A Knowledge-Grounded Neural Conversation Model

Marjan Ghazvininejad[1*] **Chris Brockett**[2] **Ming-Wei Chang**[2]
Bill Dolan[2] **Jianfeng Gao**[2] **Wen-tau Yih**[2] **Michel Galley**[2]

[1]Information Sciences Institute, USC
[2]Microsoft Research
ghazvini@isi.edu, mgalley@microsoft.com

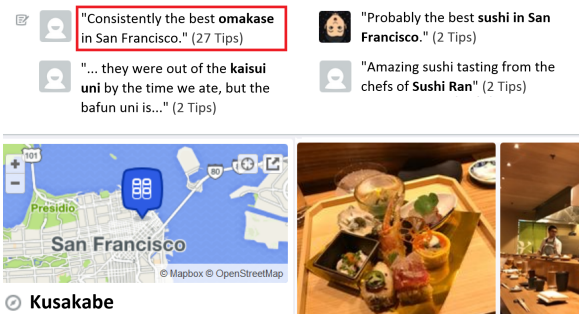
Abstract

Neural network models are capable of generating extremely natural sounding conversational interactions. Nevertheless, these models have yet to demonstrate that they can incorporate content in the form of factual information or entity-grounded opinion that would enable them to serve in more task-oriented conversational applications. This paper presents a novel, *fully* data-driven, and knowledge-grounded neural conversation model aimed at producing more contentful responses without slot filling. We generalize the widely-used SEQ2SEQ approach by conditioning responses on both conversation history and external "facts", allowing the model to be versatile and applicable in an open-domain setting. Our approach yields significant improvements over a competitive SEQ2SEQ baseline. Human judges found that our outputs are significantly more informative.

1 Introduction

Conversational agents such as Alexa, Siri, and Cortana have been increasingly popular, as they facilitate interaction between people and their devices. There is thus a growing need to build systems that can respond seamlessly and appropriately, and the task of conversational response generation has recently become an active area of research in natural language processing.

Recent work (Ritter et al., 2011; Sordoni et al., 2015; Shang et al., 2015; Vinyals and Le, 2015) has shown that it is possible to train conversational models in an end-to-end and completely data-driven fashion, without hand-coding. However, these fully data-driven systems lack grounding in the real world and do not have access to any external knowledge (textual or structured), which makes it difficult to respond substantively. Fig. 1 illustrates the difficulty: while an ideal response would directly reflect on the entities mentioned in the query (user input), neural models produce responses that, while conversationally appropriate, seldom include factual content. This contrasts with traditional dialog systems, which can easily inject entities and facts into responses using slot-filling, but often at the cost of significant hand-coding, making such systems difficult to scale to new domains or tasks.



User input: *Going to Kusakabe tonight.*
Neural model: Have a great time!
Human: You'll love it! Try omasake, the best in town.

Figure 1: Responses of fully data-driven conversation models are often appropriate, but generally lack content characteristic of human responses.

The goal of this paper is to benefit from both lines of research—fully data-driven and grounded in external knowledge. The tie to external data is critical, as the knowledge that is needed to make the conversation useful is often stored in non-conversational data, such as Wikipedia, books reviews on Goodreads, and restaurant reviews on Foursquare. While conversational agents can learn

* This work was conducted at Microsoft.

A: **Looking forward to trying** *@pizzalibretto* **tonight! my expectations are high**. B: **Get** *the rocco salad*. **Can you eat** *calamari*?
A: **Anyone in** *Chi* **have a** *dentist office* **they recommend? I'm never going back to** *[...]* **and would love a reco!** B: **Really looved** *Ora* **in** *Wicker Park*.
A: **I'm at** *California Academy of Sciences* B: **Make sure you catch** *the show* **at** *the Planetarium*. **Tickets are usually limited.**
A: **I'm at** *New Wave Cafe*. B: **Try to get to** *Dmitri's* **for dinner**. **Their** *pan fried scallops* **and** *shrimp scampi* **are to die for.**
A: **I just bought:** *[...] 4.3-inch portable GPS navigator* **for my wife, shh, don't tell her**. B: **I heard this brand** *loses battery power*.

