

# Dynamic Composition for Conversational Domain Exploration

Idan Szpektor, Deborah Cohen, Gal Elidan, Michael Fink, Avinatan Hassidim, Orgad Keller, Sayali Kulkarni, Eran Ofek, Sagie Pudinsky, Asaf Revach, Shimi Salant, Yossi Matias  
{szpektor,debbycohen,elidan,fink,avinatan,orgad,sayali,eranofek,sagiepud,asafr,shimi,yossi}@google.com  
Google Research

## ABSTRACT

We study *conversational domain exploration* (CODEX), where the user’s goal is to enrich her knowledge of a given domain by conversing with an informative bot. Such conversations should be well grounded in high-quality domain knowledge as well as engaging and open-ended. A CODEX bot should be proactive and introduce relevant information even if not directly asked for by the user. The bot should also appropriately pivot the conversation to undiscovered regions of the domain. To address these dialogue characteristics, we introduce a novel approach termed *dynamic composition* that decouples candidate content generation from the flexible composition of bot responses. This allows the bot to control the source, correctness and quality of the offered content, while achieving flexibility via a dialogue manager that selects the most appropriate contents in a compositional manner. We implemented a CODEX bot based on dynamic composition and integrated it into the Google Assistant. As an example domain, the bot conversed about the NBA basketball league in a seamless experience, such that users were not aware whether they were conversing with the vanilla system or the one augmented with our CODEX bot. Results are positive and offer insights into what makes for a good conversation. To the best of our knowledge, this is the first real user experiment of open-ended dialogues as part of a commercial assistant system.

## KEYWORDS

Conversational system, personal assistant, open-ended dialogue

### ACM Reference Format:

Idan Szpektor, Deborah Cohen, Gal Elidan, Michael Fink, Avinatan Hassidim, Orgad Keller, Sayali Kulkarni, Eran Ofek, Sagie Pudinsky, Asaf Revach, Shimi Salant, Yossi Matias. 2020. Dynamic Composition for Conversational Domain Exploration. In *Proceedings of The Web Conference 2020 (WWW ’20)*, April 20–24, 2020, Taipei, Taiwan. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3366423.3380167>

## 1 INTRODUCTION

Building on breakthroughs in machine learning and natural language processing, conversational AI has grown rapidly in the last decade and now touches the daily life of millions of users, notably via personal assistant systems like Alexa, Cortana, Google Assistant and Siri. Conversational AI systems are typically divided into three

This paper is published under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International (CC-BY-NC-ND 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.

WWW ’20, April 20–24, 2020, Taipei, Taiwan

© 2020 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC-BY-NC-ND 4.0 License.

ACM ISBN 978-1-4503-7023-3/20/04.

<https://doi.org/10.1145/3366423.3380167>

Who won the Warriors game?

They won 132-100 against the Celtics today. Stephen Curry was the top scorer.

How many points did he score?

He scored 26 points in the game. Also, he had 12 rebounds and 10 assists, recording a triple-double.

Who do they play next?

The Warriors will next host the 76ers on April 20. The 76ers are ranked first in the Eastern Conference.

Figure 1: Example conversation with a dynamic composition bot in the Sports domain.

categories: question answering (QnA), task based (e.g., ordering pizza or getting a movie recommendation), and social chat bots, focusing on human-like interaction (see recent survey of [15]).

This characterization does not capture an important category of conversations that blends aspects of QnA and social bots: *Conversational domain exploration* (CODEX) of a domain. In such dialogues, the goal of the user is to enrich her knowledge of a given domain by conversing with a knowledgeable bot. Like QnA and task-based dialogues, such conversations are focused and well grounded in high-quality domain knowledge. Like social bots, the conversation should be engaging and open-ended. Importantly, a CODEX bot should be proactive and may introduce relevant information that the user did not directly ask for. It should also appropriately pivot the dialogue to undiscovered regions of the domain.

In this work, we tackle the challenge of building CODEX bots that effectively interact with real users. One requirement from CODEX bots is the ability to provide diverse responses that would fit the wide range of user inputs issued in open-ended dialogues. In state-of-the-art bots, this is achieved via retrieval or end-to-end generative models, which are typically based on a vast corpora of human-human dialogues. At the same time, as in task-based and QnA bots, responses in CODEX dialogues need to be tightly controlled and anchored in verifiable domain knowledge. Unfortunately, this is typically not the case for open-ended bots [12, 15, 34].

To cope with the unique challenge of the CODEX setting, we introduce a novel approach termed *dynamic composition*. This approach decouples candidate content generation from the flexible composition of bot responses. This allows the bot to control the source, correctness and quality of the content that it may offer. At the same time, it is able to achieve flexibility via a dialogue manager that selects the most appropriate content in a compositional manner. At the heart of the dynamic composition approach are: (1) a collection of content providers, such as news, facts, and questions;

(2) a dialogue manager (DM); (3) a sentence fusion module. In each bot turn, the providers and DM participate in a *composition loop*. In each step of the loop, all providers offer candidates for the next utterance to be added to the constructed bot response. The DM then selects one appropriate utterance or ends the loop. Finally, the selected sequence of utterances is fused into a cohesive bot response. Examples for CODEX dialogues<sup>1</sup> conducted with a dynamic composition bot are shown in Figs. 1 and 2. Utterances from different providers in the same bot turn appear in different colors.

When selecting the series of utterances for a compositional response, a CODEX bot based on dynamic composition should construct high quality responses while globally striving to maintain an engaging experience. These short and long term objectives include many implicit sub-goals, such as identifying interesting facts that would fit together in a single response, avoiding repetitions along the dialogue, choosing the appropriate response length, and proactively changing the course of the dialogue at the right time. To this end, we implement a dynamic composition bot whose collection of content providers includes informative providers, offering content from various sources, e.g., a knowledge-graph and news articles, and conversation drivers, such as questions and topic changers, which aim at increasing user engagement. The bot’s DM follows the hierarchical recurrent neural network (RNN) paradigm [39], which captures the low-level characteristics of the immediate user inputs and bot responses as well as the dialogue-level state.

To test the effectiveness of the dynamic composition approach, we integrated our bot into the Google Assistant, allowing it to converse about the NBA basketball league as an example domain. Users were randomly selected for the experiment, without any filtering. Importantly, in contrast to Alexa prize experiments [36], this experiment was a seamless experience so that users were not aware whether they were conversing with the vanilla assistant system or the one augmented with the CODEX bot. Experimental results are positive, showing the potential of dynamic composition for CODEX dialogues, and offer insights into what makes for a good conversation. To the best of our knowledge, this constitutes the first real-user experiment of open-ended dialogues as part of a commercial assistant system.

Our contributions are as follows: (a) we introduce the novel dynamic composition approach (Section 3); (b) we describe the first implementation of a dynamic composition bot for CODEX dialogues (Section 4); (c) we present detailed human evaluation of our bot (Section 5); and (d) we report results and insights of a 6 months live experiment conducted in the Google Assistant (Section 6).

## 2 RELATED WORK

Broadly speaking, two main paradigms are followed in state-of-the-art open-ended bots: retrieval-based and generative. The body of literature for both is extensive so we only outline the central trends.

Information retrieval models select the most appropriate bot response out of a very large index of responses that were extracted from previously conducted open-ended dialogues. Such dialogues can be mined from publicly available human/human conversations, e.g., Twitter, Reddit and Weibo. Prior work focused on improving

<sup>1</sup>In all example dialogues with real users, we altered all entity mentions (e.g., teams, athletes), dates, figures, and other user-identifiable information for privacy purposes.

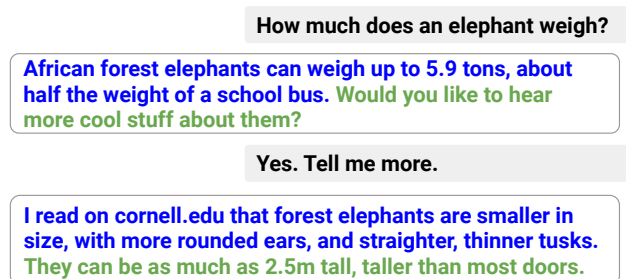


Figure 2: Example conversation with a dynamic composition bot in the Animal domain.

the retrieval quality of the next response in a dialogue, given the response index. One direction is to match between an embedding of the conversation history and the embedding of candidate responses [53, 54, 60, 63]. Another approach looks at word relationships between the conversation history and each candidate response, computing semantic similarity based on these [32, 45, 50, 64]. Yang et al. [56] apply pseudo relevance feedback to expand candidate responses with related external-knowledge terms. Fedorenko et al. [14] bias negative training examples to avoid selecting candidates that are highly similar to posts in the conversation history.

