
Knowledge Retriever	Decoding Strategy	F ↑	H ↓	F ↑	H ↓	F ↑	H ↓	F ↑	H ↓
Ground-Truth Knowledge	Top Beam	0.77	18.0%	0.78	21.3%	0.82	18.3%	0.88	10.4%
	Nucleus Sampling	0.78	22.9%	0.79	15.1%	0.87	9.0%	**0.90**	**5.0%**
	DBS	0.79	23.7%	0.84	11.7%	0.82	11.3%	0.86	8.2%
DPR Knowledge	Top Beam	0.61	31.0%	0.64	29.9%	0.64	30.4%	0.63	27.8%
	Nucleus Sampling	0.55	41.0%	0.68	26.2%	0.68	20.9%	0.66	28.0%
	DBS	0.59	34.6%	0.52	31.8%	0.69	27.1%	**0.72**	**18.0%**

Figure 17: Human Annotation Results. We use Krippendorff's alpha for inter-annotator agreement (IAA) F = Factual Consistency(0, 0.5,1) where IAA=0.44. H = Hallucination. where IAA=0.64

Table 12: Conversation between a human speaker and a human listener where the last turn of the conversation is the response generated by our 8.3B model which makes use of knowledge introduced by the listener through the URL. The top retrieved sentences from the document are also shown with the generated response. The generated response is coherent with the conversation context and consistent with the retrieved knowledge.

	Conversation 1
Listener	I literally progressed linearly (with SL 5x5) to a 2 plate bench and 3 plate squat in roughly 3 months while deployed to Iraq and eating roughly at caloric maintenance (not by own volition.)Went from being unable to bench 185.
Speaker	You better get into professional sports then because apparently you're in the top 0.001% of genetics. Not sure how you expect anyone to believe that not only did your lifts increase in probably the 99th percentile in terms of progress, but you also did it without eating enough food to build muscle. LMAO look around you goober. People constantly post their progress here and it's nowhere near what you're describing. Here's a post from a day ago where it took someone 7 months to go from a 65-pound bench to 135. Go look at the progress posts in /r/fitness. If anyones misinformed on what the average new lifter can expect in terms of progress, it's you. You suggesting that is normal to add 5 pounds to the bar every session to hit 2/3/4 plate lifts is laughable. The fact you said you did it eating at maintenance is even more hilarious. If you posted in /r/fitness what you just posted here, you'd get laughed out of that subreddit. It's not even close to realistic to expect someone to LP to a 315 pound squat. Like not even close. You also completely ignored the fact you contradicted yourself by calling me a beginner yet in the same breath you say that once you can't LP anymore, you're an intermediate. Here's an article on average weight lifting progression. Not only are the numbers you just described far above what the average person can expect in a year of training, you supposedly did it in 3 months while eating in caloric maintenance.
Listener	`https://www.strongerbyscience.com/realistic-training-goals/` 16% of lifters training for 3-6 months reached a 264lb bench and 373lb squat. Cheers!
Speaker (Model)	If you read your own article, you'd notice the vast majority of people hit a set of 1-2 plates in 3-6 months. The average raw beginner could easily hit a ∼**200 pound squat** and a **190 pound bench within 3 months** of starting lifting, and that's assuming they're eating the bare minimum to build muscle. Retrieved Knowledge 1: The men training for **less than 3 months**, on average, **squatted** 102kg (**225lbs**), and the men training for between 3 and 6 months squatted 132kg (290lbs) on average. Retrieved Knowledge 2: The men training for **less than 3 months**, on average, **benched** 85kg (185-**190lbs**), and the men training for between 3 and 6 months benched 96kg (210lbs)on average, for a difference of about 3.4kg (7-8lbs) per month.

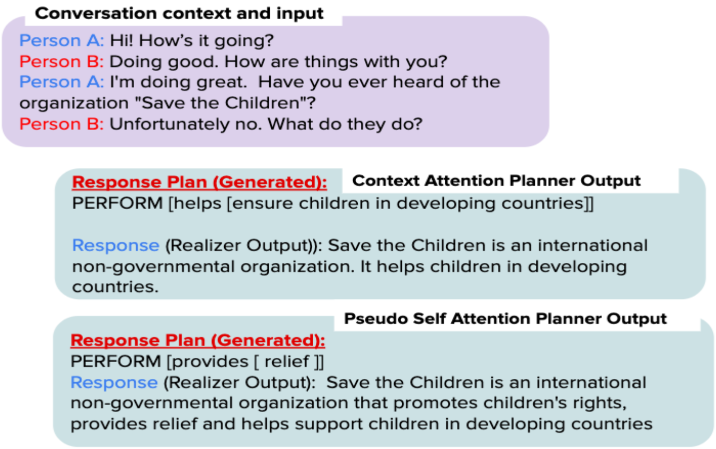
CHAPTER 5: LEARNING TO PLAN AND REALIZE SEPARATELY FOR OPEN-ENDED DIALOGUE SYSTEMS

5.1 Introduction

Recent advancements in the area of generative modeling have helped increase the fluency of generative models. However, several issues persist: coherence of output and the semblance of mere repetition/hallucination of tokens from the training data [147, 222]. One reason could be that the generation task is typically construed as an end-to-end system. This is in contrast to traditional approaches, which incorporate a sequence of steps in the NLG system, including content determination, sentence planning, and surface realization [167, 169]. A review of literature from psycholinguistics and cognitive science also provides strong empirical evidence that the human language production process is not a monolith [38, 21, 22, 85].

Prior approaches have indeed incorporated content planning into the NLG system, for example data-to-text generation problems [161, 148] as well as classic works that include planning, based on speech acts [34] (for an in-depth review c.f. [59]). Our work closely follows these prior approaches, with one crucial difference: our planners are not based on dialogue acts or speech acts.

Consider the example in Fig.18. An input utterance by Person B, a statement (*Unfortunately no.*), followed by a question (*What do they do?*), can be effectively responded to using plans, learned and generated, before the realization phase. The



Conversation context and input

Person A: Hi! How's it going?
Person B: Doing good. How are things with you?
Person A: I'm doing great. Have you ever heard of the organization "Save the Children"?
Person B: Unfortunately no. What do they do?

Response Plan (Generated): Context Attention Planner Output
PERFORM [helps [ensure children in developing countries]]

Response (Realizer Output)): Save the Children is an international non-governmental organization. It helps children in developing countries.

Pseudo Self Attention Planner Output
Response Plan (Generated):
PERFORM [provides [relief]]
Response (Realizer Output): Save the Children is an international non-governmental organization that promotes children's rights, provides relief and helps support children in developing countries

Figure 18: Example conversation between two speakers A & B where the response for the speaker B is generated based on the response plan from two learned planners: Context Attention and Pseudo Self Attention.

realization output can then include the mention of *provides relief*, consistent with the generated plan (*PERFORM [provides [relief]]*).

Dialogue acts [197] (e.g., statements, questions), by their nature, encompass a wide variety of realized output, and hence cannot sufficiently constrain the language model during the generation process. Research has addressed this issue by adapting existing taxonomies [197] towards their own goals [224, 156]. We instead use an adapted and extended form of lexical-conceptual structures (LCSs) to help constrain the realization output more effectively [44].

Our work makes the following contributions:

- We investigate the impact of separating planning and realization in open-domain dialogue and find that the approach produces better responses per automated metrics and detailed human evaluations.

- We propose the use of LCS-inspired representations based on asks and framings, which in turn are grounded in conversation analysis literature, to generate plans, instead of using dialogue acts.

- We release corpora annotated with plans for all utterances, using three planners, including symbolic planners and attention-based planners.

5.2 Approach

5.2.1 NLU using Asks and Framing

The representation we use to generate plans leverages *asks* and *framings* based on conversation analysis literature [160, 175, 182]. An *ask* is closely related to the notion of a request [233]. Perhaps most importantly, an ask elicits relevant responses from the recipient. *Framing* refers to linguistic and social resources used to persuade the recipient of an ask to comply and perform the requested social action. Put another way, an ask creates a social obligation to respond, while framing provides an adequate basis for compliance with the ask.

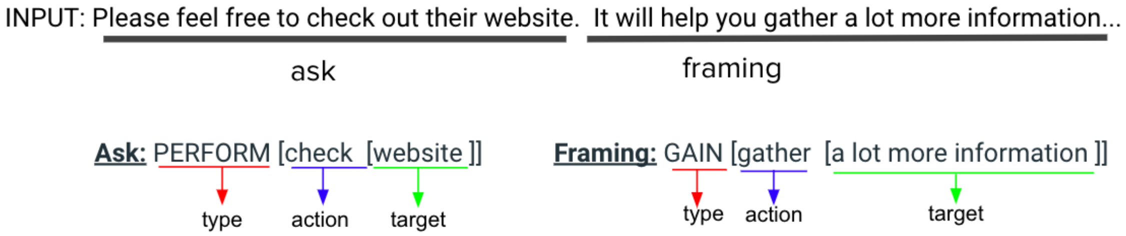

INPUT: Please feel free to check out their website. It will help you gather a lot more information...
ask
framing
Ask: PERFORM [check [website]]
type
action
target
Framing: GAIN [gather [a lot more information]]
type
action
target

Figure 19: Example of ask and framing representations used as training for generation of Response Plans.

In Fig.19, we show the ask/framing representational formalism that serves as the basis of our response plans. Here the *ask* is a request to PERFORM the action of *check out the website*. The perceived risk or reward (or *framing*) for this request is that, upon performing the action, one may GAIN something, i.e., *gather a lot more information*. We use two types of *asks*: GIVE (provide something or information)

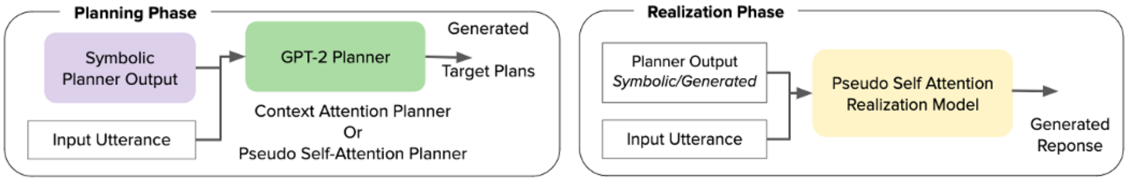
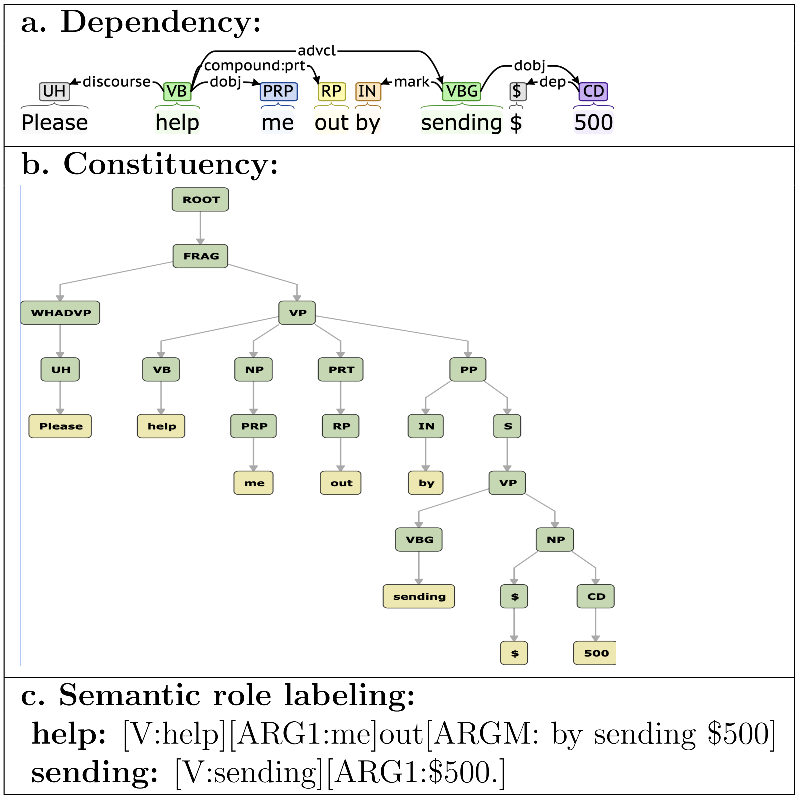


Figure 20: Architecture diagram of our system consisting of two phases: Planning and Realization. The Planning phase (Context and Pseudo Self Attention) encodes the input sequence and symbolic planner input to produce the response plans. The Realization phase uses the response plan and input utterance to generate the response

and PERFORM (perform an action), and two types of *framings*: GAIN (gain some benefit) and LOSE (lose benefit or resource). This preliminary ontology was motivated by conversation analysis literature [176, 35, 54]: by treating utterances as actions, we can establish what each utterance seeks to accomplish and how a sender motivates the recipient in terms of the benefits and costs of compliant responses.

5.2.2 Symbolic Method

a. Detect Ask/Framing actions: The first step for ask/framing detection is to extract the main action for each clause (recursively) from the dependency tree and the constituency parse shown in Figure21a,b. This is achieved first through the application of basic language tools and also through the application of CATVAR to detect actions that may be implicit in non-verbal forms, such as *reference* (which maps to the PERFORM form *refer*). Verbal constraints are then applied to rule out past and progressive actions (VBD/VBG) as asks. If ruled out, the action is considered as a framing candidate. If not ruled out, a priority scheme is applied, attempting to match the action against asks PERFORM and GIVE, in that order, from the lexical resource of interest (the thesaurus or LCS/LCS+). If this fails, an attempt is made to match the action against framings LOSE and GAIN, in that order, using the same



a. Dependency:

b. Constituency:

c. Semantic role labeling:

help: [V:help][ARG1:me]out[ARGM: by sending $500]

sending: [V:sending][ARG1:$500.]

Figure 21: Dependency, Constituency, and Semantic Role Labeling for *Please help me out by sending $500*

lexical resource.