Figure 2: Social media datasets include many contentful and useful exchanges, e.g., here recommendation dialog excerpts extracted from real tweets. While previous models (e.g., SEQ2SEQ) succeed in learning the **backbone of conversations**, they have difficulty modeling and producing *contentful words* such as named entities, which are sparsely represented in conversation data. To help solve this issue, we rely on non-conversational texts, which represent such entities much more exhaustively.

the backbone of human conversations from millions of conversations (Twitter, Reddit, etc.), we rely on non-conversational data to infuse relevant knowledge in conversation with users based on the context. More fundamentally, this line of research also targets more *useful* conversations. While prior data-driven conversational models have been essentially used as chitchat bots, access to external data can help users make better decisions (e.g., recommendation or QA systems) or accomplish specific tasks (e.g., task-completion agents).

This paper presents a novel, *fully* data-driven, and knowledge-grounded neural conversation model aimed at producing more contentful responses. It offers a framework that generalizes the SEQ2SEQ approach (Hochreiter and Schmidhuber, 1997; Sutskever et al., 2014) of most previous neural conversation models, as it naturally combines conversational and non-conversational data via multi-task learning (Caruana, 1998; Liu et al., 2015). The key idea of this approach is that it not only conditions responses based on conversation history (Sordoni et al., 2015), but also on external "facts" that are relevant to the current context (for example, Foursquare entries as in Fig. 1). Our approach only requires a way to ground external information based on conversation context (e.g., via simple entity name matching), which makes it highly versatile and applicable in an open-domain setting. This allowed us to train our system at a very large scale using 23M social media conversations and 1.1M Foursquare tips. The trained system showed significant improvements over a competitive large-scale SEQ2SEQ baseline. To the best of our knowledge, this is the first large-scale, fully data-driven neural conversation model that effectively exploits external knowledge, and it does so without explicit slot filling.

2 Grounded Response Generation

A primary challenge in building fully data-driven conversation models is that most of the world's knowledge is not represented in any existing conversational datasets. While these datasets (Serban et al., 2015) have grown dramatically in size thanks in particular to social media (Ritter et al., 2011), this data is still very far from containing discussions of every entry in Wikipedia, Foursquare, Goodreads, or IMDB. This problem considerably limits the appeal of existing data-driven conversation models, as they are bound to respond evasively or deflectively as in Fig. 1, especially with regard to those entities that are poorly represented in the conversational training data. On the other hand, even when such conversational data representing most entities of interest did exist, we would still face challenges as such huge dataset would be difficult to be used for model training, and many conversational patterns exhibited in the data (e.g., for similar entities) would be redundant.

Our approach aims to avoid redundancy and attempts to better generalize from existing conversational data, as illustrated in Fig. 2. While the conversations in the figure are about specific venues, products, and services, conversational patterns are general and equally applicable to other entities. The learned conversational behaviors could be used to, e.g., recommend other products and services. A traditional dialog system would use predefined slots to fill conversational backbone (bold text) with content; here, we present a more robust and scalable approach.

In order to infuse the response with factual information relevant to the conversational context,

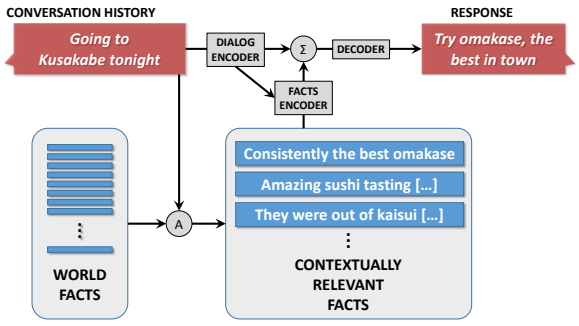


Figure 3: Knowledge-grounded model architecture.

we propose a knowledge-grounded model architecture depicted in Fig. 3. First, we have available a large collection of world facts,[1] which is a large collection of raw text entries (e.g., Foursquare, Wikipedia, or Amazon reviews) indexed by named entities as keys. Then, given a conversational history or source sequence S, we identify the "focus" in S, which is the text span (one or more entities) based on which we form a query to link to the facts. This focus can either be identified using keyword matching (e.g., a venue, city, or product name), or detected using more advanced methods such as entity linking or named entity recognition. The query is then used to retrieve all contextually relevant facts: $F = \{f_1, ..., f_k\}$.[2] Finally, both conversation history and relevant facts are fed into a neural architecture that features distinct encoders for conversation history and facts. We will detail this architecture in the subsections below.

