Proactive Suggestion Generation: Data and Methods for Stepwise Task Assistance

Elnaz Nouri, Robert Sim, Adam Fourney, and Ryen W. White
Microsoft Research, Redmond, WA 98052
{elnouri, rsim, adamfo, ryenw}@microsoft.com

ABSTRACT

Conversational systems such as digital assistants can help users perform many simple tasks upon request. Looking to the future, these systems will also need to fully support more complex, multi-step tasks (e.g., following cooking instructions), and help users complete those tasks, e.g., via useful and relevant suggestions made during the process. This paper takes the first step towards automatic generation of task-related suggestions. We introduce *proactive suggestion generation* as a novel task of natural language generation, in which a decision is made to inject a suggestion into an ongoing user dialog and one is then automatically generated. We propose two types of stepwise suggestions: multiple-choice response generation and text generation. We provide several models for each type of suggestion, including binary and multi-class classification, and text generation.

ACM Reference Format:
Elnaz Nouri, Robert Sim, Adam Fourney, and Ryen W. White . 2020. Proactive Suggestion Generation: Data and Methods for Stepwise Task Assistance. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3397271.3401272

1 INTRODUCTION

Dialog generation [9] and question answering are central features of state-of-the-art conversational assistants [11]. Assistants such as Amazon Alexa, Apple Siri, and Microsoft Cortana can answer factoid questions, perform simple tasks such as music playback and home automation, and engage in rudimentary chit-chat. Chatbots such as Microsoft's Xiaoice and Zo have also demonstrated a high-degree of conversational capability, but lack the ability to track a task to completion. Conversely, task-oriented agents such as booking and customer service bots have limited expressiveness and tend to focus on narrowly defined slot-filling tasks.

In this paper, we develop methods that enable suggestions tied to the conversational context in a task-oriented conversational assistant, so-called *proactive suggestion generation*. In particular, we examine the recipe domain and aim to produce relevant stepwise suggestions that can add color and depth to an interactive experience while preparing a recipe. For instance, after reading out a recipe step involving separating egg yolk, a contextually-aware assistant may offer a helpful tip on how to reliably accomplish this task without making a mess. Figure 1) shows a user interacting with a conversational assistant that can provide useful suggestions to the user (shown on the right of the figure) for accomplishing the task. Note that in these scenarios the system response to a user prompt is prescribed by the task (e.g., the delivery of the next step in the recipe), but the suggestion is not–it is meant to provide an automatically generated enhancement to the system response. Such suggestions have the potential to drive user engagement and increase user satisfaction with conversational systems. Outside the cooking domain, proactive suggestions can be useful in other task-directed conversational settings, including both multi-step tasks, or even single-turn conversations, such as querying the weather.

Figure 1: User preparing a recipe with an assistant offering suggestions for each step. Figure header shows current step. Thought bubbles illustrate potential user queries–proactive suggestions should preempt the need for them.

This paper makes three main contributions: first, we propose a new conversational assistance task: *proactive suggestion generation*; second, we develop a dataset designed for facilitating this task in the context of recipe preparation assistance; and finally, we develop models that demonstrate the feasibility of performing proactive suggestion generation in a conversational system.

2 RELATED WORK

Pei and Li [15] explored the generation of relevant and informative responses in dialogs using a sequence-to-sequence (seq2seq) model.Liu et al. [10] demonstrated a knowledge diffusion network

SIGIR '20, July 25–30, 2020, Virtual Event, China

ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401272

that performs fact matching and entity diffusion in order to ground both factoid and chit-chat based dialogs in real-world knowledge. Lei et al. [8] simplified task-oriented dialogs using a seq2seq model that tracks "belief spans" representing dialog segments that capture the state of the task. Gu et al. [3] demonstrates how multi-modal response generation can be performed. More generally, theoretical frameworks for conceptualizing conversation have been proposed by several authors [1, 18, 23].