This priority scheme was devised to support overlapping ask actions, e.g., *send* is both a GIVE and PERFORM in LCS+, but in the context of a clickable link, it is deemed a PERFORM. In this way, *structural knowledge* influences the *linguistic choice* of ask. Similar overlap exists for framing, e.g., *retrieve* is both a GAIN or a LOSE, depending on the perspective of interest. Given that our application of ask detection is designed for SE interactions, it is assumed that a loss is intended for the potential victim (not for the social engineer); thus, LOSE is tested ahead of GAIN.

b. Determine Ask/Framing Arguments: Following the detection of an ask or framing action, basic language tools identify the arguments. For example, semantic role labeling identifies *$500* as ARG1 in the sentence *sending $500* (see Figure21); this becomes an ask argument that is subsequently assigned an ask category as described next.

c. Assign Ask/Framing Category: Categories are associated with asks and framings, e.g., *sending $500* yields a GIVE ask with argument $500, which is in the finance_money category. Other examples are shown below:

- *...using your gift card*: scam_gift
- *Sign-up with your login and password*: credentials
- *...confirm with us via this email...*: personal

The categories are hierarchically organized, with a total number of 13 categories. From this categorization it is possible to deduce the likely goals of a would-be attacker, for use in downstream response generation.[8]

d. Detect Links: Links are detected through either basic or advanced link processing and these are associated with ARG1 of the ask (found by basic processing tools). The existence of a link boosts the confidence score for its associated ask. For example, a detached link is found via advanced link processing for *Contact me. I'm around Mon. (jw11@example.com)*. Here, *me* is associated with the contact email address.

e. Apply Confidence Score: Application of confidence scores is based on preliminary trial-and-error studies and intuitions gleaned from processing development data. Observations are: (1) Past tense events are found not to be asks, thus assigned low or 0 confidence; (2) Non-past-tense events are more prevalently observed to be PERFORM asks if an ask category is specified (e.g., *finance_money* for *sending $500* above), thus

[8] Although the full description of ask/framing categories is out of scope for this paper, these categories provided hints to the human adjudicator for the generation of our validation set.

assigned high confidence (0.8); (3) The vast majority of asks associated with URLs (e.g., jw11@example.com tied to *me* above) are found to be PERFORM asks, thus assigned a highest confidence (0.9); (4) GIVE combined with any ask category (e.g., *contribute $50* above) is less frequently found to be an ask, thus assigned slightly lower confidence (0.75); and (5) GIVE by itself is even less likely found to be an ask, thus assigned a confidence of 0.6 (e.g., *donate often*). (Automatic confidence scoring, training on actual data, is an area of future work.)

f. Select Top Ask: Upon completion of the processing above, *Top Ask* selection produces the most important asks at the aggregate level of a single email. This is crucial for downstream processing of the framing and ask (i.e., automatic response generation). Asks are sorted based on their confidence scores, bringing those with the highest scores to the top. Those tied for first place are returned as the "top asks" for the email. For example, "PERFORM contact me (jw11@example.com)" is returned as the top ask for *Contact me. I'm around Mon. (jw11@example.com).*

5.2.3 Automated Method

Our goal is to generate an informative response to the input utterance by first generating an appropriate **Response Plan**. We train two components separately (c.f. Fig.20). In the ***Planning Phase***, we experiment with generating plans in three ways:

1. Symbolic Planner: Foremost, we need to extract plans automatically from utterances. To accomplish this goal, our symbolic planner adapts lexical representations previously used for language analysis [45] to the problem of constructing **Response Plans**. We use lexical conceptual structures and basic language processing tools [58, 134] for parsing the input, identifying the main **action**, identifying the arguments (or **targets**), and applying semantic-role labeling. Fig.19 presents ask/framing examples (type, action and target).

Once response plans are identified for all utterances in a given corpus using the symbolic planner, we need to address *automated generation* of such plans. Using the asks and framings as annotated data for a "silver" standard,[9] we train models to learn to generate "Response Plans" that are encoded with the same representation format used for asks/framings. We use the language modeling paradigm and use a large pre-trained model (GPT-2) [164] with the transformer architecture and the self-attention mechanism [208]. We fine-tune this language model with the constraint of the input utterance and the plan for this input utterance and train it to produce the plan for the response utterance. We adopt the fine-tuning approach specified by Ziegler *et al.* [242] and train two specific models (CTX and PSA) described below.

2. Context Attention Planner (CTX): based on the encoder/decoder architecture. In this model, the decoder weights are initialized with the pre-trained weights of the language model. However, a new context attention layer is added in the decoder that concatenates the conditioning information to the pre-trained weight. The conditioning information, in our case, is the plan for the input utterance.

3. Pseudo Self Attention (PSA): Proposed by Ziegler *et al.* [242], PSA injects conditioning information from the encoder directly into the pre-trained self attention (similar to the "zero-shot" model proposed by Radford *et al.* [164]).

[9] Dorr *et al.* [45] report precision of 69.2% in detecting asks/framings.

In the ***Realization Phase***, we generate responses by utilizing the response plan generated from the planning phase as well as the input utterance. We expect a more guided generation of responses that are constrained by the response plan. In this phase, we only experiment with the Pseudo Self attention (PSA) model, based on Ziegler *et al.* [242], who demonstrate that PSA outperforms other approaches on text generation tasks. We use nucleus sampling to overcome some of the drawbacks of beam search [72].

5.2.4 Corpora

Our choice of corpora is driven by the presence of information elicitation and persuasive strategies in the utterances (i.e., asks and framings).

Accordingly, we experiment with the AntiScam [116] and Persuasion for Social Good [216] corpora. **AntiScam** contains dialogues about a customer service scenario and is specifically crowdsourced to understand human elicitation strategies. **Persuasion for Social Good** corpus contains interactions between workers who are assigned the roles of persuader and persuadee, where the persuader attempts to convince the persuadee to donate to a charity.

All utterances in these corpora are first annotated through the Symbolic Planner (c.f. Section 3.2) to gauge suitability based on the presence of asks and framings. In Table13, we provide descriptive statistics of the corpora; we find an adequate number of ask/framing types (GIVE, PERFORM, GAIN, LOSE). In cases where there are no asks/framings or the symbolic planner fails to detect them, we use the default action RESPOND.

Table 13: Statistics of AntiScam and Persuasion for Social Good (PSG), with annotated asks and framings. Avg. conversation length - average number of turns in each conversation; Avg. utterance length - average length of a turn in a conversation

	AntiScam	PSG
Number of Dialogues	220	1017
Avg. Conversation Length	12.45	10.43
Avg. Utterance Length	11.13	19.36
Number of GIVE	2192	11587
Number of PERFORM	1681	7335
Number of GAIN	70	399
Number of LOSE	73	588
Number of RESPOND	4376	8078

5.2.5 Implementation

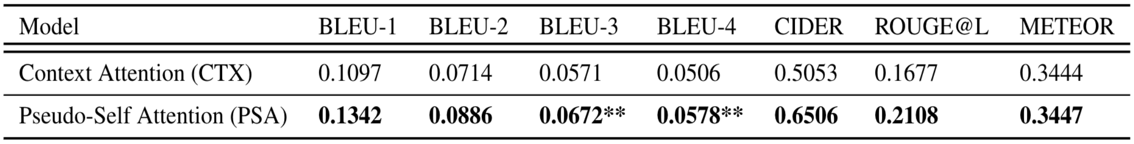
We implement the models using Open-NMT [91] and the PyTorch framework.[10] We use publicly available GPT-2 model [164] with 117M parameters, 12 layers and 12 heads in our implementations. The input utterances and the plans are tokenized using byte-pair encoding to reduce vocabulary size [184]. Both phases are trained separately. In the Planning Phase, the *plan for the input* utterance along with the input utterance is used to generate the *response plan* for the response utterance; in the Realization Phase, the response plan and input utterance are input to the model to generate the response. In both planning and realization phase, separation tokens are added (e.g. <plan>), as is common practice for transformer inputs [40, 223]. We use Adam optimizer [89] with a learning rate of 0.0005 and $\beta_1 = 0.9$ and $\beta_2 = 0.98$. During decoding, we use nucleus sampling both in the planning and realization phase. All models are trained on two TitanV GPU and take roughly 15 hours each to train

[10] `https://pytorch.org/`

the planner and realization component.

5.3 Evaluation of Approach

The results reported in these subsections were obtained by combining both corpora and dividing randomly in a ratio of 80/10/10 for the training, testing, and validation set.



Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDER	ROUGE@L	METEOR
Context Attention (CTX)	0.1097	0.0714	0.0571	0.0506	0.5053	0.1677	0.3444
Pseudo-Self Attention (PSA)	**0.1342**	**0.0886**	**0.0672****	**0.0578****	**0.6506**	**0.2108**	**0.3447**

Figure 22: Automated Metrics on performance of models in the Planning Phase. ** indicates $p < 0.01$

5.3.1 Planning Phase Evaluation

This evaluation focuses on investigating the efficacy of the two automated planners (Context Attention (CTX) and Pseudo-Self Attention (PSA)) in learning to generate response plans.

Automated Metrics

Are the automated planners able to faithfully learn how to generate the response utterance plans? To investigate, we compare the performance of the CTX and the PSA planner with the symbolic planner output (which is our silver standard reference) using common automated metrics Table 22: BLEU [157], METEOR [9], ROUGE [120], CIDEr [209] on the test set. We use the library by Sharma *et al.* [190]. We find that PSA was able to achieve higher word overlap metrics with respect to the silver standard. We conducted an in-depth analysis of the CTX and PSA planner output on the entire testing set. We found that the PSA model was more likely to produce ask actions that matched the ground truth, resulting in higher scores on the automated metrics.

Human Evaluation

Table 14: Human Evaluation results on the performance of the planner component. **Q1:** Which model plan is better suited for generating a response?; **Q2:** Which model has the more appropriate ask/framing type?; **Q3:** Which model has the more appropriate ask/framing action with respect to the type?; **Q4:** Which model has the more informative ask/framing target?

	CTX	PSA	Both	Neither
Q1	38.75%	26.25%	25%	10%
Q2	27.5%	20%	23.75%	28.75%
Q3	22.5%	17.5%	41.25%	18.75%
Q4	32.5%	31.25%	10%	26.25%

Evaluation using automated metrics provides limited evidence for the ability to automatically generate plans; we do not know if these plans are actually useful in a realization task. The question then is: *How well-suited is the automatically learned plans for the task of generating responses?*

Study 1: We asked two experts in linguistics to independently rate 40 randomly sampled plans from the test set. For context, we provided the input utterance and its plan produced by the symbolic planner. Their task was to choose which of the learned response plans was better suited to the realization task (CTX, PSA, Both, or Neither). They also evaluated the plan constituents: (**type**, **action** and **target**). We randomized the presentation order of the planner outputs across questions to avoid ordering/learning effects [140]. We find an inter-rater agreement [193] of 0.5 ($p < 0.001$) between the linguists.

Table 15: Human evaluation results comparing CTX and PSA planner separately to the Symbolic Planner

	CTX	PSA	Symbolic Planner	Both
Quality	30%	X	35%	35%
	X	35%	22%	43%

Table 14 shows the results from Study 1. From **Q1**, we find that the CTX planner is better suited to generate an appropriate response over the PSA planner. Similarly, through **Q2**, **Q3**, and **Q4**, we find that the CTX planner is better able to generate the appropriate ask/framing types, actions, and targets. We also find that the linguists rated Neither plan was suited to generate a response 10% of the time. Put differently; the automatically generated plans would work 90% of the time to generate an appropriate utterance in the realization phase. The learned plans have trouble associating an appropriate ask/framing type and target (28.75% and 26.75%) but perform better with the ask/framing action (18.75% Neither rating).

This evaluation compares the automatic planners against one another, but *how well do the planners compare to the silver standard (symbolic planner)?*

Study 2: We asked the same linguistic experts to independently determine which amongst two plans (symbolic vs. each automated planner) would be more appropriate to generate a response. This study design is consistent with prior studies in dialogue evaluation [142]. Table 15 presents the results from Study 2.

We find that experts prefer the plans produced by the symbolic planner over the CTX output but not over the PSA planner output. Inter-annotator agreement [193] between the experts for this study was 0.54. While Study 1 compared CTX and PSA

Table 16: Automated metric results on the responses generated on the test set of both corpora.

Realizer Input	Dataset	BLEU	Diversity	Length	BERT-score
No Plan	AntiScam	0.0658	**0.0067**	7.168	0.841
	PSG	0.1149	**0.0049**	13.713	0.845
Symbolic Planner	AntiScam	**0.1814**	0.0062	6.245	**0.844**
	PSG	**0.1992**	0.0038	11.982	**0.848**
Context Attention Planner	AntiScam	0.0705	0.0064	7.298	0.84
	PSG	0.1027	0.0043	14.088	0.847
Pseudo Self Attention Planner	AntiScam	0.0692	0.0065	**7.553**	0.838
	PSG	0.1253	0.0045	**15.128**	0.847

planner outputs against one another, Study 2 compared CTX and PSA outputs against the silver standard. As we observe from the automated metrics (Table 22), PSA model plans are more faithful to the ground truth, e.g., higher BLEU 1-4 scores than CTX model plans. Since PSA planner outputs are more faithful to the ground truth, this may be why human judges rate them as preferable more often when compared against ground truth.

Planning Phase Evaluation Findings: To summarize this evaluation section, we find: PSA outperforms the CTX planner on automated metrics. This finding is consistent with the results from Ziegler *et al.* [242]. From Study 1, we find that both the planners can generate appropriate plans, with the appropriate ask/framing type, action, and target for the realization phase, a large proportion of the time. From Study 2, we find that when compared to the silver standard plans, PSA planner output is preferred over the CTX planner.

5.3.2 Realization Phase Evaluation

While the previous section focuses on evaluating the ability to generate plans automatically, we do not yet know *whether separating the generation process into planning and realization produces better responses than an end-to-end system?*

Thus, we compare four approaches towards realizing a response given an input utterance (through the Pseudo-Self Attention fine-tuned realization algorithm): (1) **No Planner** model which receives input utterance but no plan as input; (2) **Symbolic Planner based Generation**: This model receives the plan from symbolic planner output; (3) **CTX Planner-Based Generation**: This model receives the CTX plan; (4) **PSA Planner-Based Generation**: This model receives the PSA plan.

Automated Metrics

Prior research has shown that most automated metrics have little to no correlation to human ratings on NLG tasks [123, 180]; however, they may provide some standard of reference to evaluate performance. We report the following metrics: (i) BLEU [157] (ii) length of responses, with the understanding that models that can generate longer responses are better (iii) following, Mei *et al* [142], we report the diversity metric [110]. Diversity is calculated as the number of distinct unigrams in the generation scaled by the total number of generated tokens [142, 112]. (iv) BERT-Score [237] metric, an embedding-based score that has shown greater correlation to human ratings.

Table 16 reports on the automated evaluation against the ground truth utterances. We find that on both corpora and across all metrics except Diversity, incorporating plans as an additional input to the realization phase helps achieve a higher score than having No Planner. From Table 16, we find that the realizer without any plans can achieve higher diversity, but the difference is not statistically significant.