This knowledge-grounded approach is more general than SEQ2SEQ response generation, as it avoids the need to learn the same conversational pattern for each distinct entity that we care about. In fact, even if a given entity (e.g., @pizzalibretto in Fig. 2) is not part of our conversational training data and therefore out-of-vocabulary, our approach is still able to rely on retrieved facts to generate an appropriate response. This also implies that we can enrich our system with new facts without the need to retrain the full system.

We train our system using multi-task learning (Luong et al., 2015) as a way of combining conversational data that is naturally associated with external data (e.g., discussions about restaurants and other businesses as in Fig. 2), and less informal exchanges (e.g., a response to *hi, how are you*). More specifically, our multi-task setup contains two types of tasks:

(1) one purely conversational, where we expose the model without fact encoder with (S, R) training examples, where S represents the conversation history and R is the response;
(2) the other task exposes the full model with $(\{f_1, \ldots, f_k, S\}, R)$ training examples.

This decoupling of the two training conditions offer several advantages, including: First, it allows us to pre-train the conversation-only dataset separately, and start multi-task training (warm start) with a dialog encoder and decoder that already learned the backbone of conversations. Second, it gives us the flexibility to expose different kind of conversational data in the two tasks. Finally, one interesting option is to replace the response in task (2) with one of the facts ($R = f_i$), which makes task (2) similar to an autoencoder and helps produce responses that are even more contentful. We will discuss the different ways we apply multi-task learning in practice in greater detail in Sec. 4.

2.1 Dialog Encoder and Decoder

The dialog encoder and response decoder form together a sequence-to-sequence (SEQ2SEQ model (Hochreiter and Schmidhuber, 1997; Sutskever et al., 2014), which has been sucessfully used in building end-to-end conversational systems (Sordoni et al., 2015; Vinyals and Le, 2015; Li et al., 2016a). Both encoder and decoder are recurrent neural network (RNN) models: an RNN that encodes a variable-length input string into a fixed-length vector representation and an RNN that decodes the vector representation into a variable-length output string. This part of our model is almost identical to prior conversational SEQ2SEQ models, except that we use gated recurrent units (GRU) (Chung et al., 2014) instead of LSTM (Hochreiter and Schmidhuber, 1997) cells. Encoders and decoders in sequence-to-sequence models sometimes share weights in monolingual tasks, but do not do so in the present model, nor do they share word embeddings.

2.2 Facts Encoder

The Facts Encoder of Fig. 3 is similar to the Memory Network model first proposed by (Weston et al., 2014; Sukhbaatar et al., 2015). It uses

[1] For presentation purposes, we refer to these items as "facts", but a "fact" here is simply any snippet of authored text, which may contain subjective or inaccurate information.

[2] In our work, we use a simple keyword-based IR engine to retrieve relevant facts from the full collection; more details are provided in Sec. 3.

an associative memory for modeling the facts relevant to a particular problem—in our case, an entity mentioned in a conversation–then retrieves and weights these facts based on the user input and conversation history to generate an answer. Memory network models are widely used in Question Answering to make inferences based on the facts saved in the memory (Weston et al., 2015).

In our adaptation of memory networks, we use an RNN encoder to turn the input sequence (conversation history) into a vector, instead of a bag of words representation as used in the original memory network models. This enables us to better exploit interlexical dependencies between different parts of the input, and makes this memory network model (facts encoder) more directly comparable to a SEQ2SEQ model.

More formally, we are given an input sentence $S = \{s_1, s_2, ..., s_n\}$, and a fact set $F = \{f_1, f_2, ..., f_k\}$ that are relevant to the conversation history. The RNN encoder reads the input string word by word and updates its hidden state. After reading the whole input sentence the hidden state of the RNN encoder, u is the summary of the input sentence. By using an RNN encoder, we have a rich representation for a source sentence.

Let us assume u is a d dimensional vector and r_i is the bag of words representation of f_i with dimension v. Based on (Sukhbaatar et al., 2015) we have:

$$m_i = Ar_i \quad (1)$$

$$c_i = Cr_i \quad (2)$$

$$p_i = \text{softmax}(u^T m_i) \quad (3)$$

$$o = \sum_{i=1}^{k} p_i c_i \quad (4)$$

$$\hat{u} = o + u \quad (5)$$

Where $A, C \in \mathbb{R}^{d \times v}$ are the parameters of the memory network. Then, unlike the original version of the memory network, we use an RNN decoder that is good for generating the response. The hidden state of the RNN is initialized with $\hat{u}$ which is a symmetrization of input sentence and the external facts, to predict the response sentence R word by word.