Our work augments conversational assistants using generative models that add dialog enhancements, given the current dialog state. These enhancements are not limited to voice or text output: the system may deliver an image or video that is relevant to the current point in time. We focus on the recipe domain since tasks have clear multi-step structure and many sources of data are already available and used by researchers, e.g., the multi-modal dataset in [13].

There is also research on recipe generation. Bosselut et al. [2] generate recipes using discourse-aware rewards with reinforcement learning. RecipeQA [26] is a dataset comprised of 36K automatically generated question-answer pairs and the authors propose different multi-modal comprehension tasks on this dataset: textual cloze and visual cloze, visual coherence, and visual ordering on a multi-modal dataset including recipe steps along with associated images.

Our proposed task for suggestion generation differs from previous tasks defined for dialog generation and for question answering. In a dialog generation task [9], the goal is to generate an utterance in response to an utterance produced by the user. The conversation itself may be goal-oriented (e.g., completion of a task) or casual chit chat. In question answering tasks, the goal is to generate the answer containing the information that is requested. An answer is expected in response to the question. However, this is not the case with proactive suggestion generation. In suggestion generation, the assistant first makes a decision on whether making a suggestion is appropriate for the dialog state, and then it proceeds to generate a suggestion based on the content and context of the interaction. The generated suggestion may contain information useful to the user without explicitly being requested.

3 RECIPE STEPWISE SUGGESTION DATASET

3.1 Data Collection

In total, we collected 16,659 recipes licensed under a Creative Commons attribution license from public web sites. The recipes contain the following information: recipe name and identification number, recipe yield, ingredients, and instructions. For human annotation, a subset of 2,000 recipes were selected based on the quality of the recipe content, language (English), and ensuring that each had 4-7 steps (to provide sufficient data for suggestion generation but still be manageable for human judges).

For judging, we used a crowdsourcing platform that requires the judges to meet the task requirements and accept a user agreement. The task instructions informed the crowdworkers that the goal was to prepare a recipe step by step while interacting with a digital assistant capable of using multiple devices (as in Figure 1) that provides recommendations for preparation of the recipe. Each recipe was shown in its entirety, including the title, its ingredients, and steps. The judgment task dynamically populates a set of questions for each step as the judge walks through them.

At each step, judges provide the following information: a binary label indicating the utility of recommending additional information as a suggestion, one or more labels indicating the best modal format(s) for presentation of the suggestion to the user, one or more labels indicating action(s) that are suitable for delivering the suggestion to the user, and one or more labels indicating the most suitable device(s) for delivering the suggestion to the user.

For each step, we elicited three classes of information by asking the following questions:

i **SUGGESTION**: What additional useful information should be presented as a suggestion regarding the content of the step. This is helpful information in accomplishing the step.
ii **QUESTION**: What question would be asked if an expert chef is available and the goal was to get help from them.
iii **SEARCH QUERY**: What search query shall be used on the web for obtaining this information via a search engine.

Questions and search queries were collected to facilitate retrieval of information relevant to the step. We chose to collect both types as dialog systems need to be able to handle the two, and it enables us to infer relationships between verbose questions and concise web queries in this domain. Robust handling of verbose queries is an increasingly important aspect of information retrieval [4].

Table 1 summarizes the schema for each dataset row. Table 2 contains a sample of the free-form responses from human judges.

3.2 Dataset Quality Evaluation

In initial evaluation, all data were reviewed by at least one expert annotator and invalid or low-quality entries were removed. This manual quality check resulted in rejection of 13.3% of the collected data. In the second round of evaluation, we used majority vote by 3 crowdworkers to evaluate the quality of each suggestion entry, question entry, and each search query by rating it on a 3-level Likert scale (good=no issues, neutral=some issues, bad=many issues) for the following 3 criteria: syntax and grammar, semantics and meaning, overall suitability. Queries for grammar are rated as (74.2% good, 15.4% neutral), for semantics (80.1% good, 10.3% neutral) and overall (61.9% good, 10.3% neutral). Questions for grammar are rated as (68% good, 21.7% neutral), for semantics (78.5% good, 8.7% neutral), and overall (64.3% good, 10.1% neutral).

