Personalization in Goal-Oriented Dialog

Chaitanya K. Joshi, Fei Mi & Boi Faltings
Artificial Intelligence Laboratory
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
{chaitanya.joshi, fei.mi, boi.faltings}@epfl.ch
June 9, 2017

Abstract
The main goal of modelling human conversation is to create agents which can interact with people in both open-ended and goal-oriented scenarios. End-to-end trained neural dialog systems are an important line of research for such generalized dialog models as they do not resort to any situation-specific handcrafting of rules. Modelling personalization of conversation in such agents is important for them to be truly ‘smart’ and to integrate seamlessly into the lives of human beings. However, the topic has been largely unexplored by researchers as there are no existing corpora for training dialog systems on conversations that are influenced by the profiles of the speakers involved. In this paper, we present a new dataset of goal-oriented dialogs with profiles attached to them. We also introduce a framework for analyzing how systems model personalization in addition to performing the task associated with each dialog. Although no existing model was able to sufficiently solve our tasks, we provide baselines using a variety of learning methods and investigate in detail the shortcomings of an end-to-end dialog system based on Memory Networks. Our dataset and experimental code are publicly available at https://github.com/chaitjo/personalized-dialog.

1 Introduction
The recent advances in memory and attention mechanisms for neural networks architectures have led to remarkable progress in machine translation (Bahdanau et al., 2014; Johnson et al., 2016), question answering (Sukhbaatar et al. 2015, Graves et al., 2016) and other language understanding tasks which require an element of logical reasoning. The main motivation for building neural network based systems over traditional systems for such tasks is that they do not require any feature engineering or domain-specific handcrafting of rules (Vinyals and Le, 2015). Conversation modelling is one such domain where end-to-end trained systems have matched or surpassed traditional dialog systems in both open-ended (Dodge et al., 2016) and goal-oriented applications (Bordes et al., 2017).

An important yet unexplored aspect of dialog systems is the ability to personalize the bot’s responses based on the profile or attributes of who it is interacting with (Serban et al., 2017). For example, a restaurant reservation system should ideally conduct dialog with the user to find values for variables such
Figure 1: **Original bAbI dialog tasks.** The user (in green) conducts a dialog with the bot (in blue) to reserve a table at a restaurant. At each turn, a model has access to the conversation history and the outputs from the API call (in light red) and must predict the next bot utterance or API call (in dark red). (Illustration taken from Bordes et al., 2017)

as location, type of cuisine and price range. It should then make recommendations based on these variables as well as certain fixed attributes about the user (dietary preference, favorite food items, etc.). The register (or style) of the language used by the bot may also be influenced by certain characteristics of the user (age, gender, etc.) (Halliday et al., 1964). However, there are no open datasets which allow researchers to train end-to-end dialog systems where each conversation is influenced by a speaker’s profile (Serban et al., 2017).

With the ultimate aim of creating such a dataset, this paper aims to be an extension of the bAbI dialog dataset introduced by Bordes et al. (2017). Set in the domain of restaurant reservation, their synthetically generated dataset breaks down a conversation into several tasks to test some crucial capabilities that dialog systems should have. Taken together, the tasks can be used as a framework for the analysis of end-to-end dialog systems in a goal-oriented setting. Given a knowledge base (KB) of restaurants and their properties (location, type of cuisine, etc.), the aim of the dialog is to book a restaurant for the user. Full dialogs are divided into various stages, each of which tests if models can learn abilities such as
implicit dialog state tracking, using KB facts in dialog, and dealing with new entities not appearing in
dialogs from the training set.

In this paper, we propose extensions to the first five tasks of the existing dataset. In addition to the goal of
the original task, the dialog system must leverage a user’s profile information to alter speech style and
personalize reasoning over the KB. The end-goal is to make a restaurant reservation that is personalized to
the user’s attributes (dietary preference, favorite food items, etc.).

The synthetic nature of the bAbI dialog dataset and by extension, our work, makes it easy to construct a
perfect handcrafted dialog system based on the same rules that were used to generate the dialogs. Hence,
the goal here is not to improve the state of the art in this domain, but to analyze existing end-to-end
goal-oriented dialog systems and to model personalization in such frameworks without handcrafting.
Section 3 presents our modifications to the original dataset and Section 4 describes the various models
that are trained on our tasks. Experimental results (Section 5) show that Memory Networks are an
efficient model for goal-oriented dialog but are not able to reason over a KB or personalize conversation
perfectly. Further work needs to be done on these aspects of end-to-end models in order to develop
reliable systems for personalization in goal-oriented dialog.

2 Related Work

This work builds upon the bAbI dialog dataset described in Bordes et al. (2017), which is aimed at testing
end-to-end dialog systems in the goal-oriented domain of restaurant reservations. Their tasks are meant to
complement the bAbI tasks for text understanding and reasoning described in Weston et al. (2015b). In
addition to the baselines provided, several new techniques proposed to solve these tasks are of interest to
us (Perez and Liu, 2017; Seo et al., 2017; Eric and Manning, 2017).

The closest work to ours is by Li et al. (2016b) who encoded speaker personas into SEQ2SEQ dialog
models (Vinyals and Le, 2015; Li et al., 2016a). The model builds an embedding of a speaker’s persona in
a vector space based on conversation history of the speaker (for example, all of their tweets).

Our work differs from their investigation in the sense that 1) they are concerned with modelling speaker
personas for solving the problem of consistent response generation in an open-ended dialog (chit-chat),
whereas our work focuses on the personalization of a goal-oriented conversation and the ranking/discrimination of the correct candidate responses from a set of utterances, and 2) their dialog
system needs to be provided with a speaker’s conversation history to build the persona, while the user’s
attributes are explicitly provided to the bot for our task. Thus, models must compose user profiles through
the possible values of each attribute. This is arguably a better representation of real-world learning
scenarios where goal-oriented dialog agents can leverage information stored in databases to personalize
conversations in domains such as customer care or restaurant reservation.