Retrieval-based dialogues sound natural and usually do not contain syntactic or prosody errors, since bot responses are copies of human responses. In addition, inappropriate content may be filtered and controlled offline at index construction. These merits made the retrieval approach the main choice for industrial chatbot systems [35, 42]. On the other hand, the set of candidate responses is fixed once the index is created, so it is harder to offer an appropriate response for every dialogue context. Specifically, offering knowledge-rich responses is limited. In contrast, we aim for a flexible open-ended bot that is grounded in domain knowledge.

The second main paradigm for modeling open-ended dialogues is to generate the bot response on the fly. One way to do so is by instantiating templates with selected structured content. This approach is common in practical chatbots, including the top Alexa prize models [4, 11, 13, 33]. Templates are usually built manually, and a model is learned to select the most appropriate one. This simplifies generation and learning and makes the model better grounded with correct structured data [20, 49]. On the other hand, it is hard to maintain a collection of templated responses that fit any context in open-ended dialogue.

To achieve maximum flexibility in the wording of the returned response, many open-ended dialogue models perform end-to-end generation of bot responses given the dialogue history. Virtually all such models learn a neural encoder/decoder generative model [3, 5, 6, 24, 28, 37, 39, 40, 46, 51, 52]. This approach is capable of generating appropriate responses to any dialogue context, and can be conditioned on semantic attributes, such as persona, emotion and topics [7, 8, 10, 41, 47, 59, 65]. However it suffers from lack of diversity and often returns “safe” generic responses. It is also hard to guarantee that the generated text reflects real or correct world knowledge, and such models tend to hallucinate fictitious information. Many recent works attempt to address the diversity [25, 30, 44, 57, 58, 61] and grounding [12, 17, 26, 34, 62] issues in neural text generation. However, performance end-to-end models

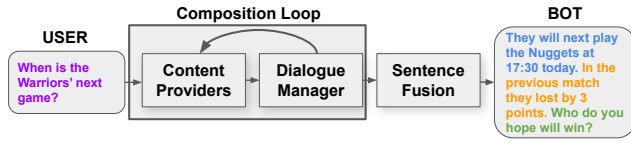


Figure 3: Dynamic composition flow diagram.

is brittle enough so that they are not the method of choice for industrial chatbots where the response quality is under scrutiny.

Several works investigated the use of an ensemble of models, each offering its own candidate response, together with a dialogue manager that selects between them. Some works combine retrieval-based and generation-based models [23, 38, 42, 43, 55]. Others select between modules that fit a specific type of response or a specific domain, such as social chat, movies, news and question answering [13, 22]. Unlike our proposed compositional approach, in all of these works, only a single candidate is selected for the bot response.

### 3 DYNAMIC COMPOSITION

In this section we introduce our novel *dynamic composition* approach as illustrated in Fig. 3. In the next section, we describe a concrete implementation of a CODEX bot based on this approach.

Dynamic composition relies on two main concepts. First, it decouples content generation from content selection. Content generation is performed by *content providers*, which offer candidate utterances. Each content provider is “specialized” and relies, for example, on a combination of a single source (e.g., news, a knowledge graph, or a dialogue corpus such as Reddit), a particular algorithm (e.g., retrieval-based, templates, encoder/decoder), or a specific dialogue act (e.g., answer, question, enhancement, acknowledgement). Given a set of candidates and the context of the conversation so far, utterance selection is performed by a learned *dialogue manager (DM)*.

Second, in dynamic composition, a single bot response is a composition of several utterances, possibly from different providers. This composition is performed in two stages, outlined in Fig. 3. In the first one, the DM and the content providers participate in a *composition loop*. In each step of the loop, providers generate candidates for the next utterance to be appended to the response constructed so far. Importantly, the providers are contextual and have access to the currently constructed response in addition to the conversation history. This allows them to offer utterances that refer to information already selected for the prefix of the response. The DM then selects a candidate and appends it to the sequence of selected utterances. This step is repeated until the DM assesses that the response contains enough utterances. The composition loop for the first bot turn in the dialogue from Fig. 1 is illustrated in Fig. 4.

In the second stage, *sentence fusion* is applied to combine the sequence of selected utterances into a coherent response by modifying the surface texts without altering their semantics. Such changes may include pronominalization, introduction of discourse markers, and compactization of repeated information.

The dynamic composition approach maintains the correctness and quality of the proposed content through the development of focused providers. At the same time, it enables a flexible composition of information from different sources and of different types within the same bot response, and tailored for the specific conversation context. Fig. 3, which shows a response in the NBA domain,

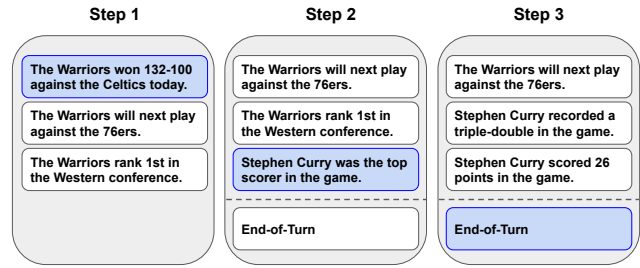


Figure 4: Composition loop. A few candidates from different providers are shown in each step and the one selected by the DM is highlighted in blue.

demonstrates the composition of a direct answer to a user question, a reflective enriching information on the provided answer, and a conversational driver in the form of a question. In Fig. 2, which exemplifies a conversation in the animal domain, the second bot response exhibits the combination of a retrieved Web content and a structured fact from a knowledge graph.

We next describe a CODEX bot implementation that relies on dynamic composition, which was used in all of our experiments.

## 4 CODEX BOT

There are four main building blocks that are involved in building our dynamic composition CODEX bot: 1) Natural Language Understanding (NLU) analysis and representation; 2) a set of content providers; 3) a DM model; 4) a sentence fusion module. We next describe each of these components.

### 4.1 Natural Language Understanding

A key factor in maintaining a coherent and engaging CODEX conversation is the ability to correctly identify the current focus of the conversation. Equally important is the ability to change the focus in order to drive domain discovery and exploration. To this end, we explicitly model and track the focus, as described below.

#### 4.1.1 Focus Representation.

We define the *focus* of the conversation to be the set of entities that the conversation is about, where entities may refer to proper nouns such as people and organizations, common nouns such as animals, events such as games, and even properties such as height or mass.

We use a two-level salience scale for entities in the focus. *Primary entities* are the prominent entities in the conversation at a given point. For example, in “*I like lions*”, ‘lion’ is the primary entity. More than a single entity may be part of the focus. For example, in the text “*how would you compare the 76ers to the Celtics*” the focus includes two primary entities: ‘76ers’ and ‘Celtics’. A focus can be augmented with *secondary entities*, which offer a scoping mechanism for the discussion around the primary entities. For example, we represent the focus in “*Curry played well in the Warriors-Lakers game*” as [Primary{‘Stephan Curry’}, Secondary{‘Warriors-Lakers game’}]. Similarly [Primary{‘lion’}, Secondary{‘weight’}] refers to the lion as the prominent entity in the dialogue, with the secondary entity indicating that the conversation revolves around its weight.