4 STEPWISE SUGGESTION CLASSIFICATION AND GENERATION TASKS

Our proactive suggestion generation tasks are defined as follows: Given a step from a multi-step task definition, the model is required to determine whether a suggestion is to be provided (Classification Task I) and then decide on the format (Classification Task II), the action (Classification Task III), the device (Classification Task IV), as well as generating the content for the suggestion (Generation Task I). Our stepwise suggestion models consist of classifiers (both binary and multi-class) and generative models for suggestion content.

4.1 Stepwise Classification Tasks

An ideal digital assistant should distinguish between points during the task where suggesting additional information is helpful or not. We frame this as binary classification. Since undesired recommendations can be distracting and dissatisfying [5, 19], Classification Task I is a critical consideration for any recommender system.

Table 1: Description of a row in in our dataset.

Recipe Metadata	Recipe Step	Recommendation Labels	Suggestion	Search Query	Question
"Name", "Image", "Id", "Url"	Instruction	Labels for suggestion, device, format	Useful Information as Suggestion	Search query for soliciting information	Conversational question to solicit information

Table 2: Sampled human-generated data from suggestion, search query, and question dataset. The first two columns represent the recipe title and step that are shown to judges. The last three columns are judge responses.

Recipe Title	Recipe Step	Suggestion	Search Query	Question
Indian Butter Chicken	Place the yogurt, ground almonds, all the dry spices, ginger, garlic, tomatoes and salt in a mixing bowl and blend together thoroughly.	Instructions on grinding your own almonds.	How to grind almonds	How to grind your own almonds for indian cuisine.
Schezwan Poha By Harpal	Heat oil in a pan. Add mustard seeds, curry leaves, green chilli, and onion. Saute for 2 mins	You can use ghee instead of oil.	Best type of oil for making Schezwan Poha?	What type of oil should I use?
Roasted Artichoke Stuffed with Garlic and Sage	2. Next, press the garlic clove into the center the artichoke, season with salt and pepper. Place sage on the top and wrap in tinfoil.	do I keep the garlic clove whole?	How much minced garlic is equal to a clove of garlic?	Can I used minced garlic?
Rose Water Iced Tea	In a pitcher or serving jug, pour enough water to steep a tea bag. Set aside.	coffee bag can be used instead of tea bag	tea bag substitute Rose Water Iced Tea	what i can substitute tea bag

Table 3: Accuracy of models for stepwise decisions on different classification tasks (Section 4.1). Models which significantly outperform the LR baseline (according to paired sample t-test, $p < 0.05$) are denoted with (*). Best performing models for each condition are shown in bold. Values are percentages.

Classification Task	I. whether a suggestion is helpful	II. format suggestion	III. action suggestion	IV. device suggestion
Logistic Regression	55.10	73.37	79.36	73.80
RNN	87.50*	91.90*	87.60*	**84.80***
RNN+GloVe 300	**88.67***	**94.40***	**88.20***	82.80*

Table 4: Sampled data from the result of the generation by our generation model (Section 5.2).

Text Input choices	Network Input	Predicted	Target
(step) fill cocktail shaker good amount ice (query) full ingredients steps shown pictures please (question) tell full ingredients steps shown pictures please	step	may may also something cream cheese prepared recipe	tell full ingredients steps shown pictures please
	step + query	tell full ingredients steps shown pictures please	
	step + question	tell full ingredients steps shown pictures please tell	
	step + query + question	tell full ingredients steps shown pictures please tell	
(step) preheat oven 180c (query) coconut oil substitute plantain bread (question) substitute coconut oil	step	many f	olive oil used instead coconut oil
	step + query	coconut oil used instead olive oil	
	step + question	coconut oil used instead olive oil	
	step + query + question	olive oil used instead coconut oil	
(step) stir green onions (query) sub meats make vegan meals (question) turkey sausage used	step	order ingredients per specifications forming patties	make turkey let chill fridge 24 hours
	step + query	make sure move around added	
	step + question	use sausage cocktail instead vegetable	
	step + query + question	use green onions instead	

Table 5: Suggestion generation performance (Section 4.2). Best performing condition is shown in bold.