3 Personalized Goal-Oriented Dialog Tasks

We build upon the first five synthetically generated bAbI dialog tasks (T1-T5), where the goal is to book
a table at a restaurant. The conversations are generated by a simulator (in the format shown in Figure 1)
based on an underlying KB containing all the restaurants and their properties. Each restaurant is defined
by a type of cuisine (10 choices, e.g., Italian, Indian), a location (10 choices, e.g. London, Tokyo), a price
range (cheap, moderate or expensive), a party size (2, 4, 6 or 8 people) and a rating (from 1 to 8). Each
restaurant also has an address and a phone number. Making an API call to the KB returns a list of facts related to all the restaurants that satisfy the four parameters: location, cuisine, price range and party size.

In addition to the user and bot utterances, dialogs in each task are comprised of API calls and the resulting facts. Conversations are generated using natural language patterns after randomly selecting each of the four required fields: location, cuisine, price range and party size. There are 43 patterns for the user and 15 for the bot (the user can say something in up to 4 different ways, while the bot only has one).

We make further additions to the KB and augment the bot utterance patterns for the creation of our tasks. In addition to fulfilling the original goal, the modified tasks also require the dialog system to personalize the conversation based on the user’s profile, which is composed of various fixed attributes. To fit in with the synthetic nature of the bAbI dialog tasks, the personalization of the bot’s speech style and KB reasoning are handcrafted to be extremely simplistic in comparison to real life situations. The details of our enhancements are described in the following sections.

3.1 Original Tasks

Tasks 1 and 2 test the model’s capability to implicitly track dialog state, Tasks 3 and 4 check if they can use KB facts in conversation. Task 3 also involves sorting through candidates and providing suggestions. Task 5 combines all tasks into a full dialog.

Task 1: Issuing API calls Users define a query containing from 0 to 4 of the required fields (sampled uniformly). The bot must ask questions to fill the missing fields and then generate the proper API call.

Task 2: Updating API calls Starting by issuing an API call as in Task 1, users then ask to update their requests between 1 and 4 times (sampled uniformly). The fields to update are selected randomly and the bot must then issue the updated API call.

Task 3: Displaying Options Given a user request, the KB is queried by the corresponding API call and the resulting facts are added to the dialog history. The bot must sort the restaurants in the facts based on their ratings (from higher to lower) and propose a restaurant to the users until they accept. Users accept a suggestion 25% of the time or always if it is the last remaining one.

Task 4: Providing extra information Given a user request for a randomly sampled restaurant, all KB facts related to the restaurant are added to the history and the dialog is conducted as if the user has agreed to book a table there. The user then asks for the address of the restaurant, its phone number or both (with probabilities 25%, 25% and 50% respectively). The bot must learn to retrieve the correct KB fact from history.

Task 5: Conducting full dialogs Conversations generated for Task 5 combine all the aspects of Tasks 1-4 into full dialogs.

3.2 User Profiles and Speech Style Changes

The first aspect of personalization incorporated into all 5 tasks was the change in the style of the language used by the bot based on the user’s gender (male or female) and age (young, middle-aged or elderly). For each of the 15 bot utterance patterns in the original tasks, we created 6 new patterns for each (age, gender)
Table 1: **Modified bot utterances for each profile.** Column 2 shows excerpts from a dialog from the bAbI dialog tasks and Columns 3-8 show the same excerpts after applying the proposed speech style changes for each profile. We do not modify the user’s utterances. The ellipses represent KB entities.

| Profile (Modified tasks) | male, young | female, young | male, middle-aged | female, middle-aged | male, elderly | female, elderly |
|-------------------------|-------------|---------------|-------------------|---------------------|---------------|---------------|
| Bot                     | what do you think of this option: ... | is this one cool: ... | how about this one: ... | is this a good option: ... | what do you think of this option: ... | may i suggest this option: ... | would you consider this option: ... |
| User                    | do you have something else | ok looking for something else | sure finding something else | ok i'll look for a better option | sure i'll find a better option | yes sir i will look for another suitable option | yes maam i shall find another suitable option |

profile permutation. Each of these patterns, while conveying the same information to the user, differed in tone, formality and word usage. Appendix A displays in a tabulated form all the original patterns and the 6 modified patterns associated with each of them.

While creating the utterance patterns, importance was given to maintaining a consistent vocabulary for each of the 6 profiles. The levels of formality and precision of the words and language used by the bot increased with the age of the user. At the same time, word choices overlapped between the same gender and age group. For example, for a given bot utterance, the pattern for a (female, young) user is similar in formality and tone to the pattern for a (male, young) user and shares certain key words with both (male, young) and (female, middle-aged) user patterns. It is comparatively unrelated to the patterns of a (male, middle-aged) or (male, elderly) user. By creating such relationships between the profiles instead of having 6 completely distinct patterns, we wanted to test whether dialog models could learn to form associations between concepts such as formality, precision and word usage, and attributes such as gender and age.

Applying our speech style changes to the bAbI dialog tasks, we obtained 6 versions of the same dialog associated with each user profile. Hence, we increased the size of each task by 6 folds.

### 3.3 KB Updates and Personalized Reasoning

The second aspect of personalization was restricted to Tasks 3 and 4, which involve reasoning over KB facts.