Both the user and bot responses in the dialogue are annotated with focus entries (see Table 1). The annotated focus is used by all of

**Table 1: Sample NBA conversation - focus tracking.**

	Dialogue	Focus
User	How are the Suns doing?	Suns
Bot	They won 97-95 against the Memphis Grizzlies last night.	Suns, last game
	They only lost once in the season, to the LA Clippers.	Suns, season
User	How did they do last night?	Clippers, last game
Bot	They won 100-94 against the Celtics.	Clippers, last game
	They will play against the 76ers tonight.	Clippers, next game

the bot components. Content providers may use the focus to limit the search space when offering candidates. Providers that specialize in focus change, which is a key driver of domain exploration and discovery, can directly use explicit focus annotation. The DM relies on features that indicate a change in focus to learn when to stay on topic and when to shift it to keep the conversation engaging. Finally, sentence fusion uses the focus for pronominalization and for adding discourse markers.

On the bot side, we require each candidate utterance offered by a provider to be annotated with its focus. On the user side, the NLU component infers the focus from the user input, as described next.

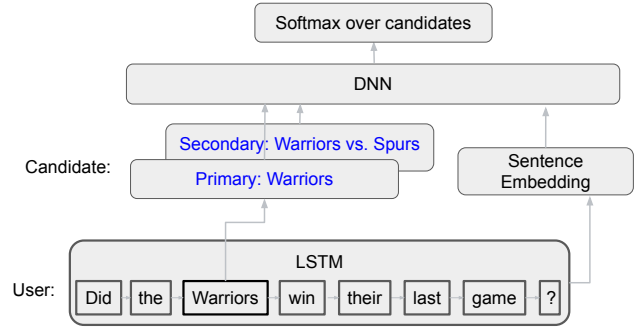
#### 4.1.2 Focus Tracking in User Input.

Our NLU component infers the conversation focus given a new user input and the conversation history using a supervised neural ranker. First, an entity linker is applied to map entities to nodes in a knowledge graph. Then, based on the annotations, the ranker generates a list of candidate focuses which, in addition to the last user input, include entities from the entire conversation. This allows for back referencing as well as implicitly remaining "on track". Using a per domain set of simple rules, we include additional related entities that are not explicitly mentioned in the dialogue.

Each focus candidate is represented using domain agnostic features, including the embedding of the focus entities, the entity types, whether the primary/secondary focus is identical to the current one, and whether its entities were mentioned in the last turn. We also incorporate domain-specific features. For the NBA domain, for example, we add whether a sports event is, and whether the focus contains an unresolved entity (e.g., a game or team that is mentioned but cannot be disambiguated). Finally, the user input is encoded at the token level by a bi-directional LSTM (biLSTM) [19]. The last hidden states of the biLSTM along with the above candidate focus features are fed to a DNN, and the scores are computed via softmax over the candidates. The focus tracking model is illustrated in Fig. 5. The model was trained first on 3K examples automatically labeled by a simple rule-based focus tracker, and then improved on 1.5K golden labels provided by human annotations.

#### 4.1.3 Factoid Question Parsing.

Users often ask concrete questions such as "What is the life span of an elephant?" or "Who was the top scorer?". To handle such questions, we followed the semantic parser approach in [48]. We constructed a context-free grammar that generates questions related to knowledge graph triplets of the target domain. We then generated examples with the grammar and trained a sequence-to-sequence model that parsed user inputs into knowledge-graph entries. We

**Figure 5: Focus tracking ML model diagram.****Table 2: Sample NBA conversation.**

	Dialogue	Provider Type
User	Who won the Warriors game?	
Bot	They won 132-100 against the Celtics today. Stephen Curry was the top scorer.	Fact Focus change + Teaser
User	How many points did he score?	
Bot	He scored 26 points in the game. Also, he had 12 rebounds and 10 assists, recording a triple-double.	Fact Fact Insight
User	Who do they play next?	
Bot	The Warriors will host the 76ers tomorrow. The 76ers have significantly more wins than the Warriors.	Fact Comparison
	Who do you think will win?	Bot question
User	The Warriors!	

note that, since not all user inputs are questions, in our training conversations (see Section 4.3), an expert labeled inputs that are not factoid questions, such as "yes", "Lakers game" as negative examples. The parser was trained to decode these as a NOP entry.

The model above considers each input as standalone, which is not the case in dialogues. We therefore augmented each input with the focus inferred by the focus tracker. This offers resolution for pronouns or implicit references to focus entities. On held out data, the trained factoid question parser reached 98% accuracy.

## 4.2 Content Providers

We next describe the different content providers in our bot. We distinguish between informative providers, which offer facts and opinions, and conversational drivers, whose goal is to proactively increase user engagement. Providers also differ in their reliance on structured data, specifically a knowledge graph, and unstructured data sources. Structured providers generate their texts using templates, and declare their structured data, which may be used in subsequent composition. Unstructured content is quoted verbatim with proper attribution. A sample dialogue in Table 2 illustrates the composition of several of the different providers described below.

#### 4.2.1 Knowledge-graph Facts.

This structured informative provider exposes pieces of information about knowledge graph entities in the primary focus, such as "Kevin Durant scored 35 points in the Warriors vs. Lakers game", or "Horses

*can run as fast as 55 MPH*". For each entity, the knowledge graph has many different facts that may be surfaced. In order to offer only interesting ones as candidates, the fact provider includes an internal ranking function,  $\text{FactRank}(\text{entity}, \text{attribute}, \text{value})$ , that takes into account several features of the fact: does it describe some extremum (e.g., the fastest animal)? is it far from the average (e.g., a team scoring far more points in a game than their season average)? how broad/specific is it (e.g., the weight of female Asian elephants in captivity vs. elephant weight on average)? The ranker is a linear function over the above features with manually tuned weights.

#### 4.2.2 Comparisons.

This structured informative provider compares a specific attribute of two entities, such as "*Cheetahs are faster than lions*" and "*the Warriors and the Rockets have similar wins in this season*". If the focus includes two primary entities, these are compared. Otherwise, entities for comparison are chosen using an entity relatedness matrix,  $\text{ReLated}(a, b)$ , which was constructed using an in-house model similar to [21] based on entity co-mentions in Web documents or Web queries. If an attribute (e.g., 'speed') is part of the focus, it is taken as the comparison dimension. Otherwise, facts based on shared attributes for the two entities are extracted and ranked using  $\text{FactRank}()$ . The attributes with highest fact rank score are selected for comparison.

#### 4.2.3 Followup Insights.

This structured informative provider suggests an additional insight for a structured fact presented in the previous utterance. For example, given "*Kevin Durant scored 35 points in the game*", this provider may offer the insight "*This is his season high*". This is done by assessing if the referred fact occurs as an extremum across some aggregated dimension. Examples include aggregation over a time period (e.g., career, season, last 10 games), or entities subsets (e.g., land mammals, bench players, home games). As before,  $\text{FactRank}()$  is employed to offer only the top interesting insights.

#### 4.2.4 Web Facts.

This unstructured informative provider uses a home-made Web crawler to collect sentences in Web pages that convey facts on target entities [31]. At serving, the provider offers candidates from the collected index for primary entities in the focus. The first step of the second bot turn in Fig. 2 is an example of a collected fact.

#### 4.2.5 News.

This unstructured informative provider adapts an extractive summarization method to obtain relevant information from news articles. To this end, the provider crawls the Web periodically (every few hours), and updates an index with recent news articles that mention entities in the domain. Next, snippets from the newly collected articles are extracted and scored by performing multi-document summarization using  $\text{TextRank}$  [29]. We filter out snippets that have either a low score, are too short (less than 3 tokens), are too long (more than 20 tokens), or contain an unresolved co-references. At serving time, the provider offers the highest scored snippets relevant to the entities in the current focus. An example conversation that includes a news snippet at the end of the second bot turn is shown in Table 3.

**Table 3: Sample conversation - News provider**

User	Atlanta Hawks season record
Bot	The Hawks have 24 wins and 47 losses. The average rebounds per game for the Hawks in their last 10 games is 52.3, compared to their seasonal 46.
User	Hawks next game
Bot	The Hawks will next host the Houston Rockets today at 4:30pm in State Farm Arena. <i>I read on chron.com that the Rockets have won 11 of their past 12 games, including five straight in the road, matching their longest road winning streak of the season.</i>

#### 4.2.6 Questions.

Two conversational drivers offer questions as candidate utterances. The first one tests for further interest of the user in the current focus, e.g., "*Are you up for more information about turtles?*". The user answer provides an explicit indication for the DM whether to stay on topic. The second type of questions elicits user preferences to drive the conversation towards entities that are of greater interest, e.g., "*Which do you like more, cats or dogs?*". Typically, one entity is taken from the primary focus entities, while the other is taken from the relatedness matrix  $\text{ReLated}()$ . We also use domain specific questions that can capture useful user preferences, such as "*Who do you think will win?*" in Sports.