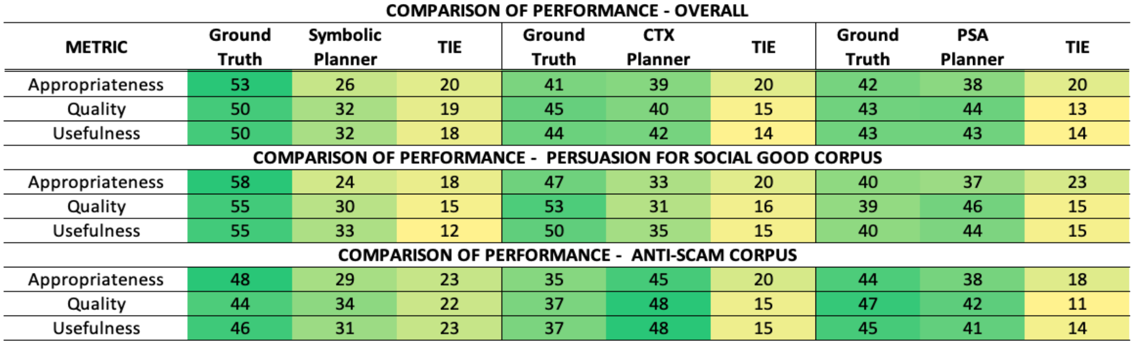
Human Evaluation

Since automated metrics are not the most informative indicators of the quality of generated responses, thorough human evaluation is necessary. We *investigate if humans prefer the responses generated by the planner-based models over those generated without the plan (No Planner).* We conducted two human evaluation studies by recruiting workers from Amazon Mechanical Turk service with strict quality control criteria: workers should have at least 90% HIT approval rate and at least 1000 approved HITs. In each survey, workers are asked to evaluate responses on these metrics, following Novikova *et al.* [155]: (i) *Appropriateness:* determines whether the response aligns with the topic of the conversation and the input utterance. (ii) *Quality:* determines the overall quality in terms of grammatical correctness, fluency, and adequacy (iii) *Usefulness:* determines if the response is highly informative to generate a response.

Table 17: Average ranking of realized output from four different planners, lower score is better

Realizer Input	Appropriateness	Quality	Usefulness
No Plan	2.54	2.61	2.58
Symbolic Planner	2.51	2.5	2.53
CTX Planner	**2.34**	**2.38**	**2.38**
PSA Planner	2.59	2.5	2.51

Study 1: We tasked 30 crowd-sourced workers to rank order the four model



METRIC	Ground Truth	Symbolic Planner	TIE	Ground Truth	CTX Planner	TIE	Ground Truth	PSA Planner	TIE
COMPARISON OF PERFORMANCE - OVERALL									
Appropriateness	53	26	20	41	39	20	42	38	20
Quality	50	32	19	45	40	15	43	44	13
Usefulness	50	32	18	44	42	14	43	43	14
COMPARISON OF PERFORMANCE - PERSUASION FOR SOCIAL GOOD CORPUS									
Appropriateness	58	24	18	47	33	20	40	37	23
Quality	55	30	15	53	31	16	39	46	15
Usefulness	55	33	12	50	35	15	40	44	15
COMPARISON OF PERFORMANCE - ANTI-SCAM CORPUS									
Appropriateness	48	29	23	35	45	20	44	38	18
Quality	44	34	22	37	48	15	47	42	11
Usefulness	46	31	23	37	48	15	45	41	14

Figure 23: Comparison of ground truth reference with realized output from each model that receives learned plans as input: Symbolic, CTX or PSA. Higher values (shown as %) darker color represent better performance.

responses from best to worst. We randomly sampled 60 examples from the test set with an even 50% split (30 examples each) between the Persuasion for Social Good and AntiScam corpora. We chose the best to worst ranking mechanism since it has shown greater consistency and agreement amongst workers on tasks related to dialogue evaluation over other evaluation designs (e.g. Likert scales) [178, 90]. The presentation order of model outputs for each question was again randomized to avoid learning effects [140]. Table 17 demonstrates the average rank position (1=Best, 4=Worst) obtained by each model. We find using the plans generated by the CTX planner helps generate better responses. On the metrics of quality and usefulness, we find that incorporating planning as additional input performs better than no plan (i.e. end-to-end system).

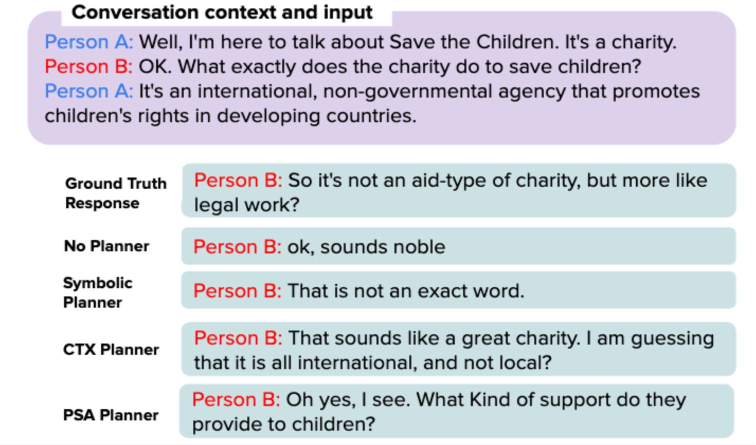
Study 2: In this study, we evaluate *how well the generated responses compare to the ground truth.* The ground truth references are those produced by humans in the PSG and Anti-Scam corpora. We recruited 11 MTurk workers with the same crowdsourcing quality controls as Study 1. For the same randomly sampled 60 examples from Study

1, workers were asked if they prefer the ground-truth response, the response generated from the three planners, or both, on the three chosen metrics. This study design is also consistent with prior work [142]. Workers were blinded to the source of the response (ground truth or generated) and were presented the responses in a randomized order across all questions to avoid ordering effects.

Fig.23 shows the results (higher value/darker color is better): we find that responses generated from the symbolic planner as input do not perform well when compared to the ground truth. In other words, the proportion of time that the ground truth response is preferred over that generated by the symbolic planner is significant (e.g. 53% vs. 26% on the Appropriateness metric overall).

We find that on all three metrics, the responses generated using CTX and PSA plans help generate responses that are comparable to the responses produced by humans (ground truth). We also find that the PSA planner-based responses perform better overall and on the Persuasion for Social Good corpus. Surprisingly, the CTX planner-based responses perform better than Ground Truth utterances for the Anti-Scam corpus (45%, 48%, and 48% of the time preferable vs. ground truth response 35%, 37% and 37% on the three metrics, Appropriateness, Quality, and Usefulness, resp.). We explain this unexpected finding in the next sub-section (Section 4.3).

Realization Phase Evaluation Findings: To summarize this evaluation subsection, we find that the Symbolic Planner-realized output outperforms the CTX, PSA, and No Planner output on the automated metrics of BLEU and BERT-score. Importantly, the CTX planner-realized output has a higher rank in terms of overall preference in human evaluation than the other models (c.f. Table17). We also find



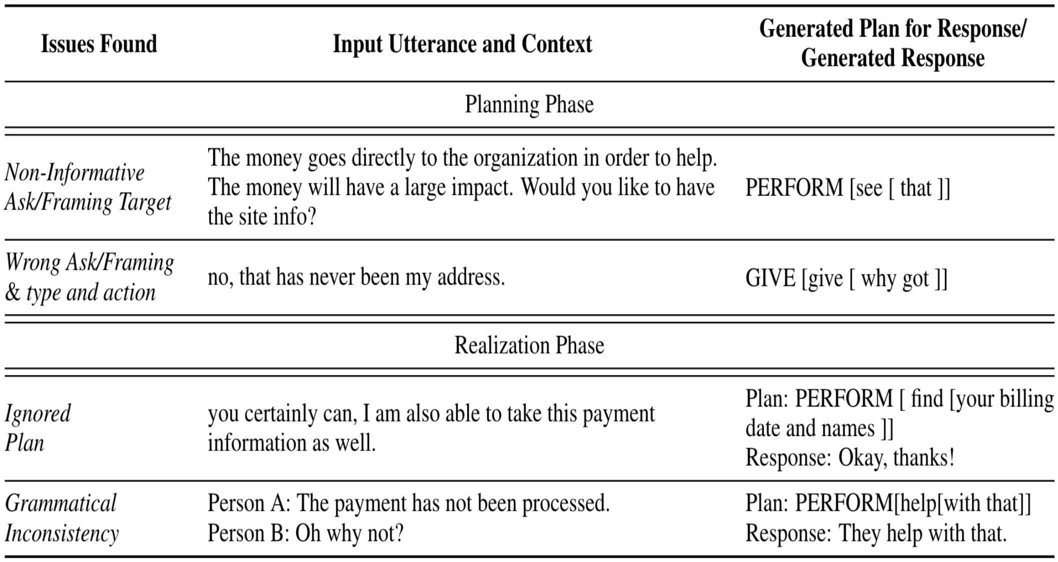
Conversation context and input

Person A: Well, I'm here to talk about Save the Children. It's a charity.
Person B: OK. What exactly does the charity do to save children?
Person A: It's an international, non-governmental agency that promotes children's rights in developing countries.

Ground Truth Response	Person B: So it's not an aid-type of charity, but more like legal work?
No Planner	Person B: ok, sounds noble
Symbolic Planner	Person B: That is not an exact word.
CTX Planner	Person B: That sounds like a great charity. I am guessing that it is all international, and not local?
PSA Planner	Person B: Oh yes, I see. What Kind of support do they provide to children?

Figure 24: Sample outputs from realization phase with all variations of planner input, as well as the ground truth response from the corpus

that human-generated utterances (ground truth) are preferred overall (c.f Fig.23) than the model outputs. We found inter-rater consistency and agreement scores to be >0.6 on average across the metrics (full tables are reported in the Appendix).



Issues Found	Input Utterance and Context	Generated Plan for Response/ Generated Response
	Planning Phase	
Non-Informative Ask/Framing Target	The money goes directly to the organization in order to help. The money will have a large impact. Would you like to have the site info?	PERFORM [see [that]]
Wrong Ask/Framing & type and action	no, that has never been my address.	GIVE [give [why got]]
	Realization Phase	
Ignored Plan	you certainly can, I am also able to take this payment information as well.	Plan: PERFORM [find [your billing date and names]] Response: Okay, thanks!
Grammatical Inconsistency	Person A: The payment has not been processed. Person B: Oh why not?	Plan: PERFORM[help[with that]] Response: They help with that.

Figure 25: Summary table of issues found through a qualitative analysis of the planning and realization phase outputs

5.3.3 Qualitative Analysis

We conduct a qualitative evaluation of the outputs and present several cherry- and lemon-picked examples here. Additional examples of success and failure cases are uploaded in the Appendix. In the sample conversation shown in Figure24, we find that realized outputs using CTX and PSA plans are more consistent with the context of conversation than the symbolic planner approach. Additionally, the No Planner output (an end-to-end system that does not get a plan as an additional input) produces an utterance that may not necessarily continue the conversation further.

This example is also illustrative of the finding in Study 2 of the Planning Phase evaluation, where the crowdsourced workers rated the automated planner-based outputs better than the symbolic planner-based outputs (c.f. Fig.25). This might seem contradictory, as the CTX and PSA planners are trained on the silver standard data from the symbolic planner. We contend that this is due to the ability of automated planners (CTX and PSA) to generalize, an ability lacking in the symbolic planner. In such cases, as shown in Fig.24, the symbolic planner defaults to the RESPOND message plan, and this leads to generated output: *That is not an exact word*, which is generic and off-topic. The symbolic planner could be improved to cover more cases; however, the effort would not be scalable.

While we find promising results for the automatically-generated planners in Sections 4.1 and 4.2, areas of improvement do exist (Table25):

Non-Informative Ask/Framing Targets: We find several examples where the ask/framing targets are non-informative words (e.g. *this, that*). Non-informative targets can cause the downstream realization process to generate an utterance that is, in turn, also non-informative. One example of such a case is shown in Row 1 of Table25.

Wrong Type and Action: Another planning phase issue category is that the constituents of plan representation (e.g., the ask/framing type and action) can be incorrect. As illustrated by the example in Table25, an ask target of *why got* is incorrect. Typically, we would expect to find a noun or a noun phrase as the ask/framing action (e.g., *your billing date and names* as shown in the plan in Row 3).

Ignored Plan: In the Realization phase, a typical issue is that the realizer may ignore the generated plan. As can be seen in Row 3 of Table25, the plan should constrain the response, and thus should contain phrases such as *finding your billing date and names.* However, the generated response is instead a generic phrase *Okay, thanks!.*

Grammatical inconsistencies: We also note that there were cases where the grammar, e.g. pronoun usage, is inconsistent. For the example shown in Row 4 of Table25, we see that the generated response is *They help with that.* whereas the conversation is between two persons; a generated response of *I can help with that* would be more consistent with the context of the conversation.

5.4 Discussion

We address the task of natural language generation in open-ended dialogue systems. We test our hypothesis that decoupling the generation process into planning and realization can achieve better performance than an end-to-end approach.

In the planning phase, we explore three methods to generate response plans, including a Symbolic Planner and two learned planners, the Context Attention and Pseudo Self Attention models. Through linguist expert evaluation, we are able to determine the efficacy of the response plans towards realization. *In the realization phase*, we use the Pseudo Self Attention model to make use of the learned response plans to generate responses.

Our key finding through two separate human crowdsourced studies is that decoupling realization, and planning phases outperforms an end-to-end No Planner system across three metrics (Appropriateness, Quality, and Usefulness).

In this work, we have taken an initial step towards the goal of replicating human language generation processes. Thorough and rigorous evaluations are required to fully support our claims, e.g., by including additional metrics and more diverse corpora. In this work, we limit the types to GIVE, GAIN, LOSE, and PERFORM. However, we do not restrict the ask action and target at all. Also, since our symbolic planner can be used to obtain silver standard training data, straightforward changes like adding additional lexicons would enable us to generalize to other corpora as well as include additional ask types in our pipeline. Another natural extension would be to explore training the planning and realization phases together in a hierarchical process [57]. This would, in principle, further validate the efficacy of our approach.

Acknowledgments

This work was supported by DARPA through AFRL Contract FA8650-18-C-7881 and through Army Contract W31P4Q-17-C-0066. All statements of fact, opinion, or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of DARPA, AFRL, Army, or the U.S. Government.

CHAPTER 6: IMPACT OF EXPERIMENT DESIGN IN EVALUATION OF DIALOGUE SYSTEMS

6.1 Introduction

A tremendous amount of recent research has focused on approaches towards generating responses for conversations in an open-domain setting [164, 226, 223]. An equally challenging task for natural language generation systems is evaluating the quality of the generated responses. Evaluation of generated output is typically conducted using a combination of crowdsourced human judgments and automated metrics adopted from machine translation and text summarization [123, 153]. However, studies conducted by Liu *et al.*[123] and Novikova *et al.* [153] show that the automated metrics have poor correlation with human judgments. Despite their shortcomings, automated metrics like BLEU, ROUGE, and METEOR are used due to a lack of alternative metrics. This puts a major imperative on obtaining high-quality crowdsourced human judgments. Previous research which employs crowdsourced judgments has focused on metrics including *ease of answering*, *information flow* and *coherence* [114, 50], *naturalness* [4], *interestingness* [6, 179], *fluency* or *readability* [236], *engagement* [210]. While experiment designs primarily use Likert scales, Belz and Kow [17] argue that discrete scales, such as the Likert scales, can be unintuitive and certain individuals may avoid extreme values in their judgments. Prior research has also shown that use of continuous scales is more viable for language evaluation [154, 18]. Such evidence places more emphasis on a careful study towards obtaining reliable and consistent human ratings for dialogue evaluation.