To personalize the order in which restaurants are recommended by the bot in Task 3, we added 2 new attributes to the user’s profile- dietary preference (vegetarian or non-vegetarian) and favorite food item (randomly sampled from a list of dishes associated with the cuisine in the dialog). We created a duplicate for each restaurant in the KB, with an additional attribute for type of restaurant (vegetarian or non-vegetarian) to differentiate the otherwise same copies. For every restaurant in the modified KB, we also added the speciality attribute (randomly sampled from a list of dishes associated with the restaurant’s cuisine). When modifying each dialog in the original task, instead of sorting and proposing the restaurants solely on their rating, we used a score calculated as: rating (out of 8) + 8 (if restaurant type matches user’s dietary preference) + 2.5 (if restaurant speciality matches user’s favorite food item). With such a metric, a
Table 2: Personalized reasoning over the KB. Columns 2 and 3 show excerpts from variants of the same dialog for two different user profiles. Row 2 contains the KB facts resulting from the same API call for both profiles. The bot’s first suggestion is a vegetarian restaurant for the vegetarian user and a non-vegetarian one for the non-vegetarian user. In column 3, the bot gives priority to a lower rated restaurant specializing in pizza as the user’s favorite food item is also pizza. Finally, the bot provides a social media link to the young user and the phone number to the elderly user when they ask for contact information.

| Locutor | Profile |
|----------|---------|
| KB Facts | resto_paris_moderate_italian_6stars_2: R_rating 6, R_type non-veg, R_speciality pasta<br>resto_paris_moderate_italian_5stars_2: R_rating 5, R_type non-veg, R_speciality risotto<br>resto_paris_moderate_italian_6stars_1: R_rating 6, R_type veg, R_speciality pasta<br>resto_paris_moderate_italian_5stars_1: R_rating 5, R_type veg, R_speciality pizza |
| Bot | is this one cool: resto_paris_moderate_italian_6stars_2<br>may i suggest this option: resto_paris_moderate_italian_5stars_1 |
| User | that looks great<br>what are the contact details of the restaurant |
| Bot | cool its done<br>here you go<br>resto_paris_moderate_italian_6stars_2_social_media |
| User | excellent i will finalize your request<br>here is the information you asked for<br>resto_paris_moderate_italian_5stars_1_phone |

vegetarian user will always be proposed all the vegetarian restaurants in the KB facts (in descending order of rating) before the bot suggests any non-vegetarian ones. Also, if a user’s favorite item is pizza, a 6 star rated restaurant specializing in pizza will always be proposed before an 8 star rated restaurant specializing in pasta or risotto. This tests a model’s ability to perform true/false reasoning based on the user’s profile and implicitly rank restaurants depending on more than one condition.

Our modification to Task 4 requires the bot to retrieve a combination of KB facts related to a restaurant based on certain attributes of the user and the restaurant itself. In addition to the phone number and address, we added 3 new attributes (social media links, parking information and public transport information) to the KB entries for every restaurant. In each modified dialog, when a user asks for the contact information of the restaurant, the bot must return the restaurant’s social media link if the user is young, or the phone number if the user is middle-aged or elderly. Similarly, when a user asks for the directions to the restaurant, the bot must return the address and the public transport information if the restaurant is cheap, or the address and the parking information if it is in the moderate or expensive price range. This tests a model’s ability to personalize KB fact retrieval based on an attribute in the user’s profile (age) or a choice made by the user during the dialog (the restaurant’s price range).

3.4 Updated Dataset

We applied our proposed modifications to bAbI dialog tasks T1-T5 to create the personalized dialog tasks PT1-PT5. In addition to the original goals of each task, all the modified tasks require the bot to personalise its speech style based on the user’s profile. Additionally, PT3 and PT4 also test the bot’s ability to personalize reasoning over a KB. PT5 combines all aspects of PT1-PT4 into full dialogs. For each dialog in all tasks, the values of the attributes of the user’s profile (gender, age, dietary preference and favorite food item) are provided before the first turn of the dialog.
Table 3: Dataset statistics. For the rows showing sizes of training, validation and test sets for each of the tasks, the first number is the size of the full set and the number in parenthesis is the size of the small set. (*) PT1-PT5 and all 5 bAbI dialog tasks have two test sets of the same size, one using the same KB entities as the training set and the other using out-of-vocabulary words.

| Task                        | PT1  | PT2  | PT3  | PT4  | PT5  | bAbI dialog tasks (T1-T5) |
|-----------------------------|------|------|------|------|------|---------------------------|
| Longest sentence length     | 21   | 21   | 21   | 11   | 19   |                           |
| Longest candidate sentence length | 13   | 13   | 13   | 13   | 13   |                           |
| Longest story length        | 15   | 21   | 201  | 19   | 219  |                           |
| Average story length        | 6    | 9    | 96   | 15   | 62   |                           |
| Vocabulary size             | 14819|      |      |      | 3747 |                           |
| Candidate set size          | 43863|      |      |      | 4212 |                           |
| Training dialogs            | 6000 (1000) | 6000 (1000) | 12000 (1000) | 6000 (1000) | 12000 (1000) | 1000 each |
| Validation dialog           | 6000 (1000) | 6000 (1000) | 12000 (1000) | 6000 (1000) | 12000 (1000) | 1000 each |
| Test dialogs                | 6000* (1000*) | 6000* (1000*) | 12000* (1000*) | 6000* (1000*) | 12000* (1000*) | 1000* each |

We generated and structured the dataset in the same way as the original bAbI dialog dataset- for each task, we provided training, validation and test set dialogs generated using half of the modified KB. We also generated another test set from the remaining KB containing new entities (restaurants, locations, cuisine types, etc.) unseen in any training dialog, called Out-Of-Vocabulary (OOV) test set. During training, the model has access to training examples and the KB. Models are evaluated on both test sets, plain and OOV, on their ability to rank the correct bot utterance at each turn of the dialog from a list of all possible candidates.