#### 4.2.7 Focus Changing.

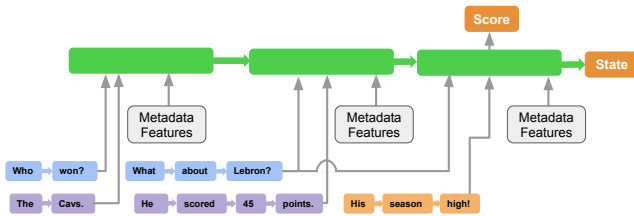
These conversational drivers intentionally generate candidates with entities that are outside of the current focus. One type of focus changers offers questions on entities related to the focus, as in "*Do you like cats too?*", when the current focus is 'dogs'. Another type of focus changers offers information about related entities, thus implicitly changing the focus. For example, if the first utterance in a bot response was "*The Raptors won*", a second utterance "*Kawhi Leonard was the top scorer*" changes the focus from the team to the athlete, while maintaining the respective game as the secondary entity. Relatedness either comes from  $\text{ReLated}()$  or through knowledge graph relations such as hypernyms/hyponyms (animal – animal family) and meronyms/holonyms (athlete – team).

#### 4.2.8 Teasers.

This conversation driver builds on interesting facts that are ranked highly by  $\text{FactRank}()$ . However, instead of conveying the full fact information, it offers only partial information. For example, instead of "*Stephen Curry was the Warriors' top scorer with 42 points*" it offers "*Stephen Curry was the Warriors' top scorer*". Similarly, instead of "*Lions sleep 16-20 hours each day*", it offers "*Lions sleep a lot*". This gives the opportunity to an engaged user to ask for the full information, or change the direction of the dialogue without receiving too much information on topics that the user is less interested in.

### 4.3 Dialogue Manager

The DM has the critical role of choosing from candidate utterances offered by the providers with the goal of composing a sequence of utterances that makes up a relevant and engaging bot response. As discussed, this is done in an iterative process termed *composition loop*. In each step, providers offer candidate utterances that take into



**Figure 6: DM hierarchical RNN architecture, including the user input RNNs (blue), previously selected bot utterance RNNs (purple), candidate utterance RNN (orange) and conversation-level RNN (green).**

account the current dialogue context and then the DM selects one of them. The composition loop algorithm is outlined in Algorithm 1.

#### 4.3.1 DM Architecture.

Hierarchical dialogue models have been showed to outperform flat models [39, 51, 60, 63]. Thus, inspired by [39], our DM is implemented as a two-level hierarchical RNN encoder, shown in Fig. 6. The input at each step is the last user input (in blue), a candidate utterance (in light orange), and the previous state. A first-level GRU [9] encodes each text into a sentence embedding vector. These are then fed to a second-level GRU (in green), along with metadata features. Thus, the context vector of the high-level RNN represents the whole conversation history as the conversation progresses. The second-level RNN unit also predicts a score for each candidate utterance, serving as a ranking function. The top scoring candidate is then selected, and its state becomes the grounding state for the next RNN step. On top of the RNN greedy selection of next utterance, we apply a beam search of depth 3. Each beam is scored by the sum of scores of all its selected utterances.

The metadata features include: (a) the provider type (question asker, focus changer etc.); (b) token-level cosine similarity between the candidate and previous bot utterances; (c) the number of tokens in the constructed response; (d) the number of tokens in the candidate utterance; (e) entity types in the candidate focus; (f) whether the candidate offers to change the focus. Content providers may also contribute features to the DM to use as signals. Specifically, we add the news provider score computed by TextRank.

The DM decides when to end the composition loop. This is done by introducing a special ‘End-Of-Turn’ candidate. When this candidate has the highest score, the loop ends. Candidate utterances can also indicate if they should be the last in the response (e.g., bot questions). When such utterances are selected, the loop ends after their addition. We also introduce a predetermined hard-limit on the number of utterances to be added (see Section 6.4).

If all candidate utterances in the first step of the composition loop are assigned a low score, we do not expect the bot response to be of good quality, and the DM ends the whole conversation. In our experiments, the score threshold was set to 0.7 achieving 90% precision. Thus, our DM combines three roles: (a) a ranker for selecting the best content; (b) a classifier for ending the composition loop; and (c) a classifier for ending the conversation.

#### 4.3.2 DM Training.

We train our DM in a supervised learning setting as follows. CrowdSource human evaluators conduct conversations with the bot and

**Input** : Dialogue history  $h$ , user input  $u$ , max turn steps  $M$ , RNN state  $s$

**Output** : Bot response, Updated RNN state  
 $bot\_response = \{\}$ ,  $step = 1$ ,  $end\_turn = false$ ;

```

while step <= M and end_turn = false do
  candidates ← CallProviders(h, u, b);
  scores, states ← ScoreCandidates(u, candidates, s);
  if step = 1 and max(scores) < 0 then
    | dialogue is ended;
  end
  selected ← arg max(scores);
  bot_response[step] ← candidates(selected);
  s ← states(selected);
  if selected should end turn then
    | end_turn = true;
  end
  step ← step + 1;
end
return bot_response, s;

```

**Algorithm 1:** Composition loop for a bot turn.

rate the bot responses. For each domain, we start with a few content providers and, as the DM quality improves, we progressively add providers, and collect additional rated training conversations that now include candidates from these new providers.

The evaluators are instructed to conduct conversations with specific personas in mind, e.g., a casual NBA fan. They are also given an example scenario from which they could seed a conversation – “Imagine you came home late from work missing your favorite team’s game and want to catch up”.

Each evaluator constructs training dialogues in two phases. First, she converses with the bot until the dialogue derails or comes to a natural end. Then, the evaluator rates the bot responses. In dynamic composition, all candidate utterances at every step of the composition is known, and the evaluator is asked to assess all of them. For example, in the bot turn illustrated in Fig. 4, all 11 candidates are rated. Thus, we obtain many training examples from a single conversation. A dozen expert CrowdSource evaluators generated ~1.5K conversations with an average of 3 responses, each with 2–3 utterances, and with 10–20 candidates for each utterance. Overall, this results in ~150K training examples.

Each candidate utterance is rated on a scale of -3 to 7, with no 0 rating. The negative scale refers to candidates that do not reply to a user’s question or are out of context. The positive scale corresponds to candidates that fit the conversation context. The broader positive rating scale enables higher granularity in evaluators’ preferences of relevance and interest, allowing the DM to learn such subtleties.

## 4.4 Sentence Fusion

The output of the composition loop is a sequence of utterances, which still need to be fused into a single coherent bot response. In particular, the providers offered these utterances as standalone sentences, not knowing the full context in which they will be conveyed. Thus, simple concatenation of all utterances would result in a cumbersome, unnatural and overly verbose response. For example,

**Table 4: Sentence fusion examples.**

Phenomenon	Before Fusion	After Fusion
Pronominalization	Cheetahs are the fastest land mammals.	They are the fastest land mammals.
Co-reference	James Harden was the top scorer in the Rockets vs. Lakers game.	Harden was the top scorer in the game.
Context Removal	Harden was the top scorer in the Rockets vs. Lakers game.	Harden was the top scorer.
Discourse Markers	Harden was the top scorer. Harden had the most turnovers.	Harden was the top scorer. But, Harden had the most turnovers.

in responding to the user input “How did James Harden do?”, the DM could pick the two utterances “James Harden was the top scorer in the game with 32 points” and “James Harden had 12 rebounds in the game”. A simple concatenation – “James Harden was the top scorer in the game with 32 points. James Harden had 12 rebounds in the game.” – sounds repetitive and unnatural. Such responses are especially unappealing when conveyed over voice-only devices, such as Google Home and Amazon Echo.