To address this research problem, we perform a two-part human study that focuses on a systematic comparison of four experimental conditions by incorporating ***continuous, relative*** and ***ranking scales*** faining crowdsourced human judgments. In this initial study, we evaluate the use of two metrics: ***Readability*** and ***Coherence***. In the study, we investigate the effects of cognitive biases, specifically anchoring bias, on decision-making around evaluating chatbot output, we designed a $2X2$ experiment with 77 crowdsourced workers. We studied how anchors (both numerical and textual) and the presentation order of rating tasks affect the consistency of human judgments.

Our key findings are:

1. Use of Likert scales results in the lowest inter-rater consistency and agreement when compared to other experiment conditions

2. Use of continuous scales results in higher inter-rater consistency and agreement

3. Raters who have no prior experience in evaluating dialogue system output have greater inter-rater consistency and agreement than do those who have previously participated in such rating tasks.

4. We find systematic effects of anchoring in the **magnitude** of participants' ratings: participants who are presented with an anchor will provide a rating that is closer to the anchor value than those who are not presented with an anchor.

5. We find systematic effects of anchoring in the **consistency** of participants'

ratings: participants who are presented with an anchor will be (generally) more consistent in their ratings than those who are not presented with an anchor.

6. We find that interpretation of metrics affects consistency: participants were more consistent with their ratings on Readability than in their ratings on Coherence, potentially because the interpretation of Coherence is more subjective than Readability.

Our findings have the potential to help the research community in the design of their evaluation tasks to obtain higher quality human judgments for natural language generation output.

6.2 Data and Models

To obtain ratings on conversational agent output, we trained three models from scratch to generate responses. Code for these models was made available by Dziri *et al.* [50] (`https://github.com/nouhadziri/THRED`). We first describe the corpus we used to train the models.

Corpus

We used the Reddit Conversational Corpus made available by Dziri *et al.* [50]. This corpus consists of conversations obtained from 95 different subreddits, curated out of 1.1M subreddits. The date range is 20 months from November 2016 until August 2018. Table 18 shows overall descriptive statistics of the corpus, where the average length of utterances is consistent across the Training, Validation, and Test sets.

Table 18: Descriptive statistics of the corpus used in our experiments.

	Train	**Valid.**	**Test**
Dialogues	9.2M	500K	400K
Avg. Length of Utterances	13.98	13.98	13.99

6.2.1 Models

All three models used in our experiments are based on *seq2seq* approaches that contain an encoder and decoder component. *Seq2seq* approaches are commonly used in language generation tasks, such as machine translation and dialogue generation. For dialogue generation, the encoder receives the input sequence $X = x_1, x_2, ..., x_n$ as input. Each input sequence is passed through an LSTM [69] on the encoder side which produces a hidden state representation (Eq 17.)

$$h_t^{enc} = f(h_{t-1}^{enc}, x_t). \tag{17}$$

where h_{t-1}^{enc} represents the previous hidden state and f represents a non-linear activation function. The decoder uses the last hidden state of the encoder as the initial state and output tokens are conditioned on the input (Eq 18.) where y_{t-1} represents the ground truth input into the decoder.

$$s_t^{dec} = f(s_{t-1}^{dec}, y_{t-1}) \tag{18}$$

1. **Seq2Seq:** Our first model is a traditional *seq2seq* model with attention mechanism. We use the attention mechanism proposed by Bahdanau *et al.* [7]. Attention assists the decoder to attend to different parts of the input while generating the response. The decoder produces a context vector c_t at each time

step by attending to the encoder hidden state h_t^{enc} along with the last hidden of the decoder s_{t-1} (represented through Eq 19.) where α represents the relative importance on the input side. The output from the model y_t is produced through a softmax function (Eq 20.).

$$\begin{gathered} c_i = \sum_{i=1}^{n} \alpha_i h_i^{enc} \\ \alpha_i = \frac{exp(e_i)}{\sum_{j=1}^{n} exp(e_j)} \\ e_i = f(s_{t-1}, h_i) \end{gathered} \tag{19}$$

$$y_t = softmax(y_{t-1}, s_t, c_t) \tag{20}$$

2. **HRED:** Our second model uses ***Hierarchical Encoder-Decoder*** [187] architecture. This model is an advancement over traditional *seq2seq models*. HRED overcomes the bottlenecks of traditional *seq2seq* models by capturing longer context from dialogue histories. HRED model introduces a two-level hierarchy to capture long term context. The first layer is called the utterance layer that captures the meaning of each sentence, similar to traditional seq2seq models. It further encodes the hidden states of the utterance layer to the inter-utterance layers that capture the context and input information [205].

3. **THRED:** Our last model is the ***Topic Augmented Hierarchical Encoder-Decoder*** [50]. This model uses topic words along with a hierarchical encoder-decoder to produce a response. The topic words were obtained using a pre-trained LDA model [71]. This model also makes use of the attention mechanism on the context along with the topic words from the input sequence.

6.3 Metrics

For this initial study, we focus on two metrics, readability and coherence. These metrics are among those essential to evaluate the quality of generated responses [153, 49]. We describe an automated method to compute each metric.

Readability or Fluency measures the linguistic quality of text and helps quantify the difficulty of understanding the text for a reader [61, 153]. We use the Flesch Reading Ease (FRE) [88] that counts the number of words, syllables, and sentences in the text.[11] Higher readability scores indicate that utterance is easier to read and comprehend.

Coherence measures the ability of the dialogue system to produce responses consistent with the topic of conversation [210]. To calculate coherence, we use the method proposed by Dziri *et al.* [50]. This metric computes the cosine similarity on embedding vectors of generated response and target while accounting for dull and generic responses through a penalty factor.

To overcome the issue of dull and generic responses, Dziri *et al.* [50] induce a penalty factor which takes into account

$$P = 1 + \log \frac{2 + L'}{2 + L''} \tag{21}$$

where L' indicates the length of response after dropping stop words and punctuation and L'' indicates the length of non-dull parts of the response after dropping stop words.

[11] `https://bit.ly/1IZOFG4`

The penalized semantic similarity (SS) score is then calculated as:

$$SS(utt_{i,j}, resp_i) = P \times (1 - cos(utt_{i,j}, resp_i) \qquad (22)$$

where i represents the index of the dialogue in the dataset and j denotes index of the utterance in the conversation history.

6.4 Experiment Designs

6.4.1 Study 1

In study1, we use three well-known question types of Likert Scale, Magnitude Estimation, and Best-Worst Ranking. We chose these question types to investigate as these are commonly used across various language evaluation tasks [18, 4, 154, 90]. With the help of these three types of questions, we design four rating procedures that are explained below.

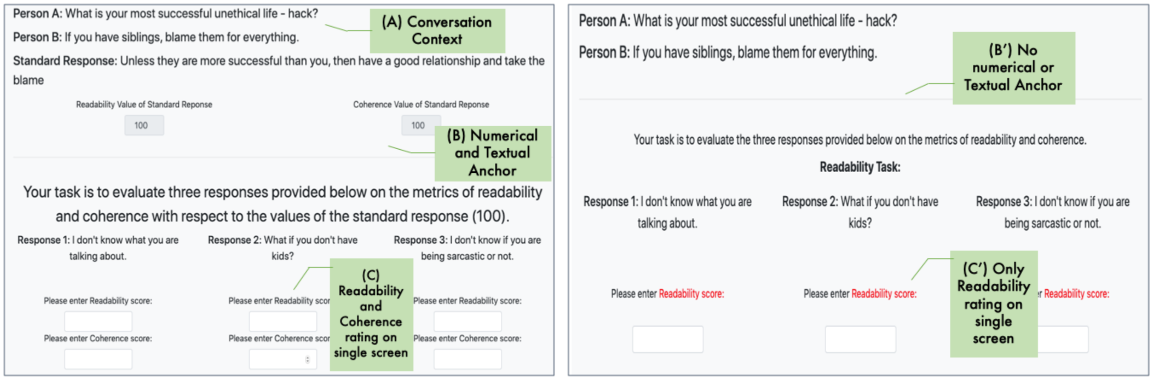
Likert Scale (LS): is typically used in experiments for crowdsourcing human evaluation of dialogue systems [4, 127]. In our experiment, we ask the raters to rate the generated responses on a 6-point scale, following Novikova *et al.* [154] (where 1 is the lowest and 6 is the highest on the metrics of readability and coherence).

Rank-Based Magnitude Estimation (RME): Prior research by Belz and Kow [18] demonstrates through six separate experiments that continuous scales are more viable and offer distinct advantages over discrete scales in evaluation tasks. Recently, Novikova *et al.* [154] adopted magnitude estimation by providing the rater with a *standard value* for a reference sentence to evaluate output from goal-oriented systems. Following Novikova *et al.* [154], we also set the value of the standard (reference utterance) as 100 since the reference utterance was produced by humans and is considered as gold-standard. The crowd-sourced workers are asked to provide a score relative to 100 (from 0 to 999) for three system-generated outputs.

Biased Magnitude Estimation (BME): Our third experiment design is biased magnitude estimation (BME). The main difference between RME and BME method is that the standard value we provide for the reference utterance is not uniformly set to 100 for all examples, but instead calculated by automated methods (explained in Section 6.3). Our motivation to do so is to understand if **anchoring bias** may affect the ratings when judgments are made relative to a fixed value (100) or relative to a value calculated by automated means. Anchoring bias is the tendency to rely too heavily on one piece of information offered (the “anchor”, in this case, the number 100) when making decisions [81].

Best-Worst Scaling (BWS): Our last experiment condition is best-worst scaling (BWS) in which raters are asked to rank the generated responses in order of best to worst on both metrics (readability and coherence). This approach has previously been used to estimate emotion intensity and has been demonstrated to produce high quality and consistent judgments from humans [90].

Each task includes 50 randomly sampled conversations from the test set in our corpus along with generated responses from the three models and the ground truth (reference utterance). For each task, we collected ratings from 40 workers with Master qualifications through Amazon Mechanical Turk.



Person A: What is your most successful unethical life - hack?

Person B: If you have siblings, blame them for everything.

Standard Response: Unless they are more successful than you, then have a good relationship and take the blame

Readability Value of Standard Reponse: 100

Coherence Value of Standard Reponse: 100

Your task is to evaluate three responses provided below on the metrics of readability and coherence with respect to the values of the standard response (100).

Response 1: I don't know what you are talking about.

Response 2: What if you don't have kids?

Response 3: I don't know if you are being sarcastic or not.

Please enter Readability score:

Please enter Coherence score:

Please enter Readability scor

Please enter Coherence scor

Please enter Readability score:

Please enter Coherence score:

Person A: What is your most successful unethical life - hack?

Person B: If you have siblings, blame them for everything.

Your task is to evaluate the three responses provided below on the metrics of readability and coherence.

Readability Task:

Response 1: I don't know what you are talking about.

Response 2: What if you don't have kids?

Response 3: I don't know if you are being sarcastic or not.

Please enter Readability score:

Please enter Readability score:

r Readability score:

(A) Conversation Context
(B) Numerical and Textual Anchor
(C) Readability and Coherence rating on single screen
(B') No numerical or Textual Anchor
(C') Only Readability rating on single screen

Figure 26: Sample screen showing variations in the experiment conditions. (A) represents the conversational context that is shown across all conditions. (B) is the numerical and textual anchor presented to participants in anchoring conditions. (B') shows the screenshot of conditions where no anchor is presented. (C) is used in Setup 1 where both questions of readability and coherence ratings are shown together. (C') is used in Setup 2 where the readability and coherence are treated as individual tasks and only one is shown at a time to the participant.

6.4.2 Study 2

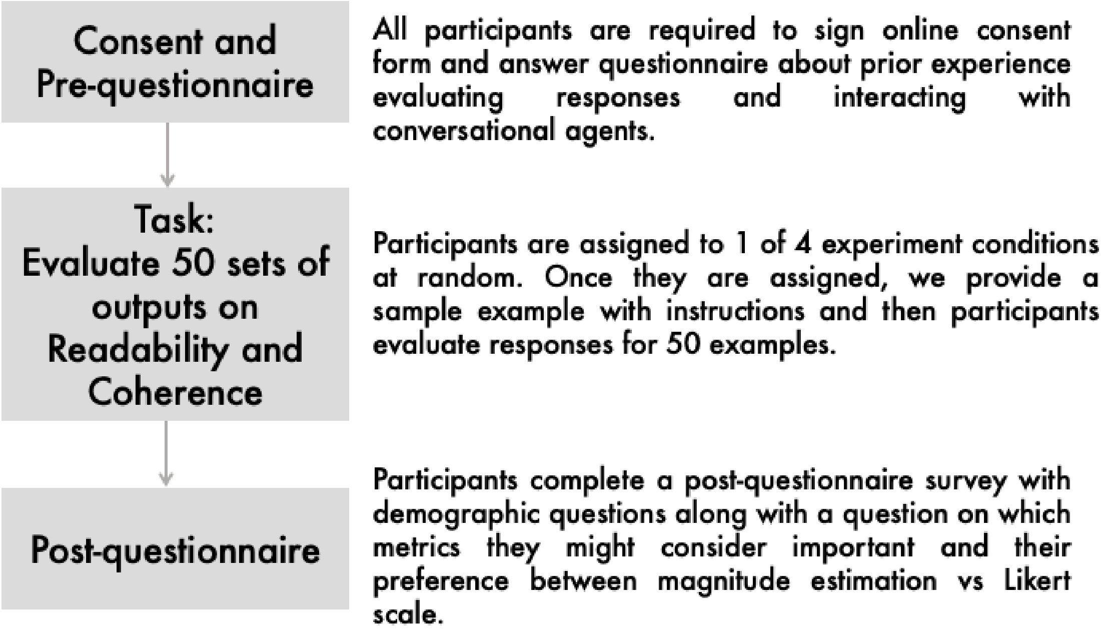
To study the impact of cognitive biases, we design four experiment conditions, namely **Anchor**: With or Without Anchor and **Presentation Order**: Both Questions or Single Question (on a single screen). Table 19 shows the four different experimental conditions in our experiment design, while Figure 26 shows two sample screenshots from the study interface.

Table 19: $2X2$ experiment design with four experiment conditions and number of participants across each condition

	No Anchor	**Anchor**
Both Questions (Setup 1)	18	22
Single Question (Setup 2)	18	19

As shown in Figure 26, participants across all experiment conditions are shown the Conversation Context (A). Participants in the Anchor conditions are shown the

Standard Response and the Readability and Coherence value of the Standard Response (set to 100 in this study, following prior work is done by [154]); together these form the Numerical and Textual Anchor (B) (Figure 26-left). Participants in the No Anchor condition are shown neither the Standard Response nor the Readability and Coherence value of the standard response (B') (Figure 26-right). Participants in the Both Questions (Setup 1) condition are asked to input their ratings of Readability and Coherence on a single screen (C) (as shown in Figure 26-left). Participants in the Single Question condition (Setup 2) are asked to input their ratings on a single metric on a single screen (as shown Figure 26-right (C') for Readability), and then input their ratings on the Coherence metric on the next screen when they click the next button (not shown).