The statistics of the datasets for each task are given in Table 3, along with a comparison to the original bAbI dialog tasks. The size of the vocabulary has increased by almost four folds due to the various speech styles associated with user profiles. The number of possible candidate responses has increased ten fold due to the duplication of each restaurant in the KB and the speech style changes. We provide two variations for each task- a full set with all generated dialogs and a small set with only 1000 dialogs each for training, validation and testing to create realistic learning conditions.

4 Models

Following Bordes et al. (2017), we provide baselines on the modified dataset by evaluating several learning methods: rule-based systems, supervised embeddings, and end-to-end Memory networks.
4.1 Rule-Based Systems
Our tasks are generated by modifying and appending to the bAbI dialog tasks T1-T5. All dialogs are built with a rule based simulator and the bot utterances are completely deterministic. Thus, it is possible to create a perfect handcrafted system based on the same rules as the simulator, similar to the bAbI QA tasks of Weston et al. (2015b). As mentioned previously, the point of the tasks is not to improve the state of the art in restaurant reservation through handcrafted systems, but to analyze the strengths and weaknesses of machine learning algorithms.

4.2 Supervised Embedding Models
Although widely known for learning unsupervised embeddings from raw text like in Word2Vec (Mikolov et al., 2013), embeddings can also be learned in a supervised manner specifically for a given task. Supervised word embedding models which score (conversation history, response) pairs have been shown to be a strong baseline for both open-ended and goal-oriented dialog (Dodge et al., 2016; Bordes et al., 2017). We do not handcraft any special embeddings for the user profiles.

The embedding vectors are trained specifically for the task of predicting the next response given the previous conversation: a candidate response \( y \) is scored against the input \( x \): \( f(x, y) = (Ax)^T By \), where \( A \) and \( B \) are \( d \times V \) word embedding matrices and input and response are treated as summed bag-of-embeddings. The model is trained with SGD to minimize a margin ranking loss: \( f(x, y) > m + f(x, y') \), where \( m \) is the size of the margin and \( N \) negative candidate responses \( y' \) are sampled per example.

4.3 Memory Networks
Memory Networks (Weston et al., 2015a; Sukhbaatar et al., 2015) are a recent class of models that have proven successful for a variety of language understanding tasks such as question answering (Weston et al., 2015b), language modelling (Sukhbaatar et al., 2015) and conducting dialog (Dodge et al., 2016; Bordes et al., 2017). For dialogs, the entire conversation history is stored in the memory component of the model. It can be iteratively read from to perform reasoning and select the best possible responses based on the context. Implementing the modifications to the Memory Network architecture described by Bordes et al. (2017), we use the model as an end-to-end baseline and analyze its performance.

The user profile information is stored in the memory of the model as if it were the first turn of the conversation history spoken by the user, i.e. the model builds an embedding of the profile by combining the values of the embeddings of each attribute in the profile. Unlike Bordes et al. (2017), we do not make use of any match type features. Our goal is to analyse the capabilities of the existing Memory Network model to leverage profile information when conducting dialog. Appendix B shows illustrative examples of Memory Network predictions based on the experiments described in the following section.

5 Experiments
We report per-response accuracy (i.e. the percentage of responses in which the correct candidate is chosen out of all possible ones) across all the models and tasks in Table 4. The rows show tasks PT1-PT5 and columns 2-4 give the accuracy for each of the models. The hyperparameters for the models were optimized on the validation sets (values are provided in Appendix C).
Table 4: **Test results across all models and tasks.** For Memory Networks, the first number is the accuracy on the full set of dialogs for each task and the number in parenthesis is the accuracy on the small set (with 1000 dialogs). All other models were evaluated on the full set only.

| Task | Rule-based System | Supervised Embeddings | Memory Networks |
|------|-------------------|-----------------------|-----------------|
| PT1: Issuing API calls | 100 | 84.37 | 99.83 (98.87) |
| PT2: Updating API calls | 100 | 12.07 | 99.99 (99.93) |
| PT3: Displaying options | 100 | 9.21 | 58.94 (58.71) |
| PT4: Providing information | 100 | 4.76 | 57.17 (57.17) |
| PT5: Full dialog | 100 | 51.60 | 82.6 (77.74) |

As expected, handcrafted rule-based systems outperformed all machine learning models and solved all 5 tasks perfectly. However, it is important to note that building a rule-based system for real conversations is not easy- our tasks use a restricted vocabulary and fixed speech patterns.

Compared to results reported on the bAbI dialog tasks in Bordes et al. (2017), supervised embeddings performed significantly worse on the modified tasks. The model was unable to complete any of the tasks successfully and had extremely low per-response accuracy for PT2-5. In contrast, the same model reported 100 percent accuracy on bAbI dialog task T1 and above 55 percent on the other tasks. We attribute this drop in performance to the increased complexity of our tasks due to the four fold increase in vocabulary and the ten fold increase in candidate set size.

Memory Networks substantially outperformed supervised embeddings for all tasks. They completed PT1 and PT2 (issuing and updating API calls) with a very high degree of accuracy. This indicates that the model is able to implicitly track dialog state and personalize the bot’s utterance based on the user’s profile. However, visualizations of the parts of the memory that the model read from at each turn of the dialog indicate that the Memory Network architecture is not ideal for personalization of speech style. We provide examples of such visualizations in Appendix B. Results on PT3-PT5 suggest that Memory Networks were unable to use KB facts in conversation reliably. Analysis of the model’s memory show that it fails to interpret knowledge about entities and link it to the attributes of a user’s profile.

Restricting Memory Networks to only 1000 dialogs for training did not lead to any significant decreases in accuracy except for PT5. This indicates an issue with the learning mechanism as providing the model with more data did not result in any advantages in terms of solving the individual tasks.