As alternatives to summing up facts and dialog encodings in equation 5, we also experimented with other operations such as concatenation, but summation seemed to yield the best results. The memory network model of (Weston et al., 2014) can be defined as a multi-layer structure. In this task, however, 1-layer memory network was used, since multi-hop induction was not needed.

3 Datasets

The approach we describe above is quite general, and is applicable to any dataset that allows us to map named entities to free-form text (e.g., Wikipedia, IMDB, TripAdvisor, etc.). For experimental purposes, we utilize datasets derived from two popular social media services: Twitter (conversational data) and Foursquare (non-conversational data). Note that none of the processing applied to these datasets is specific to any underlying task or domain.

3.1 Foursquare

Foursquare tips are comments left by customers about restaurants and other, usually commercial, establishments. A large proportion of these describe aspects of the establishment, and provide recommendations about what the customer enjoyed (or otherwise) We extracted from the web 1.1M tips relating to establishments in North America. This was achieved by identifying a set of 11 likely "foodie" cities and then collecting tip data associated with zipcodes near the city centers. While we targeted foodie cities, the dataset is very general and contains tips many types of local businesses (restaurants, theaters, museums, shopping, etc.) In the interests of manageability for experimental purposes, we ignored establishments associated with fewer than 10 tips, but other experiments with up to 50 tips per venue yield comparable results. We further limited the tips to those that for which Twitter handles were found in the Twitter conversation data.

3.2 Twitter

We collected a **23M general dataset** of 3-turn conversations. This serves as a background dataset not associated with facts, and its massive size is key to learning the conversational structure or backbone.

Separately, on the basis of Twitter handles found in the Foursquare tip data, we collected approximately 1 million two-turn conversations that contain entities that tie to Foursquare. We refer to this as the **1M grounded dataset**. Specifically, we identify conversation pairs in which the first turn contained either a handle of the business name (preceded by the "@" symbol) or a hashtag that

matched a handle.[3] Because we are interested in conversations among real users (as opposed to customer service agents), we removed conversations where the response was generated by a user with a handle found in the Foursquare data.

3.3 Grounded Conversation Datasets

We augment the 1M grounded dataset with facts (here Foursquare tips) relevant to each conversation history. The number of contextually relevant tips for some handles can sometimes be enormous, up to 10k. To filter them based on relevance to the input, the system uses tf-idf similarity between the input sentence and all of these tips and retains 10 tips with the highest score.

Furthermore, for a significant portion of the 1M Twitter conversations collected using handles found on Foursquare, the last turn was not particularly informative, e.g., when it provides a purely socializing response (e.g., *"have fun there"*) rather than a contentful one. As one of our goals is to evaluate conversational systems on their ability to produce *contentful* responses, we select a dev and test set (4k conversations in total) designed to contain responses that are informative and useful.

For each handle in our dataset we created two scoring functions:

- Perplexity according to a 1-gram LM trained on all the tips containing that handle.
- χ-square score, which measures how much content each token bears in relation to the handle. Each tweet is then scored on the basis of the average content score of its terms.

In this manner, we selected 15k top-ranked conversations using the LM score and 15k using the chi-square score. A further 15k conversations were randomly sampled. We then randomly sampled 10k conversations these data to be evaluated by crowdsourced annotators. Human judges were presented with the conversations and asked to determine whether the response contained actionable information, i.e., did they contain information that would permit the respondents to decide, e.g., whether or not they should patronize an establishment. From this, we selected the top-ranked 4k conversations to be held out validation set and test set; these were removed from our training data.

[3]This mechanism of linking conversations to facts using exact match on the handle is high precision but low recall, but low recall seems reasonable as we are far from exhausting all available Twitter and Foursquare data.