Generation Condition	BLEU
step	0.80
step + query	8.97
step + question	8.01
step + query + question	**9.79**

If the suggestion provision is affirmative, subsequent decisions about format (Classification Task II), action (Classification Task III), and device (Classification Task IV) can be made. For Classification Task II (format), the classes are "Audio," "Text," "Image," "Video," and "Other" (judge specified). For Classification Task III (action), the labels are "Provide clarification," "Search the web," "Show a video," "Show a picture," "Activate new devices," "Provide substitution," "Show useful related advertisement," and "Other" (judge specified). Finally, Classification Task IV (device): when multiple devices with varying capabilities can be synchronized together (e.g., [25]), the system needs to determine which devices are most suitable. The classes are "Smart speaker - No screen," "Smart speaker - Integrated screen," "Smart watch," "Laptop or tablet," "Smart kitchen appliances," "Device with camera," and "Other" (judge specified).

4.2 Stepwise Suggestion Generation Task

This is a text generation task to provide suggestion content at each step. Once the system determines a suggestion is needed and selects actions, devices, and format, it also must generate its content.

5 MODELS

5.1 Models for Classification

Logistic Regression We use logistic regression (LR) as our primary baseline model. Logistic regression is an interpretable model that has successfully been used on text classification problems. This model is compact and cost-efficient for training. We used unigrams combined with TF-IDF to represent text features.

Recurrent Neural Networks We use a neural model based on Recurrent Neural Networks (RNNs) [20] for both single-class and multi-class classification. Long-short-term-memory LSTM [6] is used for the RNN network, with 4 bidirectional layers. A single fully connected layer is employed on top of the network using either the last hidden state or the average of all hidden states as input. A trainable word embedding vector is employed as input layer, with both random initialization and pre-trained GloVe embeddings [16]. The embedding dimension size is 400 for random initialization, and GloVe 300 is used in pre-trained version. The best results were achieved when fine-tuning the GloVe layer. The LSTM embedding size is 128. The model is trained for 50 epochs, using the Adam optimizer, an initial learning rate of 0.005, and a batch size of 16.

5.2 Neural Models for Generation

The neural model for generation is based on a sequence-to-sequence (SEQ-2-SEQ) model [22, 24] with attention, for generation of conversational responses based on context [21]. The encoder is a 2-layer bi-directional GRU layer, and decoder is a 2-layer uni-directional one. Embedding size of 500 is employed. The decoder is using global attention [12] over encoder hidden states. A dropout rate of 0.1 in embedding and GRU layers, and a batch size of 64 is used.

6 EXPERIMENTS

To evaluate the classification models, we use 5-fold cross-validation. We use classification accuracy, common in recommendation research [5], which is simply the percentage of correct predictions.

Suggestion generation is evaluated under four input conditions, generation based on: (1) content of the current step in the task, (2) current step and user query, (3) current step and user question, and (4) current step, query, and question.

In studying these four conditions, we assume that using information on how humans would solicit suggestions through queries and questions would result in the generation of better suggestions.

Evaluation of the models for stepwise classification tasks (Section 4.1) is presented in Table 3. We observe that the model for predicting whether suggestion should be provided shows the best performance when RNN is used with GloVe embedding representation, although the improvement over using RNN alone is not statistically significant. That model also performs best at determining the suitable format and determining the best action. RNN without embeddings performs best for device suggestion.

Table 4 shows a sample of the generated suggestions by our model. Target column is the suggestion made by human judges. We use the BLEU score [14] to evaluate the performance of our generation model against the target data. Using the step content alone results in the lowest BLEU score. Using all inputs (step, query, and question in conjunction) results in highest BLEU score.