### 5.1 Multi-task Learning

We also analyzed the Memory Network architecture in a multi-task learning scenario for conducting full dialog- we trained individual profile-specific models for each of the 6 profile permutations for speech style changes (described in Section 3.2), and compared their performance to a single multi-profile model. Each of the profile-specific models were trained on 1000 full dialogs between the bot and a user with the corresponding age and gender combination. The multi-profile model was trained on 1000 full dialogs from the PT5 small set, which contains dialogs with all 6 user profiles. For each profile, we report per-response accuracies for both the profile-specific and multi-profile model on 1000 test dialogs (with users having the same profile) in Table 5.
Table 5: **Test results for multi-task learning scenarios.** Each profile is associated with 1000 training dialogs (on which each profile-specific model is trained) and 1000 testing dialogs (on which both models are tested). The multi-profile model is trained on the PT5 small set.

| Profile          | Profile-specific model | Multi-profile model |
|------------------|------------------------|---------------------|
| male, young      | 80.38                  | 77.7                |
| female, young    | 80.15                  | 77.14               |
| male, middle-aged| 80.29                  | 77.59               |
| female, middle-aged | 80.21              | 77.8                |
| male, elderly    | 80.57                  | 77.82               |
| female, elderly  | 80.41                  | 77.52               |

The profile-specific models always outperformed the multi-profile model. However, it is worth noting that the multi-profile model had six times fewer training examples for any given profile and still performed only marginally worse (2%-3%). This indicates that training a single model on dialogs with multiple profiles which share logic and vocabulary is an effective learning strategy. An ensemble model may be superior in realistic learning conditions where obtaining sufficient data to train individual models is expense. Table 12 in Appendix B illustrates the differences between the two types of models.

6 Conclusions

This paper aims to bridge a gap in research on neural conversational agents by introducing a new open dataset of goal-oriented dialogs with user profiles associated with each dialog. The dataset acts as a testbed for the training and analysis of end-to-end goal-oriented conversational agents which must personalize their conversation with the user based on attributes in the user’s profile. As this work builds on top of the bAbI dialog dataset proposed by Bordes et al. (2017), crucial aspects of goal-oriented conversation have been split into various synthetically generated tasks to evaluate the strengths and weaknesses of models in a systematic way before applying them on real data. We demonstrated how to use our tasks to break down one such system, end-to-end Memory Networks. The model was unable to sufficiently perform reasoning or personalization to solve the tasks, indicating that further work needs to be done on learning methods for these aspects.

Despite the scenarios and language of the tasks being artificial, we believe that building mechanisms that can solve them is a reasonable starting point towards the development of sophisticated dialog systems in domains such as restaurant reservation, customer care or personal assistants. We hope that future research in this field will focus on developing better models to solve our tasks and on releasing datasets with real human - bot dialogs influenced by speaker profiles.
References

Bahdanau, D., Cho, K. & Bengio, Y. (2014), ‘Neural machine translation by jointly learning to align and translate’, *arXiv preprint arXiv:1409.0473*.

Bordes, A., Boureau Y. & Weston, J. (2017), ‘Learning end-to-end goal-oriented dialog’, *arXiv preprint arXiv:1605.07683*.

Dodge, J., Gane, A., Zhang, X., Bordes, A., Chopra, S., Miller, A. H., Szlam, A. & Weston, J. (2016), ‘Evaluating prerequisite qualities for learning end-to-end dialog systems’, *arXiv preprint arXiv:1511.06931*.

Eric, M. & Manning, C. D. (2017), ‘A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue’, *arXiv preprint arXiv:1701.04024*.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwinska, A., Colmenarejo, S. G., Grefenstette, E., Ramalho, T., Agapiou, J., Badia, A. P., Hermann, K. M., Zwols, Y., Ostrovski, G., Cain, A., King, H., Summerfield, C., Blunsom, P., Kavukcuoglu, K. & Hassabis, D. (2016), ‘Hybrid computing using a neural network with dynamic external memory’, *Nature* 538(7626), 471–476.

Halliday M. A. K. (1964), 'Comparison and translation', in M. A. K. Halliday, M. McIntosh and P. Strevens, *The linguistic sciences and language teaching*, London: Longman.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F. B., Wattenberg, M., Corrado, G., Hughes, M. Dean, J. (2016), ‘Google’s multilingual neural machine translation system: Enabling zero-shot translation’, *arXiv preprint arXiv:1611.04558*.

Li, J., Galley, M., Brockett, C., Gao, J. & Dolan, B. (2016a), ‘A diversity-promoting objective function for neural conversation models’, *arXiv preprint arXiv:1510.03055*.

Li, J., Galley, M., Brockett, C., Gao, J. & Dolan, B. (2016b), ‘A persona-based neural conversation model’, *arXiv preprint arXiv:1603.06155*.

Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013), ‘Efficient estimation of word representations in vector space’, *arXiv preprint arXiv:1301.3781*.

Perez, J. & Liu, F. (2017), ‘Gated end-to-end memory networks’, *arXiv preprint arXiv:1610.04211*.

Seo, M. J., Hajishirzi, H. & Farhadi, A. (2017), ‘Query-regression networks for machine comprehension’, *arXiv preprint arXiv:1606.04582*.

Serban, I. V., Lowe, R., Henderson, P., Charlin, L. & Pineau, J. (2017), ‘A survey of available corpora for building data-driven dialogue systems’, *arXiv preprint arXiv:1512.05742*.

Sukhbaatar, S., Szlam, A., Weston, J. & Fergus, R. (2015), ‘Weakly supervised memory networks’, *arXiv preprint arXiv:1503.08895*. 
A  Speech Style Changes

Table 6 displays the 15 original bot utterance patterns from the bAbI dialog tasks and the 6 profile-based modified utterance patterns associated with each of them.