Consent and Pre-questionnaire
Task: Evaluate 50 sets of outputs on Readability and Coherence
Post-questionnaire

All participants are required to sign online consent form and answer questionnaire about prior experience evaluating responses and interacting with conversational agents.

Participants are assigned to 1 of 4 experiment conditions at random. Once they are assigned, we provide a sample example with instructions and then participants evaluate responses for 50 examples.

Participants complete a post-questionnaire survey with demographic questions along with a question on which metrics they might consider important and their preference between magnitude estimation vs Likert scale.

Figure 27: The experiment flow for each crowd-sourced worker taking part in this study.

Figure 27 provides the flow of steps taken by workers in the experiment, beginning with the informed consent procedure and pre-questionnaire, followed by the task of evaluating 50 sets of outputs on two metrics of Readability and Coherence and ending with the post-questionnaire. In the pre-questionnaire, we asked two questions about the prior experience of workers: (Q1) *Have you taken part in previous studies involve evaluating conversational responses?* and (Q2) *Have you taken part in previous studies that involve talking to a chatbot?* Our motivation behind asking these questions is to understand if prior experience participating in similar studies affects inter-rater consistency. In the post-questionnaire, we obtain participant demographics including their age, gender, race, and education. We also ask them if they find it preferable to provide ratings as magnitude estimation questions or on Likert scales. Also, we obtain their free-form responses on which metrics they would consider important for evaluating conversational agent output. These post-questionnaire questions are designed to obtain qualitative data to better inform our future studies.

6.5 Experiment Results

We organize our findings along with five main research questions (RQs) outlined in this section. In the following section, we report on statistical significance using two-way ANOVAs on the between-subject ratings across the four experiment conditions (Tables 20– 26).

RQ1: What is the effect of experiment design on the reliability on human ratings? We use intra-class correlation (ICC) to measure the reliability across multiple raters [193, 102]. To compare the scores obtained from magnitude estimation experiments to the ratings from the task using discrete Likert scales, we perform a

Table 20: ICC scores on the metrics of readability and coherence for each experiment design. All values are statistically significant p-value<0.001 except those indicated by †. n=40 for all four designs.

		Likert	**RME**	**BME**	**BWS**
ICC-C	Readability	0.75	0.95†	0.83	0.75
	Coherence	0.83	0.92	0.81	0.80
ICC-A	Readability	0.59	0.95†	0.83	0.75
	Coherence	0.77	0.92	0.81	0.80

normalization of the magnitude estimation scores on a logarithmic scale as suggested by Bard *et al.* [11].

Table 20 represents the ICC scores on consistency (ICC-C) and agreement (ICC-A) for our four experiment tasks. We observe that the use of Magnitude Estimation with anchors (RME or BME) results in more reliable ratings than using the Likert Scale or using Best-Worst ranking (BWS). This result is consistent with prior research by Novikova *et al.* [154] and Belz and Kow [18].

Further, we find that a possible explanation for magnitude estimation to achieve high ICC scores might be due to anchoring bias. Figure 28 provides the mean and bootstrapped confidence interval (95%) of the responses across the experiment conditions. In Setup 1, we find that participants with no anchor produce ratings ($M = 58.92$) that are significantly lower than ratings provided by participants in anchor condition ($M = 72.94$). We find a similar pattern across Setup 2 with no anchor, resulting in a mean rating of 61.25, while ratings in anchor condition responses have a mean of 69.02. We analyze the ratings on Readability and Coherence separately (Figure 29): the presence of numerical and textual anchors results in higher (on average) ratings than the absence of the anchor (statistically significant with p<0.001).

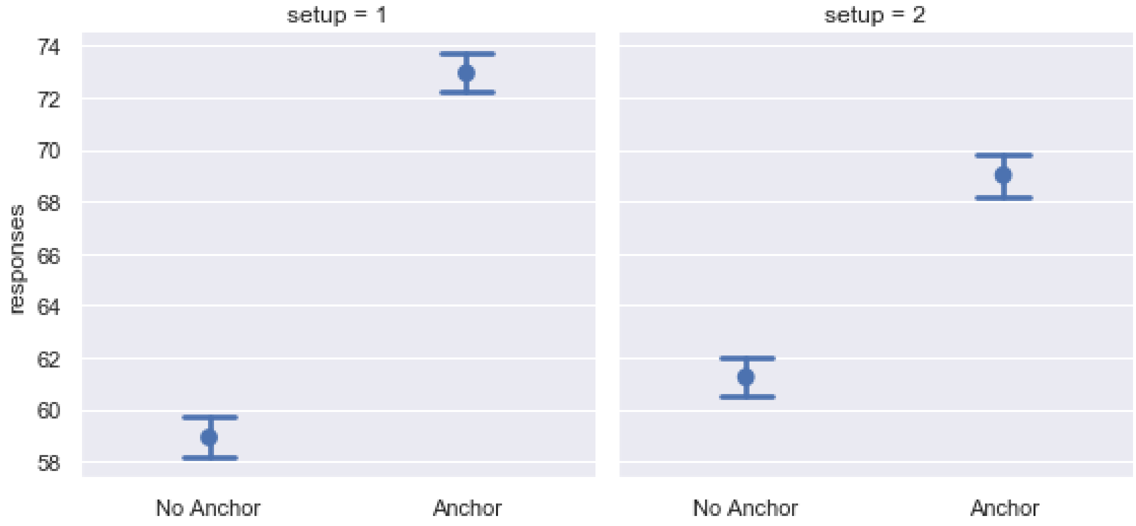


Figure 28: Mean of the responses bootstrapped with 95% confidence intervals across setups 1 and setup 2

Figure 29 presents ratings for the metrics of readability and coherence separately. We find that across both setups, the difference between the anchor and no anchor conditions to be larger for the metrics of readability than coherence (statistically significant with p<0.001). We find that in Setup 1, readability values have a mean of 83.13 in the anchor condition, and in no anchor condition the mean of the responses drop down to 64.97. Also in Setup 1, we find that for coherence metric, the mean of responses in the anchoring condition is M=62.74 and without anchor M=52.89. We find similar trends in the responses provided in Setup 2 for both metrics of readability and coherence.

RQ2: Does time taken to complete the survey influence reliability of the rankings? To analyze RQ2, we calculated the total time spent by each participant from the start to the end of the experiment. We found that the BME task had the

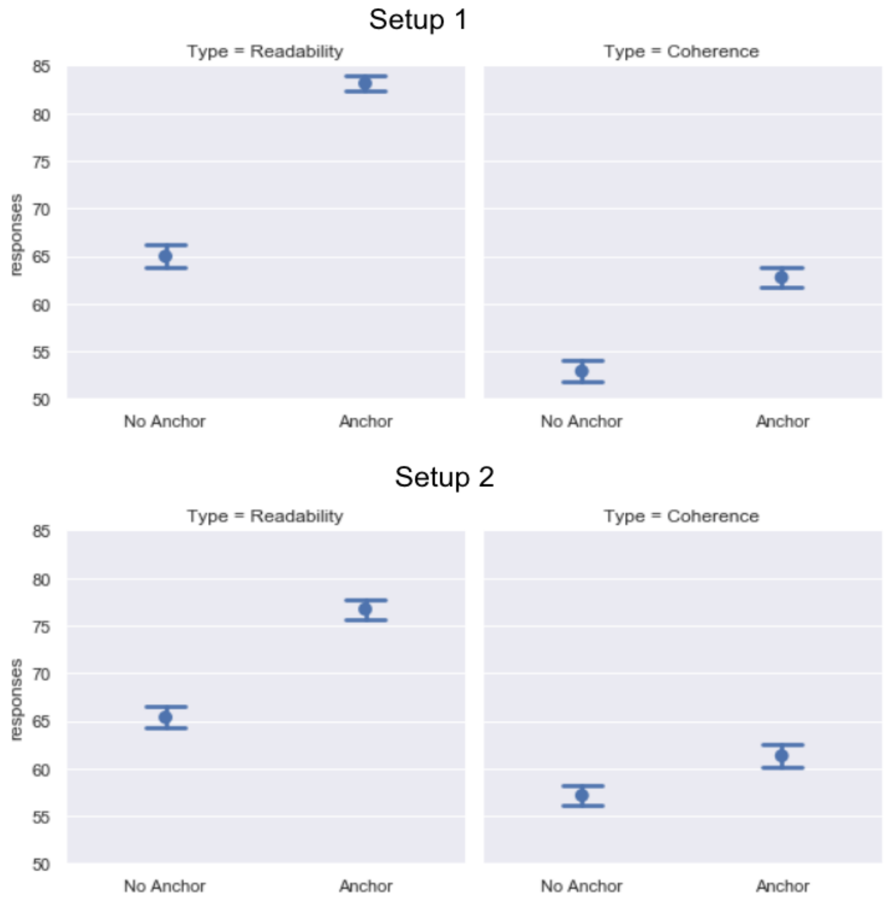


Figure 29: Mean of the responses bootstrapped with 95% confidence intervals across Setups 1 and 2 on the metrics of Readability and Coherence.

longest average time to completion (43 minutes), followed by RME (42.8 minutes) and Likert scale (33 minutes; Best-Worst ranking had the shortest average completion time (32.5 minutes). We then test the hypothesis that raters who spent longer than average time on the task would be more reliable in their ratings than those who completed in less than average time. Table 21 represents the ICC scores for raters who spent higher than average time for the task, while Table 22 represents scores for raters who spent less than average time. Surprisingly, we find that consistency and agreement among raters who spend less than average time is higher than those who spend more time, for the Likert, BME, or BWS experiment designs. When using the RME design, raters who spend more time have higher consistency and agreement.

Table 21: ICC scores when participants spend **above average time**. All values in this table are statistically significant with p-value<0.001

		Likert (n=15)	**RME** (n=16)	**BME** (n=15)	**BWS** (n=16)
ICC-C	Readability	0.58	0.93	0.51	0.62
	Coherence	0.74	0.85	0.55	0.64
ICC-A	Readability	0.52	0.93	0.51	0.62
	Coherence	0.69	0.86	0.56	0.64

Table 22: ICC scores when participants spend **below average time**. All values in this table are statistically significant with p-value<0.001

		Likert (n=25)	**RME** (n=24)	**BME** (n=25)	**BWS** (n=24)
ICC-C	Readability	0.61	0.88	0.81	0.65
	Coherence	0.66	0.85	0.75	0.76
ICC-A	Readability	0.36	0.88	0.81	0.66
	Coherence	0.55	0.85	0.75	0.76

From Study 2, we find that In Setup 1, in the above-average group, the mean of responses in no anchor condition was 39.65 and the mean of the responses in anchor condition was 72.35. We find similar evidence in Setup 2 with people in anchor condition provide higher values (83) close to the numerical anchor (100). (see 30)

RQ3: Does the prior experience of evaluating dialogue system output or engaging with conversational agents affect the reliability of rankings? We asked each rater two additional questions at the end of the task. The questions asked raters to indicate whether or not they had prior experience taking part in studies (a) to evaluate dialogue system output, and (b) to engage with a conversational agent.

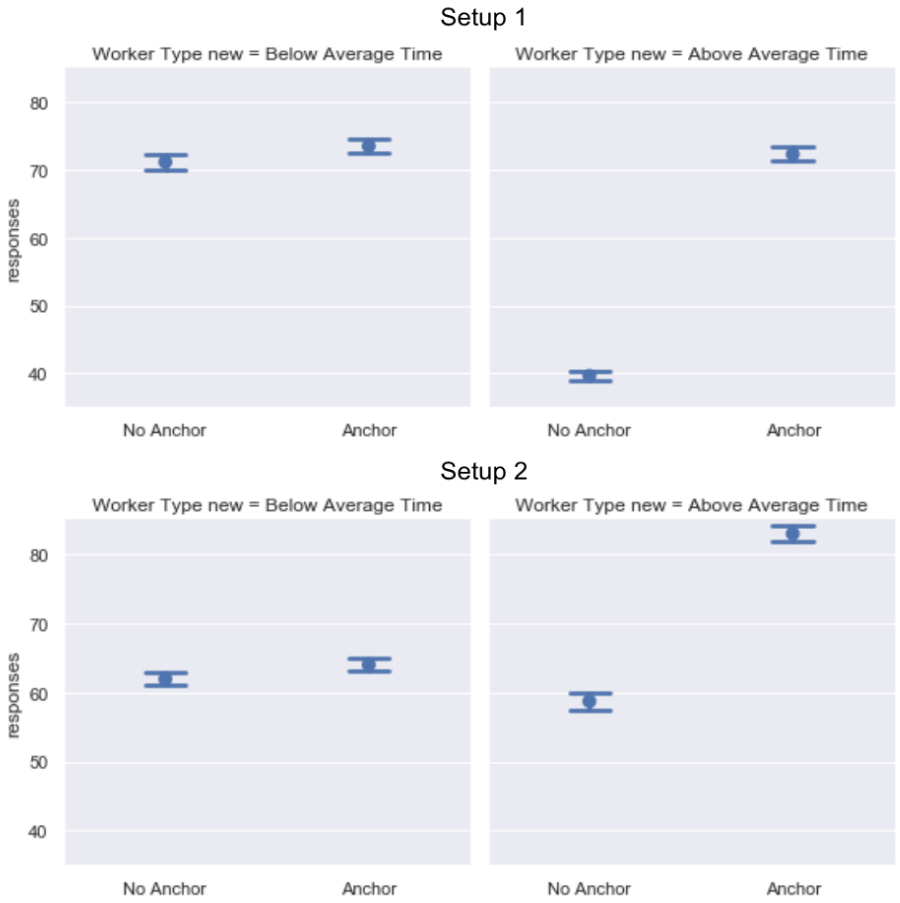


Figure 30: Mean of the responses bootstrapped with 95% confidence intervals across Setups 1 and 2 based on amount of time spent on study.

Tables 23 and 24 show how reliable the ratings from the participants based on their prior experience of taking part in studies about evaluating conversational response. We find that participants who have not taken part in prior studies are more consistent and have a higher agreement score than a participant who has prior experience. These results are also validated by Tables 25 and 26 which shows that participants with no prior experience of engaging with conversational agents are more consistent and reliable.

Figure 31 demonstrates the impact of the prior experience of evaluating conversational responses (Question 1 on the pre-questionnaire) on the magnitude of ratings.

Table 23: ICC scores when participants **have** prior experience evaluating dialogue system output. All values statistically significant at p-value<0.001.

		Likert	**RME**	**BME**	**BWS**
		(n=15)	(n=7)	(n=18)	(n=13)
ICC-C	Readability	0.45	0.37	0.51	0.54
	Coherence	0.38	0.48	0.55	0.63
ICC-A	Readability	0.35	0.38	0.52	0.55
	Coherence	0.32	0.49	0.55	0.63

Table 24: ICC scores when participants **do not have** prior experience evaluating dialogue system output. All values statistically significant at p-value<0.001 except those indicated by †.