To remedy this, the last component of the dynamic composition approach is sentence fusion, whose goal is to combine ordered utterances into a single coherent and cohesive response [2, 16, 27]. To this end, we apply a rule-based mechanism that addresses the following fusion phenomena: (a) co-reference and pronominalization, (b) removal of repetitive context mentions, such as a game or a season, and (c) introduction of a discourse marker between sentences. Table 4 shows examples for each phenomenon. For the example above, our fusion implementation returns “James Harden was the top scorer in the game with 32 points. Also, he had 12 rebounds”.

To perform co-reference and pronominalization, we detect repeated entity mentions in the currently built response using entity-linking to a knowledge-graph. If an entity mention is unambiguous, meaning that no similar entity type was mentioned since the last mention of the target entity in the response, then co-referencing is allowed. In this case, mentions for animated entities are replaced by a pronoun, e.g., ‘it’ for ‘lion’, ‘he’ for ‘Lebron’ and ‘they’ for ‘Lakers’. We use the dependency parse tree of each utterance [1] for proper pronoun inflection. For non-animated entities, we use the type as the co-reference, e.g., replacing ‘Lakers vs. Boston game’ with ‘game’. If a non-animated entity is unambiguously mentioned in consecutive utterances, we remove the repeated mentions altogether, assuming that the context is clear. We note that only entities in the conversation domain are considered for co-reference or mention removal (e.g., teams, games etc. for sports).

Our fusion implementation also adds discourse markers that indicate the sentiment relation between two consecutive utterances. Specifically, we first detect consecutive utterances in which the same entity is the focus. We then analyze the sentiment of each utterance w.r.t. that entity and if the sentiment is positive or negative and similar in both utterances, we add a continuation marker, e.g., ‘also’. If the sentiment is contradicting, we add a negation marker, e.g., ‘but’. This is exemplified in the last entry in Table 4.

We also tried using a sequence-to-sequence ML model trained on the DiscoFuse dataset [16]. Fusion performance, however, decreased noticeably. This could be due to the fact that the dataset has Wikipedia and news examples, which are inherently different from dialogue examples. We plan to study transfer learning between fusion corpora in future work.

**Table 5: End-to-end evaluation results.**

Bot Version	Average rating
Vanilla Google Assistant	3.89
CODEX Bot	4.45

## 5 HUMAN-RATED EVALUATION

We conducted human-rated evaluation of our CODEX bot for the NBA domain. We first report the results of an end-to-end evaluation, comparing our bot to a vanilla Google Assistant. We then provide additional component-wise analysis.

### 5.1 Bot Setup

Our CODEX bot (Section 4) is set up as follows. The focus tracking model is trained with a learning rate of 0.001 and batch size of 20. Its DNN has 4 layers, each with 10 units. Its LSTM has a single layer with 40 units. The DM model is trained with a learning rate of 0.0001, batch size of 16, and dropout probability of 0.8. Its architecture includes 2 GRU layers, each with 200 units. The focus annotated dataset and DM annotated dataset contain 4.5K and 150K examples, respectively. Both were split into 80% train, 10% validation and 10% test sets. All hyperparameters are selected using Google Vizier [18], performing random grid search on the validation set.

### 5.2 End-to-end evaluation

We would like to assess whether users prefer a dynamic composition bot when discussing a domain over the typical question-answer approach in current assistants. To do so, we asked a team of CrowdSource human evaluators to conduct dialogues on NBA with the vanilla Google Assistant as well as our bot and rate the overall conversation experience on a scale of 1 to 5. Importantly, these evaluators did not experience the CODEX bot beforehand. Additionally, the user interface for conducting dialogues was indistinguishable between the two settings. Further, the evaluation was blind-folded so that evaluators did not know which bot they were conversing with in each conversation. Overall, the evaluators conducted and rated 200 dialogues with at least 3 turns for the two bot settings.

The results are presented in Table 5. The relative improvement in dialogue quality when conducting conversations with the CODEX bot is 14% (statistically significant at  $p < 0.01$ ). Analysis of the rated conversations reveals that the improvement in quality is due to two main factors: (a) the bot’s ability to maintain the dialogue context even when given ambiguous user inputs; (b) the additional exploratory information that enriches the bot responses. In particular, the enriched responses created more engaging dialogues in which the users continued conversing, picking up on the dialogue directions introduced by the bot. Both of the above are encouraging indications that dynamic composition can achieve better dialogue

**Table 6: Illustrated conversation - vanilla Google Assistant**

User	Who won the Warriors game?
Bot	They won 132-100 against the Celtics today.
User	How many points did Curry score?
Bot	Stephen Curry scored 26 points in the game.
User	Who do they play next?
Bot	The Warriors will next host the 76ers on April 20.

**Table 7: Focus tracking model evaluation**

Model	Precision@1
Domain-agnostic-only	83.0%
DNN	90.5%
Full model	92.0%
Best achievable	95.0%

quality for conversational domain exploration. As an example, conducting a dialogue similar to Fig. 1 with the vanilla Google Assistant would require the user to lead in all turns, as illustrated in Table 6.

### 5.3 Components Evaluation

We next present an analysis of individual components: the focus tracking, the DM, and the fusion modules.

#### Focus tracking

Test accuracy of the learned focus tracking model is shown in Table 7, with several ablation tests. The *Full model* combines a biLSTM that encodes the last user input with other features (see Section 4.1.2). The *DNN* model includes all features fed into the DNN but not the biLSTM encoder. The *Domain-agnostic-only* model includes only domain agnostic features fed into the DNN.

As can be seen in Table 7, both domain specific features and the text encoding via the biLSTM are helpful in getting closer to the *best achievable* performance. Note that 100% accuracy is unattainable due to human-rater disagreements. For example, one rater annotated the focus in “*How are the Warriors doing?*” as referring to the team’s last game while another annotated it with the season.

Focus tracking is an important component in our bot implementation and its performance has a large impact on the dialogue quality. The example in Table 8 illustrates this. In its third turn, the user changed the focus from the season back to the last game, but the bot wrongly tracks it to be still on the season and was thus unable to construct a relevant response.

#### Dialogue Manager

We evaluated both the architecture and the features of the DM model as follows. We compute *avg-rating@1*: the human-annotated rating (see Section 4.3.2) for each selected candidate utterance by the DM, averaged across all turn steps in the test set. The maximum achievable *avg-rating@1* – if we would always choose the top rated candidate – is 6.09; random selection achieves 0.64. Our full DM model (Section 4.3) achieves an *avg-rating@1* of 5.21.

To test the impact of the conversation history, we trained a DM version that replaces the second-level RNN (Fig. 6) with a DNN with no recurrent connections. This model achieves an *avg-rating@1* of 4.74, a drop of 10% compared to the full model. This shows that

**Table 8: Sample conversation - wrong secondary focus**

User	did the Rockets beat the Warriors tonight
Bot	No, the Rockets lost 109-99 yesterday against the Warriors.
User	when do they play next
Bot	The Houston Rockets will next host the Warriors on May 24 at 5:00pm in Toyota Center. The average points per game against the Rockets in their last 10 games is 96.9, compared to their seasonal 104.3.
User	how many points did James Harden score
Bot	The total points for James Harden in the current season are 1843. He has 28.2 points, 6.8 rebounds and 7.8 assists per game.
User	how many points did James Harden score last night against the Warriors
Bot	James Harden scored 28 points in the game.

**Table 9: Ablation study for DM metadata features.**

Removed features	avg-rating@1
Provider type	4.99
Sequence lengths	5.08
Text similarity	5.17
Focus type + Focus changed	5.20
Full model	5.21

conversation history is important in our collected CODEX dialogues and that our second-layer RNN can make good use of it.

We next evaluate the gain that can be attributed to the addition of metadata features on top of encoding of the conversation text. To this end, we removed a single feature family from the full DM model at a time in an ablation study. The results are provided in Table 9. These indicate that explicit focus and text similarity features have little effect, possibly because they are already captured well via the GRU encoder. Sequence lengths are not easy to maintain using RNNs, and thus adding them directly helps. The feature that contributes most is the semantic type of each candidate utterance, i.e., the provider type, which probably helps the model learn useful compositional patterns at the pragmatics level.