B  Examples of Predictions of a Memory Network

Tables 7-12 illustrate examples of predictions by Memory Networks to support the various tasks and experiments described in the paper. All models were trained on the full modified dataset.

At any turn of the dialog, the Memory Network stores the conversation history in its memory and, based on the user’s input for that turn, pays attention to specific utterances from the memory. It can iteratively reason over the memory and uses a weighted combination of these utterances to predict the bot’s response to the user. In our visualization, we take the model state at a specific turn in the conversation and highlight the values of the attention weights over the memory for each iteration (called a hop).

C  Hyperparameters

Tables 13 and 14 display the hyperparameters used to train the best performing models for each task.
Table 6: Speech Style Changes based on profile. Column 1 shows the original bot utterance patterns from the bAbI dialog tasks. Columns 2-7 show the modified utterance patterns associated with each of the 6 profiles. Ellipses have been used in place of various KB entries such as restaurant names or attributes.

| Original bot utterance                  | Modified bot utterance (male, young) | Modified bot utterance (female, young) | Modified bot utterance (male, middle-aged) | Modified bot utterance (female, middle-aged) | Modified bot utterance (male, elderly) | Modified bot utterance (female, elderly) |
|-----------------------------------------|--------------------------------------|----------------------------------------|--------------------------------------------|---------------------------------------------|----------------------------------------|------------------------------------------|
| hello what can i help you with today    | hey dude what is up                  | hey girl how is it going               | hello sir what can i help you with         | hello maam how can i help you               | greetings sir what may i assist you with today | good day madam how could i assist you today |
| i'm on it                               | i'm on your request                  | be right back with your reservation    | i'm processing the request                 | give me a second for processing the reservation | excellent sir i will start the request now | thank you madam i shall start the reservation now |
| any preference on a type of cuisine     | what food are you looking for        | what food are you looking for          | what type of cuisine would you like to eat | what type of cuisine would you like to eat | may i know your preference on the type of cuisine | could you tell me your preference on the type of cuisine |
| where should it be                      | where should it be                   | where should it be located             | where should it be located                 | where should it be located                 | may i know where the restaurant should be located | could you tell me where the restaurant should be located |
| which price range are looking for       | what should the price be             | what should the price be               | which price range are you looking for      | which price range are you looking for      | may i know your preferred price range      | would you mind telling me your price range |
| how many people would be in your party  | how many are you                      | how many are you                       | how many people would be in your party     | how many people would be in your party     | may i know how many guests will be at your table | would you mind telling me how many guests shall be at your table |
| api_call ...                            | api_call ...                         | api_call ...                           | api_call ...                               | api_call ...                               | api_call ...                           | api_call ...                           |
| sure is there anything else to update   | cool anything else you want to update | awesome is there any other update      | great is there anything else to modify     | great is there any other thing to modify   | i will modify your request is there anything else to change | i shall modify your reservation is there any other change |
| ok let me look into some options for you | ok looking for options               | sure finding some options              | ok sir i'm looking for options for you     | sure maam i'm finding some options for you | excellent sir please give me a moment to provide you with options | thank you madam i shall provide you with options shortly |
| what do you think of this option: ...   | is this one cool: ...                | how about this one: ...               | is this a good option: ...                 | what do you think of this option: ...     | may i suggest this option: ...            | would you consider this option: ...     |
| sure let me find another option for you  | ok looking for something else        | sure finding something else            | ok i'll look for a better option           | sure i'll find a better option            | yes sir i will look for another suitable option | yes maam i shall find another suitable option |
| here it is ...                          | here you go ...                      | here you go ...                        | here it is ...                             | here it is ...                             | here is the information you asked for ... | here is the information you asked for ... |
| is there anything i can help you with   | want anything else                   | need something else                   | is there anything i can help you with      | can i assist you with something else      | may i help you in any other way sir       | could i assist you in some other manner madam |
| great me do the reservation             | cool its done                        | awesome you are done                  | great i'll finalize the request            | great let me do the reservation           | excellent i will finalize your request    | thank you i shall finish your reservation |
| you're welcome                          | no problem                           | happy to help                         | you're welcome                             | you're welcome                             | it was a pleasure to be of help to you sir | i am grateful to assist you madam         |
Table 7: **Modified Task 1 (Issuing API calls).** The model solves the task successfully by directing its attention towards the memories containing the values of the fields of the API call. The signal grows stronger on the second hop and switches almost exclusively to the price range in the third. This indicates that the model does not need multiple hops for successfully completing PT1. (Zero attention weight signifies a very small value.)

| Time | Locutor | Dialog History | Hop #1 | Hop #2 | Hop #3 |
|------|---------|----------------|-------|-------|-------|
| 1    | female elderly |                | 0.006 | 0     | 0     |
| 2    | User    | hi             | 0.012 | 0     | 0     |
| 3    | Bot     | good day madam how could i assist you today | 0     | 0     | 0     |
| 4    | User    | may i have a table | 0.006 | 0.248 | 0     |
| 5    | Bot     | thank you madam i shall start the reservation now | 0     | 0     | 0     |
| 6    | User    | <SILENCE>      | 0.064 | 0     | 0     |
| 7    | Bot     | could you tell me your preference on the type of cuisine | 0.001 | 0     | 0     |
| 8    | User    | i love french food | 0.307 | 0.103 | 0     |
| 9    | Bot     | could you tell me where the restaurant should be located | 0     | 0     | 0     |
| 10   | User    | madrid please | 0.265 | 0.282 | 0     |
| 11   | Bot     | would you mind telling me how many guests shall be at your table | 0     | 0     | 0     |
| 12   | User    | for four please | 0.225 | 0.076 | 0     |
| 13   | Bot     | would you mind telling me your price range | 0     | 0     | 0     |
| 14   | User    | in a cheap price range please | 0.110 | 0.290 | 0.999 |
| 15   | Bot     | thank you madam i shall provide you with options shortly | 0     | 0     | 0     |