7 CONCLUSIONS AND FUTURE WORK

We have presented a novel data set for generating proactive suggestions in task-oriented dialogs. We developed models for several tasks of interest, including whether a suggestion should be generated for a step in a task, the form in which it should be delivered, and on what user device. We also made good progress on auto-generating suggestion content. One challenge here is to develop a model that can accomplish both classification and generation tasks jointly. In the future, we aim to improve generation performance by collecting more data (to help account for subjective variations in annotations), and experimenting with other semantic embeddings such as BERT [7] and GPT2 [17], which have recently been shown to perform well in language generation in downstream tasks [17].

REFERENCES

[1] Leif Azzopardi, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. 2018. Conceptualizing agent-human interactions during the conversational search process. In *International Workshop on Conversational Approaches to Information Retrieval*.

[2] Antoine Bosselut, Asli Celikyilmaz, et al. 2018. Discourse-Aware Neural Rewards for Coherent Text Generation. In *NAACL-HLT*. 173–184.

[3] Xiaodong Gu, Kyunghyun Cho, Jung Woo Ha, and Sunghun Kim. 2019. Dialogwae: Multimodal response generation with conditional Wasserstein auto-encoder. In *ICLR*.

[4] Manish Gupta, Michael Bendersky, et al. 2015. Information retrieval with verbose queries. *Foundations and Trends in Information Retrieval* 9, 3-4 (2015), 209–354.

[5] Jonathan L Herlocker, Joseph A Konstan, et al. 2004. Evaluating collaborative filtering recommender systems. *TOIS* 22, 1 (2004), 5–53.

[6] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[7] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. 4171–4186.

[8] Wenqiang Lei, Xisen Jin, et al. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *ACL*. 1437–1447.

[9] Jiwei Li, Will Monroe, et al. 2017. Adversarial learning for neural dialogue generation. In *EMNLP*. 2157–2169.

[10] Shuman Liu, Hongshen Chen, et al. 2018. Knowledge diffusion for neural dialogue generation. In *ACL*. 1489–1498.

[11] Sidi Lu, Yaoming Zhu, et al. 2018. Neural text generation: past, present and beyond. *arXiv preprint arXiv:1803.07133* (2018).

[12] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*. 1412–1421.

[13] J. Marin, A. Biswas, et al. 2019. Recipe1M+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019), 1–1.

[14] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. [n. d.]. Bleu: A method for automatic evaluation of machine translation. In *ACL*. 311–318.

[15] Jiaxin Pei and Chenliang Li. 2018. S2SPMN: A simple and effective framework for response generation with relevant information. In *EMNLP*. 745–750.

[16] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. 1532–1543.

[17] Alec Radford, Jeffrey Wu, et al. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1, 8 (2019), 9.

[18] Filip Radlinski and Nick Craswell. 2017. A theoretical framework for conversational search. In *CHIIR*. 117–126.

[19] Francesco Ricci, Lior Rokach, et al. 2011. Introduction to recommender systems handbook. In *Recommender Systems Handbook*. Springer, 1–35.

[20] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45, 11 (1997), 2673–2681.

[21] Iulian V Serban, Alessandro Sordoni, et al. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*.

[22] Ilya Sutskever, Oriol Vinyals, et al. 2014. Sequence to sequence learning with neural networks. In *NeurIPS*. 3104–3112.

[23] Johanne R Trippas, Damiano Spina, et al. 2018. Informing the design of spoken conversational search. In *CHIIR*. 32–41.

[24] Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869* (2015).

[25] Ryen W White, Adam Fourney, et al. 2019. Multi-device digital assistance. *CACM* 62, 10 (2019), 28–31.

[26] Semih Yagcioglu, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. 2018. RecipeQA: A challenge dataset for multimodal comprehension of cooking recipes. In *EMNLP*. 1358–1368.