#### Fusion

Finally, we conducted the following blind side-by-side experiment of two bot variants: one with our sentence fusion mechanism, and the other with a naive concatenation of the selected utterances. The annotators were asked to annotate which version they prefer. In 76% out of 100 randomly sampled examples, the fusion variant was preferred, in 17% of the cases the non-fused variant was preferred, and in the other cases there was no preference. This is a clear indication that fusion improves the quality of composite responses.

## 6 LIVE EXPERIMENT

In the previous section, we described the performance of our dynamic composition bot as evaluated by dedicated human raters. Such evaluation allows for controlled settings and is necessary before deployment in a user-facing commercial product. However, dedicated human evaluators typically behave differently from real users. First, they do not cover the full range of information users



need (e.g., inquiring about which TV channel will transmit an NBA game or setting a reminder for watching a game). Second, they usually do not mimic real user conditions, such as rushing to work in a car, or sitting in front of the TV with background noise. Third, evaluators are co-operative, and will conduct a conversation with a bot even if the conversation reaches an awkward or inappropriate stage. Finally, having only dozens of evaluators introduces biases to personal tastes of what makes for a good conversation in a certain domain. Thus, to gain accurate understanding of how real users interact with our CODEX bot, we conducted a live experiment.

## 6.1 Experimental Setup

For the live experiment, we use our CODEX bot implementation for the NBA domain. To interact with users under a real online setting, we integrated the bot into the Google Assistant, termed onward as *assistant*. The experiment was conducted under an A/B testing protocol, in which a small percentage of assistant users were randomly sampled to interact with an assistant version augmented with our bot, termed *experiment*, and other users (same percentage) were sampled to interact with the vanilla assistant, termed *control*. This experiment spanned the entire 2018-2019 NBA season during which user assignment to control/experiment remained constant.

Whenever a user queried for information about an NBA team, the control returned the default assistant answer, while in the experiment, our bot was triggered. On top of answering the requested information, our bot tried to engage the user into conversing about the NBA. This interaction was seamless to the users, who could not distinguish between the vanilla assistant and the one integrated with the CODEX bot. Once started, a conversation with a user could end if the bot predicted that its response is not of sufficient quality (see Section 4.3), if the user issued a query not in the NBA domain (e.g., about the weather), or if the user issued a standard stop command. The last two options were handled by the assistant.

The bot implementation in the live experiment was as described in Section 4 except for bot questions and the news provider, which were turned off due to product decisions.

## 6.2 Bot Response Metrics

During the experiment, we measured daily user interaction with the assistant about the NBA domain in both the experiment and the control versions. In order to assess user engagement, we rely on several surrogate metrics that could be directly measured from the interaction logs. First, we define a *logged conversation* to be the succession of user and bot turns, starting with a triggering user turn (Section 6.1). We counted the following conversation metrics:

- An *NBA query* is a user query that contains a mention of an NBA-related entity or invites an NBA-related response.
- The *Followup rate* is the fraction of bot turns followed by a user input occurring less than 100 seconds afterwards.
- The *NBA followup rate* is the fraction of bot turns followed by an NBA query.
- The *Other-sports followup rate* is the fraction of bot turns followed by a user query which has a non-NBA but still sports-related entity, e.g., NFL or MLB.
- *Explicit positive feedback* refers to followup user queries with explicit gratitude such as “thank you” and “wonderful”.

**Table 10: Mean relative change of experiment vs. the control.**

Metric	Difference (%)
Number of NBA queries	+3.9
Followup rate	+2.9
NBA followup rate	+4.2
Other-sports followup rate	+3.1
Explicit positive feedback	+15.6
Explicit negative feedback	+23.5

- *Explicit negative feedback* refers to followup user queries that contain negative feedback, such as “stop” and “shut up”.

For the last two metrics, we use predefined lists of positive and negative feedback phrases collected from user logs.

## 6.3 Main Results

The average relative change in the metrics of the experiment w.r.t the control is shown in Table 10. All changes except positive feedback are statistically significant at  $p < 0.01$ . The increase in user engagement with the assistant in the experiment is clear along multiple metrics. First, at the aggregate level, an increase of 3.9% in the number of daily NBA queries per user is a strong indication that, overall, users like the enriched responses of our CODEX bot, which contain more information than requested in the user query. Second, the metrics that pertain to the progress of the conversation itself reveal an increase of 2.9% in followup user queries, resulting in longer user-assistant interaction.

Looking at types of followup queries, the most frequent one is the followup NBA-related query, a continuation of an NBA conversation. The *NBA followup rate* metric increased relatively by 4.2%, indicating that users were more engaged as they conducted longer NBA multi-turn dialogues with our bot compared to vanilla assistant. The next most frequent followup query type is about other sports fields. The relative increase of 3.1% in *Other-sports followup rate* suggests that users who liked the bot’s NBA experience sought a similar experience for other types of sports.

Some followup queries contain explicit feedback and, while more than an order of magnitude fewer than the followup queries discussed above, such queries represent direct user (dis)satisfaction from the bot response. Positive feedback was increased by 15.6% in the experiment. But, negative feedback increased as well, by 23.5%. This shows that some users prefer only direct answers. This could depend on the situation the user is in. For instance, willingness to engage in a CODEX dialogue at home, but not at work. In future work, we plan to explore ways to model these differences so as to offer a better personalized and timely experience.

## 6.4 Device Type Analysis

Users engage with the assistant via different types of devices. We analyzed the differences in interaction when a user used a voice-only device, e.g., Google Home, compared to a device with a screen, e.g., a mobile phone. The relative increase in number of NBA queries and followup rate are respectively 8% and 24% higher in devices with a screen compared to screenless devices. We also found that most of the explicit feedback by users (both positive and negative) is issued on voice-only devices. This may hint that users find it

**Table 11: Experiment sample conversation - teaser.**

User	next Indiana Pacers game
Bot	They will next play against the Milwaukee Bucks tomorrow at 2:00am in Fiserv Forum. The last game between them ended 111 - 105 to the Pacers. Victor Oladipo contributed less than usual.
User	how many rebounds did he have
Bot	Victor Oladipo was the top rebounder in the Pacers vs. Bucks game with 13. Additionally, he contributed 18 points and 1 assist, recording a double double.
User	how many points did he have
Bot	Victor Oladipo scored 18 points in the game.

harder to interact with such devices, and therefore improved or reduced quality is more explicitly noted.

In addition, we found that bot replies with more than two utterances resulted in a significant rise in bad feedback in voice-only devices. Therefore, for such devices, the length of each composed bot response was limited early in the experiment to at most 2 utterances (as opposed to 3 for devices with screens). We plan to further investigate the modeling of differences in devices in future work.

## 6.5 Conversation Driver Analysis

One of our primary goals was to develop a CODEX bot that does not only answers a user’s question but also proactively drives the conversation. One way to do so under dynamic composition is to enrich the response with additional factual content. Another option is to incorporate conversation drivers to more explicitly drive the dialogue towards other topics in the domain (see Section 4.2).

In the live experiment, our bot made use of teasers, as exemplified in Fig. 1 and Table 11. To assess the effect of teasers, we measured the NBA followup rate after the first conversation turn when the first bot response contained a teaser compared to dialogues when it did not. We found that enriching the vanilla bot response with a fact increases the NBA followup rate of the first turn by 2.5%. Yet, enriching the first bot response with a teaser increases the followup rate by 13%, revealing teasers as effective conversation drivers. This also suggests that users are willing to engage in a multi-turn dialogue if they expect it to be interesting.

We next analyzed the effect of focus change. This implicit conversation driver is realized by responses in which the bot adds enriching utterances whose focus is different from that of the user query. This, in essence, drives the conversation to other regions of the domain, as exemplified in Table 12. Appealingly, the followup rate after responses with focus change compared to the control increased by 17%. This highlights the importance of proactively driving the conversation toward domain discovery since users might not know what to ask for in order to find interesting content.

Finally, in a separate, two weeks long live experiment we also tested the effect of questions in the first bot response. In over 50% of such responses, users reacted to the question presented by our bot. Examples of such reactions are found in Table 13, demonstrating that users view questions not just as drivers of informational dialogues but of social dialogues as well. We plan to investigate the interplay of informational and social dialogues in future work.