**User input** <SILENCE>

**Correct answer** api_call french madrid four cheap

**Predicted answer** api_call french madrid four cheap
Table 8: **Speech Style and user profiles.** In order to study how the model uses profiles to modify speech style, we analysed a turn of the dialog unrelated to any tasks. The model learns to focus on the user’s profile along with the utterance containing the incomplete demand for the table. However, it also pays attention to the salutation, which intuitively should not have any impact on the bot’s output at the given turn. Multiple hops are unnecessary for modelling speech style.

| Time | Locutor   | Dialog History                                                                 | Hop #1 | Hop #2 | Hop #3 |
|------|-----------|--------------------------------------------------------------------------------|--------|--------|--------|
| 1    | male elderly|                                                                                  | 0.15   | 0      | 0      |
| 2    | User      | hello                                                                             | 0.306  | 0      | 0      |
| 3    | Bot       | greetings sir what may i assist you with today                                     | 0.008  | 0      | 0      |
| 4    | User      | may i have a table in a moderate price range with italian food for eight           | 0.536  | 0.999  | 0      |
| 5    | Bot       | excellent sir i will start the request now                                        | 0      | 0      | 0.999  |

**User input**  

**Correct answer**  
may i know where the restaurant should be located

**Predicted answer**  
may i know where the restaurant should be located

Table 9: **Modified task 2 (Updating API calls).** The model correctly learns to focus on the original API call and the changes demanded by the user. It does not require multiple hops to solve this task.

| Time | Locutor    | Dialog History                                                                 | Hop #1 | Hop #2 | Hop #3 |
|------|------------|--------------------------------------------------------------------------------|--------|--------|--------|
| 1    | female middle-aged |                                                                                  | 0.004  | 0      | 0      |
| 2    | User       | good morning                                                                     | 0.008  | 0      | 0      |
| 3    | Bot        | hello maam how can i help you                                                    | 0      | 0      | 0      |
| 4    | User       | may i have a table in bombay with spanish food for four in a expensive price range | 0.001  | 0      | 0.999  |
| 5    | Bot        | give me a second for processing the reservation                                  | 0.001  | 0      | 0      |
| 6    | User       | <SILENCE>                                                                        | 0.082  | 0      | 0      |
| 7    | Bot        | sure maam i'm finding some options for you                                       | 0      | 0      | 0      |
| 8    | User       | <SILENCE>                                                                        | 0.05   | 0      | 0      |
| 9    | Bot        | api_call spanish bombay four expensive                                           | 0.214  | 0.002  | 0      |
| 10   | User       | actually i would prefer with french food                                         | 0.635  | 0.997  | 0      |
| 11   | Bot        | great is there any other thing to modify                                         | 0      | 0      | 0      |
| 12   | User       | no                                                                                | 0.001  | 0      | 0      |
| 13   | Bot        | sure maam i'm finding some options for you                                       | 0      | 0      | 0      |

**User input**  

**Correct answer**  
api_call french bombay four expensive

**Predicted answer**  
api_call french bombay four expensive
Table 10: **Modified task 3 (Displaying options)**. The model should ideally be focusing on the user’s profile and the ratings, types and specialities of the various restaurants in the KB facts. It should also pay attention to the restaurants that have already been suggested to the user. However, it attends primarily to the location of a restaurant (which is not even the next one to be suggested). No meaningful attention to the user profile or KB facts used for implicit ranking indicates that the model is insufficient at reasoning over the KB. We have only shown important utterances in the table.

| Time | Locutor | Dialog History | Hop #1 | Hop #2 | Hop #3 |
|------|---------|----------------|--------|--------|--------|
| 1    | User    | female young non-veg pizza | 0.0001 | 0      | 0      |
| 20   | User    | resto_rome_moderate_italian_8stars_1 R_rating 8 | 0.0001 | 0      | 0      |
| 21   | User    | resto_rome_moderate_italian_8stars_1 R_type veg | 0      | 0      | 0      |
| 22   | User    | resto_rome_moderate_italian_8stars_1 R_speciality pizza | 0      | 0      | 0      |
| 29   | User    | resto_rome_moderate_italian_3stars_1 R_location rome | 0.879  | 0.999  | 0      |
| 33   | User    | resto_rome_moderate_italian_3stars_1 R_type veg | 0.0001 | 0      | 0      |
| 44   | User    | resto_rome_moderate_italian_2stars_2 R_rating 2 | 0      | 0      | 0      |
| 45   | User    | resto_rome_moderate_italian_2stars_2 R_type non-veg | 0.0001 | 0      | 0      |
| 46   | User    | resto_rome_moderate_italian_2stars_2 R_speciality pasta | 0      | 0      | 0      |
| 74   | User    | hi | 0      | 0      | 0      |
| 75   | Bot     | hey girl how is it going | 0      | 0      | 0      |
| 76   | User    | may i have a table in rome with italian food in a moderate price range | 0      | 0      | 0      |
| 77   | Bot     | be right back with your reservation | 0      | 0      | 0.999  |
| 83   | Bot     | how about this one: resto_rome_moderate_italian_8stars_2 | 0      | 0      | 0      |
| 84   | User    | no i don't like that | 0      | 0      | 0      |
| 85   | Bot     | sure finding something else | 0      | 0      | 0      |
| 86   | User    | <SILENCE> | 0.0003 | 0      | 0      |
| 87   | Bot     | how about this one: resto_rome_moderate_italian_3stars_2 | 0      | 0      | 0      |
| 88   | User    | no i don't like that | 0      | 0      | 0      |
| 89   | Bot     | sure finding something else | 0      | 0      | 0      |