4 Experimental Setup

4.1 Multi-Task Learning

We use multi-task learning with these tasks:

- FACTS task: We expose the full model with $(\{f_1, ..., f_n, S\}, R)$ training examples.
- NOFACTS task: We expose the model without fact encoder with (S, R) examples.
- AUTOENCODER task: It is similar to the FACTS task, except that we replace the response with each of the facts, i.e., this model is trained on $(\{f_1, ..., f_n, S\}, f_i)$ examples. There are n times many samples for this task than for the FACTS task.[4]

The tasks FACTS and NOFACTS are representative of how our model is intended to work, but we found that the AUTOENCODER tasks helps inject more factual content into the response. Then, the different variants of our multi-task learned system exploits these tasks as follows:

- SEQ2SEQ: This system is trained on task NOFACTS with the 23M general conversation dataset. Since there is only one task, it is not *per se* a multi-task setting.
- MTASK: This system is trained on two instances of the NOFACTS task, respectively with the 23M general dataset and 1M grounded dataset (but without the facts). While not an interesting system in itself, we include it to assess the effect of multi-task learning separately from facts.
- MTASK-R: This system is trained on the NOFACTS task with the 23M dataset, and the FACTS task with the 1M grounded dataset.
- MTASK-F: This system is trained on the NOFACTS task with the 23M dataset, and the AUTOENCODER task with the 1M dataset.
- MTASK-RF: This system blends MTASK-F and MTASK-R, as it incorporates 3 tasks: NOFACTS with the 23M general dataset, FACTS with the 1M grounded dataset, and AUTOENCODER again with the 1M dataset.

We trained a one-layer memory network structure with two-layer SEQ2SEQ models. More specifically, we use 2-layer GRU models with 512 hidden cells for each layer is used for encoder and decoder, the dimensionality of word embeddings

[4]This is akin to an autoencoder (hence the name) as the fact f_i is represented both in the input and output, but of course not strictly an autoencoder.

is set to 512, and the size of input/output memory representation is 1024. We used the Adam optimizer with a fixed learning rate of 0.1, with a batch size is set to 128. All parameters are initialized from a uniform distribution in $[-\sqrt{3/d}, \sqrt{3/d}]$, where d is the dimension of the parameter. Gradients are clipped at 5 to avoid gradient explosion.

Encoder and decoder use different sets of parameters. The top 50k frequent types from conversation data is used as vocabulary which is shared between both conversation and non-conversation data. We use the same learning technique as (Luong et al., 2015) for multi-task learning. In each batch, all training data is sampled from one task only. For task i we define its mixing ratio value of α_i, and for each batch we select randomly a new task i with probability of $\alpha_i / \sum_j \alpha_j$ and train the system by its training data.

4.2 Decoding and Reranking

We use a beam-search decoder similar to (Sutskever et al., 2014) with beam size of 200, and maximum response length of 30. Following (Li et al., 2016a), we generate N-best lists containing three features: (1) the log-likelihood $\log P(R|S,F)$ according to the decoder; (2) word count; (3) the log-likelihood $\log P(S|R)$ of the source given the response. The third feature is added to deal with the issue of generating commonplace and generic responses such as "*I don't know*", which is discussed in details in (Li et al., 2016a). Our models often do not need the third feature to be effective, but—since our baseline needs it to avoid commonplace responses—we include this feature in all systems. This yields the following reranking score:

$$\log P(R|S,F) + \lambda \log P(S|R) + \gamma |R|$$

λ and γ are free parameters, which we tune on our development N-best lists using MERT (Och, 2003) by optimizing BLEU (Papineni et al., 2002a). To estimate $P(S|R)$ we train a Sequence-to-sequence model by swapping messages and responses. In this model we do not use any facts.

4.3 Evaluation Metrics

Following (Sordoni et al., 2015; Wen et al., 2016; Li et al., 2016a), we use BLEU automatic evaluation. While (Liu et al., 2016) suggest that BLEU correlates poorly with human judgment at the sentence-level,[5] we use instead corpus-level BLEU, which is known to better correlate with human judgments (Przybocki et al., 2008) including for response generation (Galley et al., 2015). We also report perplexity and lexical diversity, the latter as a raw yet automatic measure of informativeness and diversity. Automatic evaluation is augmented with human judgments of appropriateness and informativeness.

Model	Perplexity	
	General Data	Grounded Data
SEQ2SEQ	**55.0**	214.4
SEQ2SEQ-S	125.7	**82.6**
MTASK	57.2	82.5
MTASK-R	**55.1**	**77.6**
MTASK-F	77.3	448.8
MTASK-RF	67.2	97.7

Table 1: Perplexity of different models. SEQ2SEQ-S is a SEQ2SEQ model that is trained on NOFACTS task with 1M grounded dataset (without the facts).