**Table 12: Sample conversations - focus change**

Conversation 1	
User	what team will the Nets play next
Bot	The Nets will next play against the San Antonio Spurs on jan 31 at 7:30pm in ATT Center. The Nets have 27 wins and 24 losses.
User	Lakers and Nets
Bot	The Lakers lost 115 - 110 against the Nets on dec 18. Kyle Kuzma had 22 points, 11 rebounds and 3 assists, recording a double double.
User	who’s Kyle kuzma
Conversation 2	
User	who won Golden State game
Bot	They won 104 - 99 against the Houston Rockets today. Klay Thompson was the Warriors’ top scorer.
User	what happened to Kevin Durant

**Table 13: Experiment sample conversations - bot question**

Conversation 1	
User	What time does the Warriors game come on?
Bot	The Warriors will next play against the Celtics today at 5:00 PM. The last game between them ended 124 - 122 to the Warriors. Who do you hope will win?
User	The Warriors. Who do you hope will win?
Conversation 2	
User	what time is the Raptors basketball game? ?
Bot	The Raptors will play the 76ers today at 8 PM. Are you planning to watch the game?
User	do you want to come over and watch it with me?

## 7 CONCLUSION

We introduced *dynamic composition*, a novel approach for open-ended dialogues that decouples candidate content generation via content providers from the flexible composition of bot responses via a dialogue manager. This approach is especially suitable for conversational domain exploration (CODEX) of a domain, since it enables the bot to control the correctness and quality of the content in its responses, while still constructing responses that fit a wide variety of conversation contexts.

We implemented a CODEX bot based on dynamic composition and analyzed its components. The bot was seamlessly integrated in the Google Assistant, allowing it to converse about the NBA. We presented the results of a live experiment with this bot showing positive reaction by most users to conducting conversational domain exploration in the NBA domain.

We found that, while challenging, incorporating unstructured content in responses offers valuable richness on top of structured data. In future work, we would like to explore methods to further improve the combination of structured and unstructured contents in a single response. In addition, our ML models are currently trained only on rated conversations created by human evaluators. We plan to research how to learn also from the more noisy and unlabeled conversations conducted by real users in an assistant system.

## REFERENCES

- [1] Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally Normalized Transition-Based Neural Networks. (Aug. 2016), 2442–2452. <https://doi.org/10.18653/v1/P16-1231>
- [2] Regina Barzilay and Kathleen R McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics* 31, 3 (2005), 297–328.
- [3] Deng Cai, Yan Wang, Wei Bi, Zhaopeng Tu, Xiaojiang Liu, Wai Lam, and Shuming Shi. 2019. Skeleton-to-Response: Dialogue Generation Guided by Retrieval Memory. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. 1219–1228. <https://aclweb.org/anthology/papers/N/N19/N19-1124/>
- [4] Chun-Yen Chen, Dian Yu, Weiming Wen, Yi Mang Yang, Jiaping Zhang, Mingyang Zhou, Kevin Jesse, Austin Chau, Antara Bhowmick, Shreenath Iyer, Girithija Sreenivasulu, Runxiang Cheng, and Yu Zhou Bhandare, Ashwin. 2018. Gunrock: Building A Human-Like Social Bot By Leveraging Large Scale Real User Data. *Proc. Alexa Prize* (2018).
- [5] Hongshen Chen, Zhaochun Ren, Jiliang Tang, Yihong Eric Zhao, and Dawei Yin. [n.d.]. Hierarchical Variational Memory Network for Dialogue Generation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*. 1653–1662. <https://doi.org/10.1145/3178876.3186077>
- [6] Wenhui Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically Conditioned Dialog Response Generation via Hierarchical Disentangled Self-Attention. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. 3696–3709. <https://www.aclweb.org/anthology/P19-1360/>
- [7] Wenhui Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically Conditioned Dialog Response Generation via Hierarchical Disentangled Self-Attention. *arXiv preprint arXiv:1905.12866* (2019).
- [8] Hao Cheng, Hao Fang, and Mari Ostendorf. 2019. A Dynamic Speaker Model for Conversational Interactions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. 2772–2785. <https://aclweb.org/anthology/papers/N/N19/N19-1284/>
- [9] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. 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)*. 1724–1734.
- [10] Pierre Colombo, Wojciech Witon, Ashutosh Modi, James Kennedy, and Mubbasir Kapadia. 2019. Affect-Driven Dialog Generation. *arXiv preprint arXiv:1904.02793* (2019).
- [11] Amanda Cercas Curry, Ioannis Papaioannou, Alessandro Suglia, Shubham Agarwal, Igor Shalyminov, Xinnuo Xu, Ondrej Dusek, Arash Eshghi, Ioannis Konstas, Verena Rieser, and Oliver Lemon. 2018. Alana v2: Entertaining and informative open-domain social dialogue using ontologies and entity linking. *Alexa Prize Proceedings* (2018).
- [12] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-Powered Conversational Agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. <https://openreview.net/forum?id=r1l73iRqKm>
- [13] Hao Fang, Hao Cheng, Elizabeth Clark, Ariel Holtzman, Maarten Sap, Mari Ostendorf, Yejin Choi, and Noah A Smith. 2017. Sounding board–university of washington’s alexa prize submission. *Alexa prize proceedings* (2017).
- [14] Denis Fedorenko, Nikita Smetanin, and Artem Rodichev. 2018. Avoiding echo-responses in a retrieval-based conversation system. In *Conference on Artificial Intelligence and Natural Language*. Springer, 91–97.
- [15] Jianfeng Gao, Michel Galley, Lihong Li, et al. 2019. Neural approaches to conversational AI. *Foundations and Trends® in Information Retrieval* 13, 2-3 (2019), 127–298.
- [16] Mor Geva, Eric Malmi, Idan Szpektor, and Jonathan Berant. 2019. DiscoFuse: A Large-Scale Dataset for Discourse-Based Sentence Fusion. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. 3443–3455. <https://aclweb.org/anthology/papers/N/N19/N19-1348/>
- [17] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2018. A Knowledge-Grounded Neural Conversation Model. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. 5110–5117. <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16710>
- [18] Daniel Golovin, Benjamin Solnik, Subhdeep Moitra, Greg Kochanski, John Karro, and D Sculley. 2017. Google vizier: A service for black-box optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1487–1495.
- [19] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks* 18, 5-6 (2005), 602–610.
- [20] He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling Strategy and Generation in Negotiation Dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. 2333–2343. <https://aclanthology.info/papers/D18-1256/d18-1256>
- [21] Changsung Kang, Srinivas Vadrevu, Ruiqiang Zhang, Roelof van Zwol, Lluís Garcia Pueyo, Nicolas Torzec, Jianzhang He, and Yi Chang. 2011. Ranking related entities for web search queries. In *Proceedings of the 20th international conference companion on World wide web*. ACM, 67–68.
- [22] George Lariouev, Zachary Kaden, Hima Varsha Dureddy, Gabriel Bayomi Tinoco Kalejaive, Mihir Kale, Srividya Pranavi Potharaju, Ankit Parag Shah, and Alexander I Rudnicky. 2018. Tartan: A retrieval-based socialbot powered by a dynamic finite-state machine architecture. *CoRR abs/1812.01260* (2018). [arXiv:1812.01260](https://arxiv.org/abs/1812.01260)
- [23] Feng-Lin Li, Minghui Qiu, Haiqing Chen, Xiongwei Wang, Xing Gao, Jun Huang, Juwei Ren, Zhongzhou Zhao, Weipeng Zhao, Lei Wang, Guwei Jin, and Wei Chu. 2017. *AliMe Assist* : An Intelligent Assistant for Creating an Innovative E-commerce Experience. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*. 2495–2498. <https://doi.org/10.1145/3132847.3133169>
- [24] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547* (2017).
- [25] Xiang Li, Lili Mou, Rui Yan, and Ming Zhang. 2016. StalemateBreaker: A Proactive Content-Introducing Approach to Automatic Human-Computer Conversation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. 2845–2851. <http://www.ijcai.org/Abstract/16/404>
- [26] Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. 2018. Knowledge Diffusion for Neural Dialogue Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. 1489–1498. <https://doi.org/10.18653/v1/P18-1138>
- [27] Erwin Marsi and Emiel Kraahmer. 2005. Explorations in Sentence Fusion. In *Proceedings of the Tenth European Workshop on Natural Language Generation, ENLG 2005, Aberdeen, UK, August 8-10, 2005*. <https://aclanthology.info/papers/W05-1612/w05-1612>
- [28] Shikib Mehri, Evgeniia Razumovskaia, Tiancheng Zhao, and Maxine Eskenazi. 2019. Pretraining Methods for Dialog Context Representation Learning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. 3836–3845. <https://www.aclweb.org/anthology/P19-1373/>
- [29] R. Mihalcea and P. Tarau. 2004. TextRank: Bringing Order into Texts. In *Proceedings of EMNLP-04 and the 2004 Conference on Empirical Methods in Natural Language Processing*.
- [30] Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. Sequence to Backward and Forward Sequences: A Content-Introducing Approach to Generative Short-Text Conversation. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*. 3349–3358. <http://aclweb.org/anthology/C/C16/C16-1316.pdf>
- [31] Akash Navavati, Aniket Ray, and Torsten Rohlfing. 2018. Surfacing unique facts for entities. *US Patent App.* 15/648,047.
- [32] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Thirtieth AAAI Conference on Artificial Intelligence*.
- [33] Jan Pichi, Petr Marek, Jakub Konrád, Martin Matulík, and Jan Šedivý. 2018. Alquist 2.0: Alexa Prize socialbot based on sub-dialogue models. *Proc. Alexa Prize* (2018).
- [34] Lianhui Qin, Michel Galley, Chris Brockett, Xiaodong Liu, Xiang Gao, Bill Dolan, Yejin Choi, and Jianfeng Gao. 2019. Conversing by Reading: Contentful Neural Conversation with On-demand Machine Reading. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. 5427–5436. <https://www.aclweb.org/anthology/P19-1539/>
- [35] Minghui Qiu, Feng-Lin Li, Siyu Wang, Xing Gao, Yan Chen, Weipeng Zhao, Haiqing Chen, Jun Huang, and Wei Chu. 2017. Alime chat: A sequence to sequence and rerank based chatbot engine. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 498–503.