**User Input**  &lt;SILENCE&gt;

**Correct Answer**  how about this one: resto_rome_moderate_italian_8stars_1

**Predicted Answer**  what food are you looking for
Table 11: Modified task 4 (Providing information). The model directs its attention to all the KB facts that it may need to provide but does not focus on the user profile sufficiently. Instead, it also attends to its own final utterance before the turn, which may have helped it judge the user’s age instead of using the profile. It correctly predicts that it has to display the social media information instead of the phone number for the young user, but provides the information for the wrong restaurant. Bordes et al. (2017) claim that ‘embeddings mix up the information and make it hard to distinguish between different KB entities, making answering correctly very hard.’ They overcome this problem by using match type features to emphasize entities that appear in the conversation history. We are yet to implement this technique and consider PT4 to not be sufficiently solved by Memory Networks.

| Time | Locutor | Dialog History | Hop #1 | Hop #2 | Hop #3 |
|------|---------|----------------|--------|--------|--------|
| 1    | male young |                | 0.003  | 0      | 0      |
| 2    | User | resto_madridCheap_indian_8stars_1 R_phone		resto_madridCheap_indian_8stars_1_phone | 0.379  | 0.311  | 0.001  |
| 3    | User | resto_madridCheap_indian_8stars_1 R_cuisine indian | 0.001  | 0      | 0      |
| 4    | User | resto_madridCheap_indian_8stars_1 R_address		resto_madridCheap_indian_8stars_1_address | 0.021  | 0.005  | 0      |
| 5    | User | resto_madridCheap_indian_8stars_1 R_location madrid | 0.003  | 0.015  | 0.342  |
| 6    | User | resto_madridCheap_indian_8stars_1 R_number eight | 0.002  | 0      | 0      |
| 7    | User | resto_madridCheap_indian_8stars_1 R_price cheap | 0.002  | 0.001  | 0      |
| 8    | User | resto_madridCheap_indian_8stars_1 R_rating 8 | 0.001  | 0      | 0      |
| 9    | User | resto_madridCheap_indian_8stars_1 R_type veg | 0.001  | 0      | 0      |
| 10   | User | resto_madridCheap_indian_8stars_1 R_speciality biryani | 0.002  | 0      | 0      |
| 11   | User | resto_madridCheap_indian_8stars_1 R_social_media	resto_madridCheap_indian_8stars_1_social_media | 0.084  | 0.069  | 0.001  |
| 12   | User | resto_madridCheap_indian_8stars_1 R_parking		resto_madridCheap_indian_8stars_1_parking | 0.354  | 0.222  | 0      |
| 13   | User | resto_madridCheap_indian_8stars_1 R_public_transport	resto_madridCheap_indian_8stars_1_public_transport | 0.019  | 0.001  | 0      |
| 14   | User | hello | 0.001  | 0      | 0      |
| 15   | Bot | hey dude what is up | 0      | 0      | 0.004  |
| 16   | User | can you make a restaurant reservation at	resto_madridCheap_indian_8stars_1 | 0.005  | 0      | 0      |
| 17   | Bot | cool its done | 0.116  | 0.37   | 0.652  |

User input:  may i have the contact details of the restaurant
Correct answer:  here you go resto_madridCheap_indian_8stars_1_social_media
Predicted answer:  here you go resto_romeCheap_indian_7stars_2_social_media
Table 12: **Predictions of multi-profile model versus profile-specific model.** For the chosen profile (female, middle-aged), the multi-profile model attends to the user’s profile, greeting and incomplete inquiry to modify its speech style and ask for the missing field. The profile-specific model does not need to perform such personalization, and thus has a narrow focus.

| Time | Locutor | Dialog History | Multi-profile model | Profile-specific model |
|------|---------|----------------|---------------------|------------------------|
| 1    | female middle-aged non-veg pizza | 0.044 | 0 |
| 2    | User | good morning | 0.397 | 0 |
| 3    | Bot | hello maam how can i help you | 0.005 | 0.012 |
| 4    | User | can you make a restaurant reservation for two people in bombay with italian cuisine | 0.533 | 0.987 |
| 5    | Bot | give me a second for processing the reservation | 0 | 0 |

*User input* | <SILENCE>

*Correct answer* | which price range are you looking for

*Predicted answer* | which price range are you looking for (for both models)

Table 13: **Hyperparameters for Supervised Embeddings.** If Use History is True, conversation history is added to the last user utterance to create the input. If False, only the last utterance is used as input.

| Task | Learning Rate | Margin | Embedding Dimension | Negative Candidates | Use History |
|------|---------------|--------|---------------------|---------------------|-------------|
| PT1  | 0.01          | 0.01   | 32                  | 100                 | True        |
| PT2  | 0.01          | 0.01   | 128                 | 100                 | False       |
| PT3  | 0.01          | 0.1    | 128                 | 1000                | False       |
| PT4  | 0.001         | 0.1    | 128                 | 1000                | False       |
| PT5  | 0.01          | 0.01   | 32                  | 100                 | True        |

Table 14: **Hyperparameters for Memory Networks.**

| Task | Learning Rate | Margin | Embedding Dimension | Negative Candidates | Number of Hops |
|------|---------------|--------|---------------------|---------------------|----------------|
| PT1  | 0.001         | 0.01   | 20                  | 100                 | 1              |
| PT2  | 0.001         | 0.01   | 20                  | 100                 | 1              |
| PT3  | 0.001         | 0.01   | 20                  | 100                 | 3              |
| PT4  | 0.001         | 0.01   | 20                  | 100                 | 3              |
| PT5  | 0.001         | 0.01   | 20                  | 100                 | 3              |