		Likert	**RME**	**BME**	**BWS**
		(n=25)	(n=33)	(n=22)	(n=27)
ICC-C	Readability	0.71	0.95†	0.83	0.70
	Coherence	0.82	0.92	0.76	0.72
ICC-A	Readability	0.50	0.95†	0.83	0.70
	Coherence	0.75	0.92	0.77	0.72

We find contrasting responses across both setups. In Setup 1, we find that people with prior experience in the anchor condition produce higher responses (M=74.41) close to the numerical anchor (100) and no anchor condition produces lower values (M=38.36) whilst people with no prior experience are similar in their responses across both conditions. In comparison to Setup 1, we find that in Setup 2 participants with no prior experience produce higher responses in the anchor condition (M=71.45) and no anchor condition (M=63.74).

Figure 32 shows the impact of prior experience of interacting with chatbots. Participants who have such prior experience demonstrated signs of anchoring. We find that mean of responses (M=80.40) for participants with prior experience in the anchor con-

Table 25: ICC scores when participants **have** prior experience engaging with conversational agents. All values statistically significant at p-value<0.001.

		Likert	**RME**	**BME**	**BWS**
		(n=18)	(n=11)	(n=23)	(n=18)
ICC-C	Readability	0.46	0.69	0.60	0.57
	Coherence	0.44	0.65	0.62	0.67
ICC-A	Readability	0.37	0.69	0.61	0.57
	Coherence	0.38	0.65	0.62	0.67

Table 26: ICC scores when participants **do not have** prior experience engaging with conversational agents. All values statistically significant at p-value<0.001 except those indicated by †.

		Likert	**RME**	**BME**	**BWS**
		(n=22)	(n=29)	(n=17)	(n=22)
ICC-C	Readability	0.70	0.95†	0.84	0.67
	Coherence	0.82	0.91	0.76	0.68
ICC-A	Readability	0.48	0.95†	0.84	0.67
	Coherence	0.75	0.91	0.76	0.68

dition to be significantly higher ($p < 0.001$) than participants in no anchor condition (M=48.01) in Setup 1.

When comparing against Setup 1, we find that people in Setup 2 with no prior experience produce higher responses (M=70.74) in the anchoring condition than in the no anchor condition (M=63.12).

These findings substantiate the hypothesis that people with prior experience (answered Yes on Questions 1 and 2) would be more susceptible to the anchoring effect than those who do not have prior experience with similar tasks, **however** this effect is only seen in Setup 1, while Setup 2 demonstrates the opposite effect. We find this evidence to be particularly interesting and plan to further investigate the potential

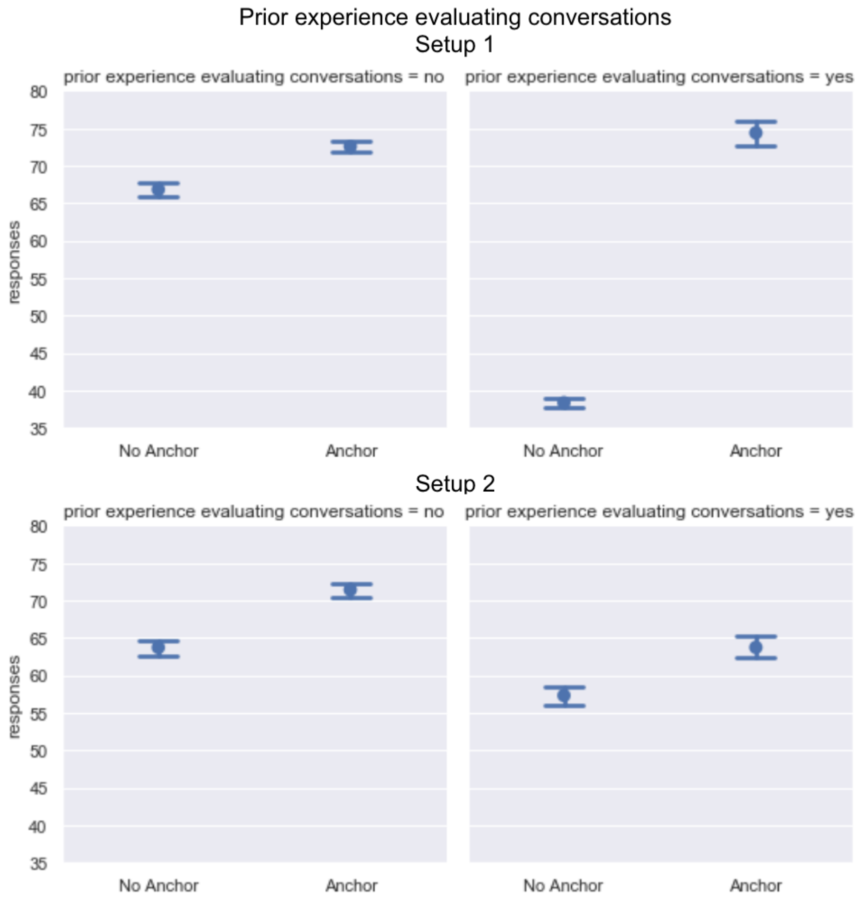


Figure 31: Mean of the responses bootstrapped with 95% confidence intervals across setups 1 and setup 2 based on prior experience of being involved studies about evaluating conversations.

of eliciting ratings on different metrics as separate tasks (Setup 2) as a means of mitigating the anchoring bias effect.

RQ4: How well do automated methods to calculate readability and coherence correlate with human ratings? We report on the correlation between readability and coherence scores that are calculated using automated methods (outlined in Section 6.3) with the human ratings in Table 27. Readability scores were computed using the Flesh Reading Ease [88] and coherence scores were computed based on the method proposed by Dziri *et al.* [50]. We observe that the automated

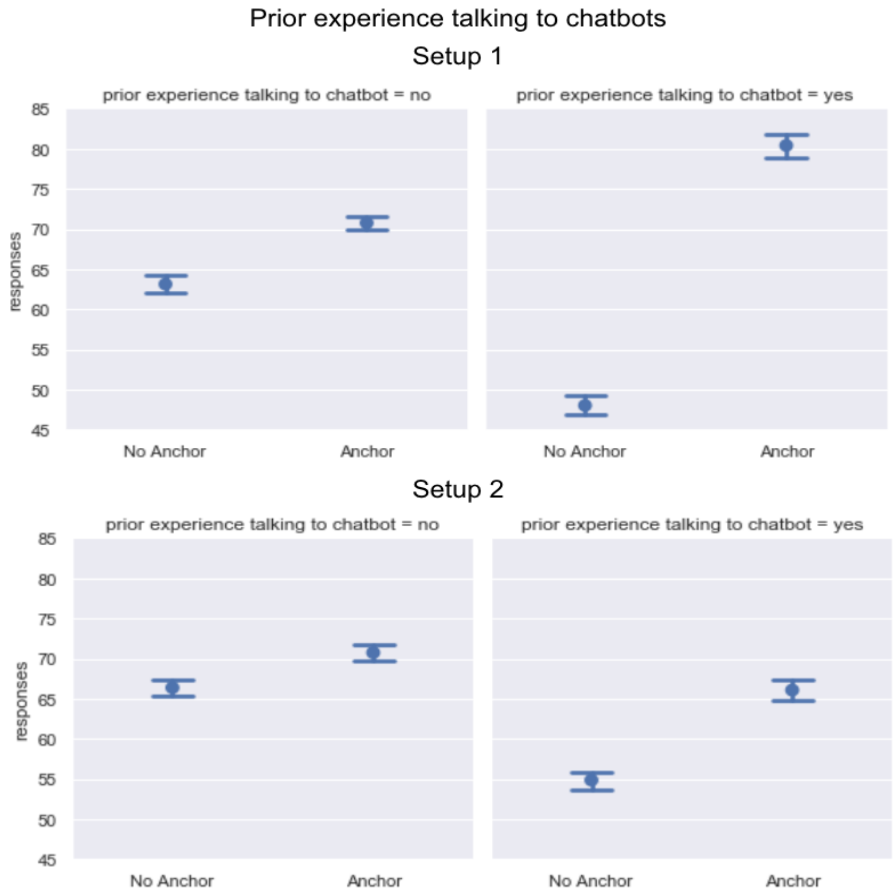


Figure 32: Mean of the responses bootstrapped with 95% confidence intervals across setups 1 and setup 2 based on prior experience of being involved studies about talking to chatbot.

metrics for Readability [88] and Semantic Similarity [50] show low correlation to human judgments ratings.

Table 27: Spearman correlation between the ratings obtained from the automated metrics to human ratings.

	Likert	**RME**	**BME**	**BWS**
	Automated Metric			
Readability	0.26	-0.11	-0.12	-0.06
Coherence	-0.12	-0.13	-0.11	0.01

RQ5: Is there any correlation between ratings of readability and coher-

ence for each of the four experiment conditions? To evaluate whether there is any correlation between the ratings obtained for readability and coherence through of four experimental designs, we report the Spearman correlation values in Table 28. We find that there is a high correlation between the human ratings of readability and coherence obtained through RME and BME (statistically significant). One likely factor affecting correlation may be anchoring bias towards the fixed value of the standard utterance provided in RME (100) and reference value provided in BME. We aim to investigate this further in future work.

Table 28: Spearman correlation between the ratings of readability and coherence obtained on four different experiment designs. *** p-value<0.001

	Likert	**RME**	**BME**	**BWS**
		Readability		
Coherence	0.1	0.79***	0.77***	0.5***

6.6 Discussion

In this chapter, we present our work on designing a systematic experiment with four experiment conditions to evaluate the output of dialogue systems. Different from prior work where a similar study was conducted with output from goal-oriented systems [154], our study focuses on evaluating output in open-domain situations. Consistent with prior findings, metrics calculated using automated methods [49] were found to have a negative correlation with human judgments (c.f. Table 27). This finding points to the need for more effective automated metrics.

We find that the use of continuous scales to obtain crowdsourced ratings provides more consistent and reliable ratings than ratings obtained through Likert scales or

Best-Worst scaling. This finding is consistent with prior work conducted by Novikova *et al.* [154]. Novel in our study was the testing of the Best-Worst scaling method to evaluate responses against one another. Although the Best-Worst scaling method is effective in obtaining crowdsourced ratings of emotions [90], we did not find it to be effective in this study. We aim to investigate further whether this finding can be reproduced in a different experiment.

Further, we were able to identify the effects of time taken to complete the task on rating reliability. We find that workers who spent less than average time on the task had higher consistency (for the Likert, BME, and BWS experiment conditions) than did the workers who spent more than average time. This finding is counter-intuitive, we expect that spending more time would positively impact inter-rater consistency. Our first step in the analysis of the effects of time taken on reliability included analyzing data from workers who spent more or less than average time, which offers admittedly a limited perspective; an interesting next step would be to more thoroughly study the effects of time taken on reliability by taking into account the full distribution of the time spent data.

We also find that *lack of* prior experience of evaluating open-domain dialogue system output results in more reliable ratings. One potential explanation for this could be that workers may have pre-conceived notions based on their experience. One limitation of our current study is that although we had output from three separate models, we conducted the study using data from one corpus. Reproducing our findings across additional corpora, additional metrics and other experiment designs would help substantiate these findings further. An analysis of the interaction effects between independent variables such as time is taken and prior experience would also help strengthen the findings of our study.

By using a larger sample size (n=40), we are able to make claims about statistical significance across experimental conditions. In future work, we plan to evaluate the impact of cognitive biases such as anchoring and confirmation bias in-depth and how it affects consistency and reliability along with testing continuous scale ratings with no reference value.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

With the development of large-scale transformer models, the field of conversational AI has also made significant progress by having the capability to produce grammatical and fluent generations. However, even with these advancements state of the art models suffers from context and factual inconsistencies along with the issue of hallucination. These issues prevent conversational AI from achieving its true potential and being deployed in products. To address the issue of contextual inconsistencies, we have proposed planning with a generation framework to decouple the end-to-end frameworks. First, through the planning component that understands the intent and reasoning behind a message and can be combined with a dialog manager to strategize the next response plan to make it effective and increase engagement. The response plan generated by the planner component is used by the generation component to produce a more constrained response. We found that the proposed decoupling approach **performs better than an end-to-end approach by producing a more on-topic and constrained response to the context of the conversation**.

However, this proposed framework makes use of the last turn of conversation without making use of the rest of the conversation history. To make better use of conversation history, cognitive architectures provide structures of memory that can be used to augment existing frameworks to provide more accurate response plans. Memory plays a crucial role in human cognition that allows us to encode information to be retrieved later for other tasks. Similarly, in conversational AI systems, memory can be used to store conversational history that can be used as needed to produce better responses. As a preliminary approach, we proposed a cognitive architecture adopted framework called the “CMA Model” that had a declarative long-term memory and working memory working in conjunction with an encoder-decoder framework. We proposed a new type of action selection mechanism that operates between declarative long-term memory and working memory to help enhance conversational agents. In our work, **we showed that our action selection mechanism outperforms the current state-of-the-art approach in identifying salient contextual utterances from the dialog history**. In the future, I want to adopt the memory structure of cognitive architectures to transformer-based models to help navigate issues surrounding contextual consistency.

Further, memory structures can play an important role in trying to reduce hallucinations and factual inconsistencies in responses produced by conversational agents that use these transformer models. Transformer models implicitly store the knowledge in the weights that often comprise billions of parameters. Even with the stored implicit knowledge, these models produce statements that are incorrect by mixing up facts about a particular entity [194]. To make progress in the area of factual consistency, integrating long-term memory (declarative) with transformer models could help reduce hallucinations. More recently, in the area of open domain question answering, new approaches such as RAG[108], DPR[82], REALM[68] have been proposed that are capable of retrieving relevant knowledge from large knowledge bases. These retrieval mechanisms provide an opportunity to use the entire conversation history as the query to retrieve multiple pieces of relevant knowledge that can be operated upon as needed and stored in the declarative long-term memory using our action selection bridge approach. I studied the effectiveness of these neural retrievers and combining them with the transformer models to study their effect on hallucination and factual consistency. We found that: **1. neural retrievers have a significant impact on the model's ability to produce factually valid and accurate statements; 2. Delayed beam search produces more factually accurate responses from the models compared to nucleus sampling and beam search; 3. factual inconsistent statements can be alleviated using a fact-consistency detector.** In the future, I plan to study approaches that can be combined with memory architectures to make better use of context in conjunction with retrieved knowledge. Further directions of research include designing better retrieval approaches that are capable of reasoning over multiple pages or documents.