Model	BLEU	Diversity	
		1-gram	2-gram
SEQ2SEQ	0.55	4.14%	14.4%
MTASK	0.79	2.34%	5.9%
MTASK-F	0.38	8.35%	23.1%
MTASK-R	**1.08**	7.08%	21.9%
MTASK-RF	0.58	**8.71%**	**26.0%**

Table 2: BLEU-4 and lexical diversity.

5 Results

Automatic Evaluation: We computed perplexity and BLEU (Papineni et al., 2002b) for each system. These are shown in Tables 1 and 2 respectively. We observe that the perplexity of MTASK and MTASK-R models on both general and grounded data is as low as the SEQ2SEQ models that are trained specifically on general and grounded data respectively. As expected, injecting more factual content into the response in MTASK-F and MTASK-RF increased the perplexity especially on grounded data.

BLEU scores are low, but this is not untypical of conversational systems (e.g., (Li et al., 2016a,b)). Table 2 shows that the MTASK-R model yields a significant performance boost, with a BLEU score increase of 96% and 71% jump in 1-gram diversity compared to the competitive SEQ2SEQ baseline. In terms of BLEU scores, MTASK-RF improvements is not significant, but it generates the highest

[5] This corroborates earlier findings that accurate sentence-level automatic evaluation is indeed difficult, even for Machine Translation (Graham et al., 2015), as BLEU and related metrics were originally designed as corpus-level metrics.

1-gram and 2-gram diversity among all models.

Human Evaluation: We conducted human evaluations using a crowdsourcing service. We had annotators judge 500 paired conversations, asking which is better on two parameters: appropriateness to the topic, and informativeness. Seven judges were assigned to each pair. Annotators whose variance fell greater than two standard deviations from the mean variance were dropped. Ties were permitted.

The results of annotation are show in Table 3. On *Appropriateness*, no system performed significantly better than baseline, and in two cases, MTASK and MTASK-F, the baseline system was significantly better. MTASK-R appears to be slightly better than baseline in the table, but the difference is small and not statistically significant by conventional standards of $\alpha = 0.05$. On *Informativeness*, MTASK-F and MTASK-R perform significantly better than Baseline (p = 0.005 and p = 0.003 respectively). Since the baseline system outperforms MTASK-F with respect to *Appropriateness*, this may mean that MTASK-F encounters difficulty representing the social dimensions of conversation, but is strong on informational content. MTASK-R, on the other hand, appears to hold its own on *Appropriateness* while improving with respect to *Informativeness*.

The narrow differences in averages in Table 3 tend to obfuscate the judges' voting trends. Accordingly, we translated the scores for each output into the ratio of judges who preferred that system and binned their counts. The results are shown in Figs. 4 and 5 where we compare MTASK-R with the SEQ2SEQ baseline. Bin 7 on the left corresponds to the case where all 7 judges "voted" for the system, bin 6 to that where 6 out of 7 judges "voted" for the system, and so on.[6] Other bins are not shown since these are a mirror image of bins 7 through 4. The distributions in Fig. 4 are more similar to each other than in Fig. 5. indicating that judge preference for the MTASK-R model is relatively stronger with regard to informativeness.

6 Discussion

Figure 6 presents examples of outputs generated by MTASK-RF model. It illustrates that responses of our model are generally not only generally adequate, but also more informative and useful.

[6]Partial scores were rounded up. This affects both systems equally.

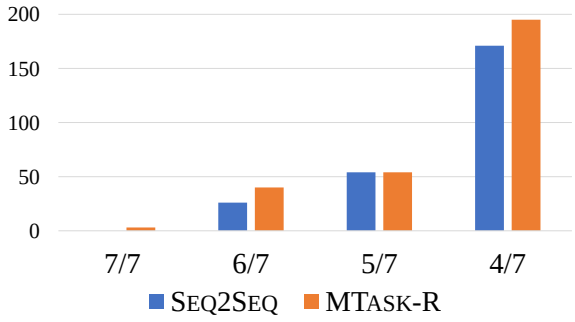


Figure 4: Judge preference counts (appropriateness) for MTASK-R versus SEQ2SEQ.