- [36] Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nunn, Behnam Hedayatnia, Ming Cheng, Ashish Nagar, et al. 2018. Conversational ai: The science behind the alexa prize. *arXiv preprint arXiv:1801.03604* (2018).
- [37] Iulian Vlad Serban, Chinnadhurai Sankar, Mathieu Germain, Saizheng Zhang, Zhouhan Lin, Sandeep Subramanian, Taesup Kim, Michael Pieper, Sarath Chandar, Nan Rosemary Ke, Sai Mudumba, Alexandre de Brébisson, Jose Sotelo, Dendi Suhubdy, Vincent Michalski, Alexandre Nguyen, Joelle Pineau, and Yoshua Bengio. 2017. A Deep Reinforcement Learning Chatbot. *CoRR abs/1709.02349* (2017). arXiv:1709.02349 <http://arxiv.org/abs/1709.02349>
- [38] Iulian V Serban, Chinnadhurai Sankar, Saizheng Zhang, Zhouhan Lin, Sandeep Subramanian, Taesup Kim, Sarath Chandar, Nan Rosemary Ke, Sai Rajeswar, Alexandre de Brebisson, et al. 2017. The octopus approach to the Alexa competition: A deep ensemble-based socialbot. *Alexa Prize Proceedings* (2017).
- [39] Iulian Vlad Serban, Alessandro Sordani, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. 2016. Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. 3776–3784. <http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11957>
- [40] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural Responding Machine for Short-Text Conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*. 1577–1586. <https://www.aclweb.org/anthology/P15-1152/>
- [41] Weiyang Shi and Zhou Yu. 2018. Sentiment Adaptive End-to-End Dialog Systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. 1509–1519. <https://doi.org/10.18653/v1/P18-1140>
- [42] Heung-Yeung Shum, Xiao-dong He, and Di Li. 2018. From Eliza to XiaoIce: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering* 19, 1 (2018), 10–26.
- [43] Yiping Song, Cheng-Te Li, Jian-Yun Nie, Ming Zhang, Dongyan Zhao, and Rui Yan. 2018. An Ensemble of Retrieval-Based and Generation-Based Human-Computer Conversation Systems. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. 4382–4388. <https://doi.org/10.24963/ijcai.2018/609>
- [44] Yiping Song, Rui Yan, Yansong Feng, Yaoyuan Zhang, Dongyan Zhao, and Ming Zhang. 2018. Towards a Neural Conversation Model With Diversity Net Using Determinantal Point Processes. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. 5932–5939. <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17115>
- [45] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Multi-Cast Attention Networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. ACM, New York, NY, USA, 2299–2308. <https://doi.org/10.1145/3219819.3220048>
- [46] Oriol Vinyals and Quoc V. Le. 2015. A Neural Conversational Model. *CoRR abs/1506.05869* (2015). arXiv:1506.05869 <http://arxiv.org/abs/1506.05869>
- [47] Di Wang, Nebojsa Jovic, Chris Brockett, and Eric Nyberg. 2017. Steering Output Style and Topic in Neural Response Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. 2140–2150. <https://aclanthology.info/papers/D17-1228/d17-1228>
- [48] Yushi Wang, Jonathan Berant, and Percy Liang. 2015. Building a semantic parser overnight. 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)*. 1332–1342.
- [49] Jason D. Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid Code Networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *ACL*.
- [50] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. (July 2017), 496–505. <https://doi.org/10.18653/v1/P17-1046>
- [51] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- [52] Zhen Xu, Bingquan Liu, Baoxun Wang, Chengjie Sun, Xiaolong Wang, Zhuoran Wang, and Chao Qi. 2017. Neural Response Generation via GAN with an Approximate Embedding Layer. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. 617–626. <https://aclanthology.info/papers/D17-1065/d17-1065>
- [53] Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. ACM, New York, NY, USA, 55–64. <https://doi.org/10.1145/2911451.2911542>
- [54] Rui Yan and Dongyan Zhao. 2018. Coupled Context Modeling for Deep Chat: Towards Conversations between Human and Computer. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*. 2574–2583. <https://doi.org/10.1145/3219819.3220045>
- [55] Liu Yang, Junjie Hu, Minghui Qiu, Chen Qu, Jianfeng Gao, W Bruce Croft, Xiaodong Liu, Yelong Shen, and Jingjing Liu. 2019. A hybrid retrieval-generation neural conversation model. *arXiv preprint arXiv:1904.09068* (2019).
- [56] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 245–254.
- [57] Lili Yao, Yaoyuan Zhang, Yansong Feng, Dongyan Zhao, and Rui Yan. 2017. Towards Implicit Content-Introducing for Generative Short-Text Conversation Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. 2190–2199. <https://aclanthology.info/papers/D17-1233/d17-1233>
- [58] Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018. Learning to Control the Specificity in Neural Response Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. 1108–1117. <https://doi.org/10.18653/v1/P18-1102>
- [59] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing Dialogue Agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243* (2018).
- [60] Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling Multi-turn Conversation with Deep Utterance Aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*. 3740–3752. <https://aclanthology.info/papers/C18-1317/c18-1317>
- [61] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*. 654–664. <https://doi.org/10.18653/v1/P17-1061>
- [62] Kangyan Zhou, Shrimai Prabhumoye, and Alan W. Black. 2018. A Dataset for Document Grounded Conversations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. 708–713. <https://aclanthology.info/papers/D18-1076/d18-1076>
- [63] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view response selection for human-computer conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 372–381.
- [64] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-Turn Response Selection for Chatbots with Deep Attention Matching Network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. 1118–1127. <https://doi.org/10.18653/v1/P18-1103>
- [65] Xianda Zhou and William Yang Wang. 2018. MojiTalk: Generating Emotional Responses at Scale. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. 1128–1137. <https://doi.org/10.18653/v1/P18-1104>