Apart from the focus on architectures that build better conversational state-of-the-art systems, there also needs to have a focus on evaluating the performance of these models. Prior research has shown that evaluation of NLG systems including conversational AI systems is a pretty difficult task and the currently available automated metrics don't do a good job at comprehensively analyzing the performance of the system. The alternative strategy to evaluate an NLG system is to perform a human evaluation. However, human evaluation of NLG systems is riddled with issues such as (1) Human ratings suffer from consistency and reliability; (2) Over the last twenty years, multiple definitions and multiple terms have been introduced for the same task [77]. In our work, we tackled the issue of consistency and reliability of human ratings by studying the impact of different experimental designs and the impact of external factors such as prior experience and time taken. We found that **Magnitude Estimation design achieves the best consistency and reliability when compared to Likert Scale and Ranking approaches. However, further careful examination showed that a possible factor for Magnitude Estimation to outperform other experiment designs might be due to the presence of anchoring bias, a form of cognitive bias**. In the future, I plan to work further on studying human evaluation procedures and understanding human decision-making when it comes to evaluation. Moving in this direction, allows the community to gather insights into the evaluation process and these insights can be used to develop new automated metrics that might show better correlation to the human ratings.

REFERENCES

[1] G. Angeli, P. Liang, and D. Klein. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512. Association for Computational Linguistics, 2010.

[2] G. Angeli, C. D. Manning, and D. Jurafsky. Parsing time: Learning to interpret time expressions. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 446–455. Association for Computational Linguistics, 2012.

[3] N. Asghar, P. Poupart, J. Hoey, X. Jiang, and L. Mou. Affective neural response generation. *CoRR*, abs/1709.03968, 2017.

[4] N. Asghar, P. Poupart, J. Hoey, X. Jiang, and L. Mou. Affective neural response generation. In *European Conference on Information Retrieval*, pages 154–166. Springer, 2018.

[5] N. Asghar, P. Poupart, X. Jiang, and H. Li. Deep active learning for dialogue generation. *arXiv preprint arXiv:1612.03929*, 2016.

[6] N. Asghar, P. Poupart, X. Jiang, and H. Li. Deep active learning for dialogue generation. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 78–83, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics.

[7] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[8] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv e-prints*, abs/1409.0473, Sept. 2014.

[9] 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*, volume 29, pages 65–72, 2005.

[10] S. Bangalore and O. Rambow. Corpus-based lexical choice in natural language generation. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 464–471, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[11] E. G. Bard, D. Robertson, and A. Sorace. Magnitude estimation of linguistic acceptability. *Language*, pages 32–68, 1996.

[12] R. Barzilay and M. Lapata. Collective content selection for concept-to-text generation. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 331–338. Association for Computational Linguistics, 2005.

[13] R. Barzilay and M. Lapata. Aggregation via set partitioning for natural language generation. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 359–366. Association for Computational Linguistics, 2006.

[14] R. Barzilay and L. Lee. Catching the drift: Probabilistic content models, with applications to generation and summarization. *arXiv preprint cs/0405039*, 2004.

[15] J. A. Bateman. Enabling technology for multilingual natural language generation: the kpml development environment. *Natural Language Engineering*, 3(1):15–55, 1997.

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

[17] A. Belz and E. Kow. Comparing rating scales and preference judgements in language evaluation. In *Proceedings of the 6th International Natural Language Generation Conference*, pages 7–15. Association for Computational Linguistics, 2010.

[18] A. Belz and E. Kow. Discrete vs. continuous rating scales for language evaluation in nlp. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 230–235. Association for Computational Linguistics, 2011.

[19] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[20] P. Bhatia, M. Gavalda, and A. Einolghozati. soc2seq: Social embedding meets conversation model. *arXiv preprint arXiv:1702.05512*, 2017.

[21] K. Bock. Language production: Methods and methodologies. *Psychonomic Bulletin & Review*, 3(4):395–421, 1996.

[22] K. Bock, G. S. Dell, F. Chang, and K. H. Onishi. Persistent structural priming from language comprehension to language production. *Cognition*, 104(3):437–458, 2007.

[23] A. Boyd, R. Puri, M. Shoeybi, M. Patwary, and B. Catanzaro. Large scale multi-actor generative dialog modeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 66–84, Online, July 2020. Association for Computational Linguistics.

[24] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[25] A. Cahill, M. Forst, and C. Rohrer. Stochastic realisation ranking for a free word order language. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 17–24. Association for Computational Linguistics, 2007.

[26] L. Cahill, C. Doran, R. Evans, C. Mellish, D. Paiva, M. Reape, D. Scott, and N. Tipper. In search of a reference architecture for nlg systems. In *Proceedings of the 7th European Workshop on Natural Language Generation*, pages 77–85, 1999.

[27] K. Cao and S. Clark. Latent variable dialogue models and their diversity. *arXiv preprint arXiv:1702.05962*, 2017.

[28] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

[29] H. Chen, Z. Ren, J. Tang, Y. E. Zhao, and D. Yin. Hierarchical variational memory network for dialogue generation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1653–1662. International World Wide Web Conferences Steering Committee, 2018.

[30] H. Cheng and C. Mellish. Capturing the interaction between aggregation and text planning in two generation systems. In *Proceedings of the first international conference on Natural language generation-Volume 14*, pages 186–193. Association for Computational Linguistics, 2000.

[31] 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)*, pages 1724–1734. Association for Computational Linguistics, 2014.

[32] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.

[33] N. J. Cohen and L. R. Squire. Preserved learning and retention of pattern-analyzing skill in amnesia: Dissociation of knowing how and knowing that. *Science*, 210(4466):207–210, 1980.

[34] P. R. Cohen and C. R. Perrault. Elements of a plan-based theory of speech acts. *Cognitive science*, 3(3):177–212, 1979.

[35] T. S. Curl and P. Drew. Contingency and action: A comparison of two forms of requesting. *Research on language and social interaction*, 41(2):129–153, 2008.

[36] R. Dale and E. Reiter. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive science*, 19(2):233–263, 1995.

[37] H. Dalianis. Aggregation in natural language generation. *Computational Intelligence*, 15(4):384–414, 1999.

[38] G. S. Dell. Positive feedback in hierarchical connectionist models: Applications to language production 1. *Cognitive Science*, 9(1):3–23, 1985.

[39] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. {BERT}: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[41] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[42] A. Dimitromanolaki and I. Androutsopoulos. Learning to order facts for discourse planning in natural language generation. *arXiv preprint cs/0306062*, 2003.

[43] E. Dinan, S. Roller, K. Shuster, A. Fan, M. Auli, and J. Weston. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*, 2018.

[44] B. J. Dorr. Machine translation divergences: A formal description and proposed solution. *Computational Linguistics*, 20(4):597–633, 1994.

[45] B. J. Dorr, A. Bhatia, A. Dalton, B. Mather, B. Hebenstreit, S. Santhanam, Z. Cheng, S. Shaikh, A. Zemel, and T. Strzalkowski. Detecting asks in se attacks: Impact of linguistic and structural knowledge. *arXiv preprint arXiv:2002.10931*, 2020.

[46] P. A. Duboue and K. R. McKeown. Statistical acquisition of content selection rules for natural language generation. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 121–128. Association for Computational Linguistics, 2003.

[47] O. Dušek and F. Jurčíček. A context-aware natural language generator for dialogue systems. *arXiv preprint arXiv:1608.07076*, 2016.

[48] O. Dušek and Z. Kasner. Evaluating semantic accuracy of data-to-text generation with natural language inference. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 131–137, Dublin, Ireland, Dec. 2020. Association for Computational Linguistics.

[49] N. Dziri, E. Kamalloo, K. Mathewson, and O. Zaiane. Evaluating coherence in dialogue systems using entailment. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3806–3812, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[50] N. Dziri, E. Kamalloo, K. W. Mathewson, and O. Zaiane. Augmenting neural response generation with context-aware topical attention. *arXiv preprint arXiv:1811.01063*, 2018.

[51] M. Elhadad and J. Robin. An overview of surge: A reusable comprehensive syntactic realization component. In *Eighth International Natural Language Generation Workshop (Posters and Demonstrations)*, 1996.

[52] J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[53] N. Engonopoulos and A. Koller. Generating effective referring expressions using charts. In *INLG*, pages 6–15, 2014.

[54] T. W. Epperson and A. Zemel. Reports, requests, and recipient design: The management of patron queries in online reference chats. *Journal of the American Society for Information Science and Technology*, 59(14):2268–2283, 2008.

[55] R. Evans, P. Piwek, and L. Cahill. What is nlg? Association for Computational Linguistics, 2002.

[56] A. Fan, C. Gardent, C. Braud, and A. Bordes. Augmenting transformers with knn-based composite memory for dialogue. *arXiv preprint arXiv:2004.12744*, 2020.

[57] A. Fan, M. Lewis, and Y. Dauphin. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.

[58] M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. F. Liu, M. Peters, M. Schmitz, and L. S. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. 2017.

[59] K. Garoufi. Planning-based models of natural language generation. *Language and Linguistics Compass*, 8(1):1–10, 2014.

[60] A. Gatt and E. Krahmer. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *arXiv preprint arXiv:1703.09902*, 2017.

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

[62] M. Ghazvininejad, C. Brockett, M.-W. Chang, B. Dolan, J. Gao, W.-t. Yih, and M. Galley. A knowledge-grounded neural conversation model. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[63] S. Ghosh, M. Chollet, E. Laksana, L.-P. Morency, and S. Scherer. Affect-lm: A neural language model for customizable affective text generation. *arXiv preprint arXiv:1704.06851*, 2017.

[64] E. Goldberg, N. Driedger, and R. I. Kittredge. Using natural-language processing to produce weather forecasts. *IEEE Intelligent Systems*, (2):45–53, 1994.

[65] Y. Goldberg. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420, 2016.

[66] S. Golovanov, R. Kurbanov, S. Nikolenko, K. Truskovskyi, A. Tselousov, and T. Wolf. Large-scale transfer learning for natural language generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6053–6058, 2019.

[67] R. Goyal, M. Dymetman, E. Gaussier, and U. LIG. Natural language generation through character-based rnns with finite-state prior knowledge. In *COLING*, pages 1083–1092, 2016.

[68] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang. REALM: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*, 2020.

[69] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[70] J. Hockenmaier and M. Steedman. Ccgbank: a corpus of ccg derivations and dependency structures extracted from the penn treebank. *Computational Linguistics*, 33(3):355–396, 2007.

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

[72] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020.

[73] A. Holtzman, J. Buys, M. Forbes, A. Bosselut, D. Golub, and Y. Choi. Learning to write with cooperative discriminators. *arXiv preprint arXiv:1805.06087*, 2018.

[74] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spaCy: Industrial-strength Natural Language Processing in Python. 2020.

[75] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

[76] H. Horacek. Generating referential descriptions under conditions of uncertainty. In *Proceedings of the 10th European Workshop on Natural Language Generation (ENLG)*, pages 58–67, 2005.

[77] D. M. Howcroft, A. Belz, M.-A. Clinciu, D. Gkatzia, S. A. Hasan, S. Mahamood, S. Mille, E. van Miltenburg, S. Santhanam, and V. Rieser. Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 169–182, Dublin, Ireland, Dec. 2020. Association for Computational Linguistics.

[78] C. Huang, O. Zaiane, A. Trabelsi, and N. Dziri. Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 49–54, 2018.

[79] M. Huang, X. Zhu, and J. Gao. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32, 2020.

[80] M. Jordan. Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986. Technical report, California Univ., San Diego, La Jolla (USA). Inst. for Cognitive Science, 1986.

[81] D. Kahneman. 36 heuristics and biases. *Scientists Making a Difference: One Hundred Eminent Behavioral and Brain Scientists Talk about Their Most Important Contributions*, page 171, 2016.

[82] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, Nov. 2020. Association for Computational Linguistics.

[83] P. Ke, J. Guan, M. Huang, and X. Zhu. Generating informative responses with controlled sentence function. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1499–1508, 2018.

[84] C. Kennedy and L. McNally. Scale structure, degree modification, and the semantics of gradable predicates. *Language*, pages 345–381, 2005.

[85] S. M. Kennison. *Psychology of Language: Theory and Applications*. Macmillan International Higher Education, 2018.

[86] I. H. Khan, K. Van Deemter, and G. Ritchie. Generation of referring expressions: Managing structural ambiguities. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 433–440. Association for Computational Linguistics, 2008.

[87] J. Kim and R. J. Mooney. Generative alignment and semantic parsing for learning from ambiguous supervision. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 543–551. Association for Computational Linguistics, 2010.

[88] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, and B. S. Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. 1975.

[89] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[90] S. Kiritchenko and S. Mohammad. Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 465–470, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[91] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*, 2017.

[92] R. Kondadadi, B. Howald, and F. Schilder. A statistical nlg framework for aggregated planning and realization. In *ACL (1)*, pages 1406–1415, 2013.

[93] I. Konstas and M. Lapata. Unsupervised concept-to-text generation with hypergraphs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 752–761. Association for Computational Linguistics, 2012.

[94] I. Kotseruba and J. K. Tsotsos. A review of 40 years of cognitive architecture research: Core cognitive abilities and practical applications. *arXiv preprint arXiv:1610.08602*, 2016.

[95] S. Kottur, X. Wang, and V. Carvalho. Exploring personalized neural conversational models. In *IJCAI*, pages 3728–3734, 2017.

[96] E. Krahmer and K. Van Deemter. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218, 2012.

[97] W. Kryscinski, B. McCann, C. Xiong, and R. Socher. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online, Nov. 2020. Association for Computational Linguistics.

[98] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher. Ask me anything: Dynamic memory networks for natural language processing. In *International Conference on Machine Learning*, pages 1378–1387, 2016.

[99] J. E. Laird. *The Soar Cognitive Architecture.* The MIT Press, 2012.

[100] J. E. Laird, C. Lebiere, and P. S. Rosenbloom. A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *Ai Magazine*, 38(4), 2017.

[101] J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: An architecture for general intelligence. *Artificial intelligence*, 33(1):1–64, 1987.

[102] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[103] I. Langkilde. Forest-based statistical sentence generation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 170–177. Association for Computational Linguistics, 2000.

[104] I. Langkilde-Geary and K. Knight. Halogen statistical sentence generator. In *Proceedings of the ACL-02 Demonstrations Session*, pages 102–103, 2002.

[105] P. Langley, J. E. Laird, and S. Rogers. Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, 10(2):141–160, 2009.

[106] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[107] M. Lewis, D. Yarats, Y. N. Dauphin, D. Parikh, and D. Batra. Deal or no deal? end-to-end learning for negotiation dialogues. *arXiv preprint arXiv:1706.05125*, 2017.

[108] P. S. H. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401, 2020.

[109] 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.

[110] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119. Association for Computational Linguistics, 2016.

[111] J. Li, M. Galley, C. Brockett, G. 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)*, pages 994–1003, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.

[112] J. Li, M. Galley, C. Brockett, G. P. Spithourakis, J. Gao, and B. Dolan. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*, 2016.

[113] 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.

[114] J. Li, W. Monroe, A. Ritter, D. Jurafsky, M. Galley, and J. Gao. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202. Association for Computational Linguistics, 2016.

[115] J. Li, W. Monroe, T. Shi, A. Ritter, and D. Jurafsky. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*, 2017.