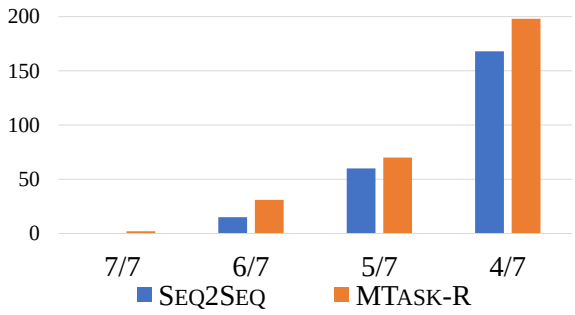


Figure 5: Judge preference counts (informativeness) for MTASK-R versus SEQ2SEQ.

For example the first response combines "*have a safe flight*", which is safe and appropriate and as such typical of existing neural conversational systems, but also "*nice airport terminal*", which is grounded in knowledge about the specific airport. While our model sometimes regurgitates fragments of tips in its responses, it often blends together information from various tips and the conversational in order to produce a response, such as in the the 5th and last responses of the figure. The 5th is mainly influenced by two Foursquare tips[7] and the model creates a fusion of the two, a kind of text manipulation that would be difficult with slot filling.

7 Related Work

The present work extends the data-driven paradigm of conversation generation by injecting knowledge from textual data into models derived from conversational data. This paradigm was introduced by Ritter et al. (2011) who first proposed using statistical Machine Translation models to generate conversational responses from social media data. It has been was further advanced by the introduction of neural network

[7]("*sit with and take a picture of the Simpsons on the 3rd floor*") and ("*Check out the video feed on 6 and Simpsons/billiards on 3!*").

Comparison	Appropriateness				Informativeness			
SEQ2SEQ vs MTASK	**0.530**	**±0.017**	0.470	±0.017	0.501	±0.016	0.499	±0.015
SEQ2SEQ vs MTASK-F	**0.537**	**±0.016**	0.463	±0.017	0.478	±0.015	**0.522**	**±0.015**
SEQ2SEQ vs MTASK-R	0.495	±0.015	0.505	±0.015	0.479	±0.017	**0.521**	**±0.013**
SEQ2SEQ vs MTASK-RF	0.502	±0.015	0.498	±0.014	0.492	±0.013	0.508	±0.013

Table 3: Mean differences in judgments in crowdsourced human evaluation, together with 95% confidence intervals. Differences sum to 1.0. Boldface items are significantly better ($p < 0.05$) than their comparator.

A: Visiting the celebs at Los Angeles International Airport (LAX) - [...] w/ 70 others B: Nice airport terminal. Have a safe flight.
A: Is that [...] in your photos? It's on my list of places to visit in NYC. B: Don't forget to check out the 5th floor, while you are here, it's a great view.
A: Live right now on [...] Tune in!!!!! B: Listen to Lisa Paige
A: Been craving Chicken Pot Pie-who has the best? Trying [...] at [...] Must be Change of weather! B: Love the pasta trattoria.
A: So [...] is down to one copy of Pound Foolish. I'm curious to see if they are re-ordering it. B: Check out the video feed on 6 and take a picture of the Simpsons on the 3rd floor.
A: I wish [...] would introduce another vegetarian option besides the shroomburger. It's delicious but kind of ridiculous. B: This is the best j.crew in the world. Try the lemonade!
A: Just had an awesome dinner at [...] Great recommendation [...] B: One of my favorite places I've ever been to in NYC. The food is great and the service is lackluster.

Figure 6: Responses produced by our knowledge-grounded neural response generation system (MTASK-RF). Entities marked with [...] have been anonymized to avoid (potentially negative) publicity.

models (Sordoni et al., 2015; Serban et al., 2016; Shang et al., 2015; Vinyals and Le, 2015; Li et al., 2016a,b). The introduction of contextual models by (Sordoni et al., 2015) is an important advance; we build on this by incorporating context from outside the conversation.

This work distinguishes itself from a second paradigm of neural dialog modeling in which question answer slots are explicitly learned from small amounts of crowd-sourced data or customer support logs (Wen et al., 2015, 2016). In many respects, this second paradigm can be characterized as an extension of conventional dialog models with or without statistical modelling, e.g., (Oh and Rudnicky, 2000; Ratnaparkhi, 2002; Banchs and Li, 2012; Ameixa et al., 2014; Nio et al., 2014).

Relevant to the current work is (Bordes and Weston, 2016), who employ memory networks to handle restaurant reservations, using a small number of keywords to handle entity types in a knowledge base (cuisine type, location, price range, party size, rating, phone number and address). That approach requires a highly structured knowledge base, whereas we are attempting to leverage free-form text using a highly scalable approach in order to learning implicit slots.

8 Conclusions

We have presented a novel knowledge-grounded conversation engine that could serve as the core component of a multi-turn recommendation or conversational QA system. The model is a large-scale, scalable, fully data-driven neural conversation model that effectively exploits external knowledge, and does so without explicit slot filling. It generalizes the SEQ2SEQ approach to neural conversation models by naturally combining conversational and non-conversational data through multi-task learning. Our simple entity matching approach to grounding external information based on conversation context makes for a model that is informative, versatile and applicable in open-domain systems.

Acknowledgments

We thank Xuetao Yin and Leon Xu for helping us extract Foursquare data. We also thank Kevin Knight, Chris Quirk, Nebojsa Jojic, Lucy Vanderwende, Vighnesh Shiv, Yi Luan, John Wieting, Alan Ritter, Donald Brinkman, and Puneet Agrawal for helpful suggestions and discussions.

References

David Ameixa, Luisa Coheur, Pedro Fialho, and Paulo Quaresma. 2014. Luke, i am your father: dealing with out-of-domain requests by using movies subtitles. In *Intelligent Virtual Agents*. Springer, pages 13–21.

Rafael E Banchs and Haizhou Li. 2012. Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the ACL 2012 System Demonstrations*. Association for Computational Linguistics, pages 37–42.

Antoine Bordes and Jason Weston. 2016. Learning end-to-end goal-oriented dialog. *CoRR* abs/1605.07683.

Rich Caruana. 1998. Multitask learning. In *Learning to learn*, Springer, pages 95–133.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR* abs/1412.3555.

Michel Galley, Chris Brockett, Alessandro Sordoni, Yangfeng Ji, Michael Auli, Chris Quirk, Margaret Mitchell, Jianfeng Gao, and Bill Dolan. 2015. deltableu: A discriminative metric for generation tasks with intrinsically diverse targets. In *Proc. of ACL-IJCNLP*.

Yvette Graham, Timothy Baldwin, and Nitika Mathur. 2015. Accurate evaluation of segment-level machine translation metrics. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Denver, Colorado, pages 1183–1191.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proc. of NAACL-HLT*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proc. of ACL*.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *CoRR* abs/1603.08023.

Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. 2015. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado, pages 912–921.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114* .

Lasguido Nio, Sakriani Sakti, Graham Neubig, Tomoki Toda, Mirna Adriani, and Satoshi Nakamura. 2014. Developing non-goal dialog system based on examples of drama television. In *Natural Interaction with Robots, Knowbots and Smartphones*, Springer, pages 355–361.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, pages 160–167.

Alice H Oh and Alexander I Rudnicky. 2000. Stochastic language generation for spoken dialogue systems. In *Proceedings of the 2000 ANLP/NAACL Workshop on Conversational systems-Volume 3*. Association for Computational Linguistics, pages 27–32.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002a. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 311–318.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002b. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL*. pages 311–318.

M. Przybocki, K. Peterson, and S. Bronsart. 2008. Official results of the nist 2008 metrics for machine translation challenge. In *MetricsMATR08 workshop*. http://itl.nist.gov/iad/mig/tests/metricsmatr/2008/.

Adwait Ratnaparkhi. 2002. Trainable approaches to surface natural language generation and their application to conversational dialog systems. *Computer Speech & Language* 16(3):435–455.

Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pages 583–593.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proc. of AAAI*.

Iulian Vlad Serban, Ryan Lowe, Laurent Charlin, and Joelle Pineau. 2015. A survey of available corpora for building data-driven dialogue systems. *CoRR* abs/1512.05742.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *ACL-IJCNLP*. pages 1577–1586.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proc. of NAACL-HLT*.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*. pages 2440–2448.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems (NIPS)*. pages 3104–3112.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In *Proc. of ICML Deep Learning Workshop*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei-hao Su, David Vandyke, and Steve J. Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. *CoRR* abs/1603.01232.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proc. of EMNLP*. Association for Computational Linguistics, Lisbon, Portugal, pages 1711–1721.

Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. 2015. Towards AI-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698* .

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916* .