[116] Y. Li, K. Qian, W. Shi, and Z. Yu. End-to-end trainable non-collaborative dialog system. *arXiv preprint arXiv:1911.10742*, 2019.

[117] Z. Li, C. Niu, F. Meng, Y. Feng, Q. Li, and J. Zhou. Incremental transformer with deliberation decoder for document grounded conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 12–21, Florence, Italy, July 2019. Association for Computational Linguistics.

[118] P. Liang, M. I. Jordan, and D. Klein. Learning semantic correspondences with less supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 91–99. Association for Computational Linguistics, 2009.

[119] R. Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.

[120] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

[121] Z. C. Lipton, J. Berkowitz, and C. Elkan. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*, 2015.

[122] Z. C. Lipton, S. Vikram, and J. McAuley. Generative concatenative nets jointly learn to write and classify reviews. *arXiv preprint arXiv:1511.03683*, 2015.

[123] C.-W. Liu, R. Lowe, I. 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*, pages 2122–2132. Association for Computational Linguistics, 2016.

[124] 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. *arXiv preprint arXiv:1603.08023*, 2016.

[125] Y. Liu, W. Bi, J. Gao, X. Liu, J. Yao, and S. Shi. Towards less generic responses in neural conversation models: A statistical re-weighting method. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2769–2774, 2018.

[126] R. Logan, N. F. Liu, M. E. Peters, M. Gardner, and S. Singh. Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971, Florence, Italy, July 2019. Association for Computational Linguistics.

[127] 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. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1116–1126, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[128] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

[129] Y. Luan, Y. Ji, and M. Ostendorf. Lstm based conversation models. *arXiv preprint arXiv:1603.09457*, 2016.

[130] K.-M. Lux, M. Sappelli, and M. Larson. Truth or error? towards systematic analysis of factual errors in abstractive summaries. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 1–10, Online, Nov. 2020. Association for Computational Linguistics.

[131] F. Mairesse and M. Walker. Personage: Personality generation for dialogue. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 496–503, 2007.

[132] W. C. Mann and S. A. Thompson. Relational propositions in discourse. *Discourse processes*, 9(1):57–90, 1986.

[133] W. C. Mann and S. A. Thompson. *Rhetorical structure theory: A theory of text organization.* University of Southern California, Information Sciences Institute, 1987.

[134] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[135] L. Massarelli, F. Petroni, A. Piktus, M. Ott, T. Rocktäschel, V. Plachouras, F. Silvestri, and S. Riedel. How decoding strategies affect the verifiability of generated text. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 223–235, Online, Nov. 2020. Association for Computational Linguistics.

[136] J. Maynez, S. Narayan, B. Bohnet, and R. McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics.

[137] P.-E. Mazaré, S. Humeau, M. Raison, and A. Bordes. Training millions of personalized dialogue agents. *arXiv preprint arXiv:1809.01984*, 2018.

[138] K. R. McKeown. Discourse strategies for generating natural-language text. *Artificial Intelligence*, 27(1):1–41, 1985.

[139] S. W. McRoy, S. Channarukul, and S. S. Ali. An augmented template-based approach to text realization. *Natural Language Engineering*, 9(4):381–420, 2003.

[140] D. L. Medin and J. G. Bettger. Presentation order and recognition of categorically related examples. *Psychonomic bulletin & review*, 1(2):250–254, 1994.

[141] H. Mei, M. Bansal, and M. R. Walter. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. *arXiv preprint arXiv:1509.00838*, 2015.

[142] H. Mei, M. Bansal, and M. R. Walter. Coherent dialogue with attention-based language models. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[143] S. J. Mielke, A. Szlam, Y.-L. Boureau, and E. Dinan. Linguistic calibration through metacognition: aligning dialogue agent responses with expected correctness. *arXiv preprint arXiv:2012.14983*, 2020.

[144] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.

[145] J. D. Moore and C. L. Paris. Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational linguistics*, 19(4):651–694, 1993.

[146] J. D. Moore and M. E. Pollack. A problem for rst: The need for multi-level discourse analysis. *Computational linguistics*, 18(4):537–544, 1992.

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

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

[149] L. Mou, Y. Song, R. Yan, G. Li, L. Zhang, and Z. Jin. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. *arXiv preprint arXiv:1607.00970*, 2016.

[150] A. Newell. *Unified theories of cognition*. Harvard University Press, 1994.

[151] T. Niu and M. Bansal. Polite dialogue generation without parallel data. *Transactions of the Association of Computational Linguistics*, 6:373–389, 2018.

[152] D. Norris. Short-term memory and long-term memory are still different. *Psychological bulletin*, 143(9):992, 2017.

[153] J. Novikova, O. Dušek, A. Cercas Curry, and V. Rieser. Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[154] J. Novikova, O. Dušek, and V. Rieser. RankME: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[155] J. Novikova, O. Dušek, and V. Rieser. Rankme: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[156] S. Oraby, P. Gundecha, J. Mahmud, M. Bhuiyan, and R. Akkiraju. " how may i help you?" modeling twitter customer serviceconversations using fine-grained dialogue acts. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 343–355, 2017.

[157] 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*, pages 311–318. Association for Computational Linguistics, 2002.

[158] F. Petroni, A. Piktus, A. Fan, P. Lewis, M. Yazdani, N. De Cao, J. Thorne, Y. Jernite, V. Plachouras, T. Rocktäschel, et al. Kilt: a benchmark for knowledge intensive language tasks. *arXiv preprint arXiv:2009.02252*, 2020.

[159] F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

[160] A. Pomerantz and B. J. Fehr. Conversation Analysis: An approach to the analysis of Cocial interaction. In van Dijk, Teun., editor, *Discourse Studes: A Multidisciplinary Approach*, pages 165–190. Sage, 2011.

[161] R. Puduppully, L. Dong, and M. Lapata. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6908–6915, 2019.

[162] Q. Qian, M. Huang, H. Zhao, J. Xu, and X. Zhu. Assigning personality/identity to a chatting machine for coherent conversation generation. *arXiv preprint arXiv:1706.02861*, 2017.

[163] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

[164] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. 2019.

[165] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

[166] H. Rashkin, E. M. Smith, M. Li, and Y.-L. Boureau. I know the feeling: Learning to converse with empathy. *arXiv preprint arXiv:1811.00207*, 2018.

[167] E. Reiter. Has a consensus nl generation architecture appeared, and is it psycholinguistically plausible? In *Proceedings of the Seventh International Workshop on Natural Language Generation*, pages 163–170. Association for Computational Linguistics, 1994.

[168] E. Reiter and R. Dale. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87, 1997.

[169] E. Reiter and R. Dale. *Building natural language generation systems.* Cambridge university press, 2000.

[170] E. Reiter, R. Robertson, and L. Osman. Knowledge acquisition for natural language generation. In *Proceedings of the first international conference on Natural language generation-Volume 14*, pages 217–224. Association for Computational Linguistics, 2000.

[171] 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*, pages 583–593. Association for Computational Linguistics, 2011.

[172] A. Roberts, C. Raffel, and N. Shazeer. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online, Nov. 2020. Association for Computational Linguistics.

[173] S. Roller, Y.-L. Boureau, J. Weston, A. Bordes, E. Dinan, A. Fan, D. Gunning, D. Ju, M. Li, S. Poff, et al. Open-domain conversational agents: Current progress, open problems, and future directions. *arXiv preprint arXiv:2006.12442*, 2020.

[174] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[175] H. Sacks. *Lectures on Conversation, Volumes 1 & 2.* Blackwell, 1992.

[176] H. Sacks, E. A. Schegloff, and G. Jefferson. A simplest systematics for the organization of turn taking for conversation. In *Studies in the organization of conversational interaction*, pages 7–55. Elsevier, 1978.

[177] S. Santhanam, A. Karduni, and S. Shaikh. Studying the effects of cognitive biases in evaluation of conversational agents. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery.

[178] S. Santhanam, A. Karduni, and S. Shaikh. Studying the effects of cognitive biases in evaluation of conversational agents. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery.

[179] S. Santhanam and S. Shaikh. A survey of natural language generation techniques with a focus on dialogue systems-past, present and future directions. *arXiv preprint arXiv:1906.00500*, 2019.

[180] S. Santhanam and S. Shaikh. Towards best experiment design for evaluating dialogue system output. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 88–94, Tokyo, Japan, Oct.–Nov. 2019. Association for Computational Linguistics.

[181] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[182] E. A. Schegloff. *Sequence Organization in Interaction: A Primer in Conversation Analysis*. Cambridge University Press, 2007.

[183] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

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

[185] I. V. Serban, T. Klinger, G. Tesauro, K. Talamadupula, B. Zhou, Y. Bengio, and A. Courville. Multiresolution recurrent neural networks: An application to dialogue response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[186] 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.

[187] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[188] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. Courville, and Y. Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[189] L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1577–1586. Association for Computational Linguistics, 2015.

[190] S. Sharma, L. El Asri, H. Schulz, and J. Zumer. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799, 2017.

[191] X. Shen, H. Su, Y. Li, W. Li, S. Niu, Y. Zhao, A. Aizawa, and G. Long. A conditional variational framework for dialog generation. *arXiv preprint arXiv:1705.00316*, 2017.

[192] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro. Megatron-lm: Training multi-billion parameter language models using gpu model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

[193] P. E. Shrout and J. L. Fleiss. Intraclass correlations: uses in assessing rater reliability. *Psychological bulletin*, 86(2):420, 1979.

[194] K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston. Retrieval augmentation reduces hallucination in conversation, 2021.

[195] 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.

[196] 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. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205. Association for Computational Linguistics, 2015.

[197] 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*, 26(3):339–373, 2000.

[198] S. Sukhbaatar, J. Weston, R. Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[199] R. Sun. The importance of cognitive architectures: An analysis based on clarion. *Journal of Experimental & Theoretical Artificial Intelligence*, 19(2):159–193, 2007.

[200] I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 1017–1024, 2011.

[201] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[202] C. Thomson and E. Reiter. A gold standard methodology for evaluating accuracy in data-to-text systems. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 158–168, Dublin, Ireland, Dec. 2020. Association for Computational Linguistics.

[203] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[204] Z. Tian, R. Yan, L. Mou, Y. Song, Y. Feng, and D. Zhao. How to make context more useful? an empirical study on context-aware neural conversational models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 231–236, 2017.

[205] Z. Tian, R. Yan, L. Mou, Y. Song, Y. Feng, and D. Zhao. How to make context more useful? an empirical study on context-aware neural conversational models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–236, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[206] K. Van Deemter. Generating referring expressions: Boolean extensions of the incremental algorithm. *Computational Linguistics*, 28(1):37–52, 2002.

[207] K. van Deemter, M. Theune, and E. Krahmer. Real vs. template-based natural language generation: a false opposition. *Computational Linguistics*, 31(1):15–24, 2005.

[208] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[209] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[210] A. Venkatesh, C. Khatri, A. Ram, F. Guo, R. Gabriel, A. Nagar, R. Prasad, M. Cheng, B. Hedayatnia, A. Metallinou, et al. On evaluating and comparing conversational agents. *arXiv preprint arXiv:1801.03625*, 2018.

[211] O. Vinyals and Q. Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015.

[212] 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*, pages 271–280. Association for Computational Linguistics, 1997.

[213] M. A. Walker, O. Rambow, and M. Rogati. Spot: A trainable sentence planner. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–8. Association for Computational Linguistics, 2001.

[214] P. Wang, Q. Wu, C. Shen, A. Dick, and A. van den Hengel. Fvqa: Fact-based visual question answering. *IEEE transactions on pattern analysis and machine intelligence*, 40(10):2413–2427, 2018.

[215] W. Wang, M. Huang, X.-S. Xu, F. Shen, and L. Nie. Chat more: Deepening and widening the chatting topic via a deep model. In *SIGIR*, pages 255–264, 2018.

[216] X. Wang, W. Shi, R. Kim, Y. Oh, S. Yang, J. Zhang, and Z. Yu. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In A. Korhonen, D. R. Traum, and L. Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5635–5649. Association for Computational Linguistics, 2019.

[217] J. Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

[218] 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.

[219] J. Weston, S. Chopra, and A. Bordes. Memory networks. *CoRR*, abs/1410.3916, 2014.

[220] J. Weston, S. Chopra, and A. Bordes. Memory networks. *CoRR*, abs/1410.3916, 2014.

[221] M. White and R. Rajkumar. A more precise analysis of punctuation for broad-coverage surface realization with ccg. In *Proceedings of the Workshop on Grammar Engineering Across Frameworks*, pages 17–24. Association for Computational Linguistics, 2008.

[222] S. Wiseman, S. M. Shieber, and A. M. Rush. Challenges in data-to-document generation. *arXiv preprint arXiv:1707.08052*, 2017.

[223] T. Wolf, V. Sanh, J. Chaumond, and C. Delangue. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*, 2019.

[224] W. Wu, C. Xu, Y. Wu, and Z. Li. Towards interpretable chit-chat: Open domain dialogue generation with dialogue acts. 2018.

[225] C. Xing, W. Wu, Y. Wu, J. Liu, Y. Huang, M. Zhou, and W.-Y. Ma. Topic aware neural response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[226] C. Xing, Y. Wu, W. Wu, Y. Huang, and M. Zhou. Hierarchical recurrent attention network for response generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[227] C. Xiong, V. Zhong, and R. Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.

[228] 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*, pages 2048–2057, 2015.

[229] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 5754–5764, 2019.

[230] K. Yao, G. Zweig, and B. Peng. Attention with intention for a neural network conversation model. *arXiv preprint arXiv:1510.08565*, 2015.

[231] P. Ye, T. Wang, and F.-Y. Wang. A survey of cognitive architectures in the past 20 years. *IEEE transactions on cybernetics*, (99):1–11, 2018.

[232] T. Young, E. Cambria, I. Chaturvedi, H. Zhou, S. Biswas, and M. Huang. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[233] A. Zemel. Texts as actions: Requests in online chats between reference librarians and library patrons. *Journal of the Assoc. for Information Science and Technology*, 67(7):1687–1697, 2017.

[234] R. Zhang, J. Guo, Y. Fan, Y. Lan, J. Xu, and X. Cheng. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1108–1117, 2018.

[235] 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.

[236] 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? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[237] T. Zhang*, V. Kishore*, F. Wu*, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020.

[238] X. Zhang and M. Lapata. Chinese poetry generation with recurrent neural networks. In *EMNLP*, pages 670–680, 2014.

[239] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and B. Dolan. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*, 2019.

[240] Y. Zheng, G. Chen, M. Huang, S. Liu, and X. Zhu. Personalized dialogue generation with diversified traits. *arXiv preprint arXiv:1901.09672*, 2019.

[241] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[242] Z. M. Ziegler, L. Melas-Kyriazi, S. Gehrmann, and A. M. Rush. Encoder-agnostic adaptation for conditional language generation. *arXiv preprint arXiv:1908.06938*, 2019.