

NUANCED: Natural Utterance Annotation for Nuanced Conversation with Estimated Distributions

Zhiyu Chen^{1*}, Honglei Liu², Hu Xu², Seungwhan Moon², Hao Zhou² and Bing Liu²

¹University of California, Santa Barbara

²Facebook

zhiyuchen@cs.ucsb.edu,

{honglei, huxu, shanemoon, haozhoustat, bingli}@fb.com

Abstract

Existing conversational systems are mostly agent-centric, which assumes the user utterances will closely follow the system ontology. However, in real-world scenarios, it is highly desirable that users can speak freely and naturally. In this work, we attempt to build a user-centric dialogue system for conversational recommendation. As there is no clean mapping for a user's free form utterance to an ontology, we first model the user preferences as estimated distributions over the system ontology and map the user's utterances to such distributions. Learning such a mapping poses new challenges on reasoning over various types of knowledge, ranging from factoid knowledge, commonsense knowledge to the users' own situations. To this end, we build a new dataset named NUANCED that focuses on such realistic settings, with 5.1k dialogues, 26k turns of high-quality user responses. We conduct experiments, showing both the usefulness and challenges of our problem setting. We believe NUANCED can serve as a valuable resource to push existing research from the agent-centric system to the user-centric system. The dataset is publicly available¹.

1 Introduction

Conversational artificial intelligence is one of the long-standing research problems in natural language processing, such as task-oriented dialogue (Wen et al., 2017; Budzianowski et al., 2018; Hosseini-Asl et al., 2020), conversational recommendation (Sun and Zhang, 2018; Zhang et al., 2018) and chi-chat (Adiwardana et al., 2020; Roller et al., 2020) etc. However, most existing systems are *agent-centric*. Such systems require the users to *unnaturally* adapt to and even have a learning curve on the system ontology, which is largely unknown

*Work done as a research intern at Facebook.

¹<https://github.com/facebookresearch/nuanced>

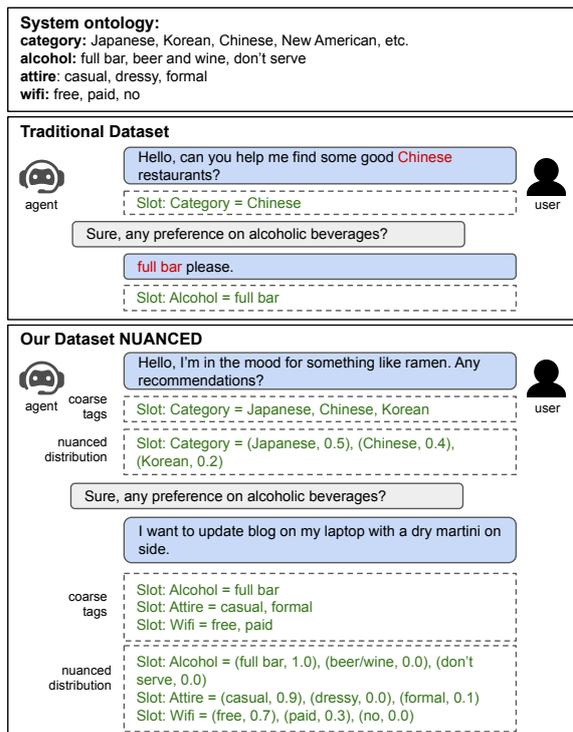


Figure 1: Examples of traditional dataset and NUANCED: In NUANCED, we model the user preferences as distributions over the ontology to allow mapping of entities unknown to multiple values and slots for efficient conversation.

to the users (such as the sample instructions for most smart speakers). Figure 1 shows a dialogue snippet commonly found in traditional datasets: the user is *expected* to closely follow the system ontology with the exact ontology terms, or at most with minor variations like synonyms.

In the real-world use cases, such formulation may easily result in information loss, or breaks a conversation if the user speaks anything out of the system ontology; In this work, we argue that a smart agent can ideally be more *user-centric*, by allowing users to speak freely without restrictions. The system is expected to uncover the connection between the freestyle user utterance and one or more slots and values by the system ontology.

To build a *user-centric* dialogue system, we propose to model the mapping from the free form user utterances to the system ontology as probability distributions to capture fine-grained user preferences. To learn the distributions, we construct a new dataset, named NUANCED (Natural Utterance Annotation for Nuanced Conversation with Estimated Distributions). NUANCED targets conversational recommendation because such type of dialogue system encourages more modeling of soft matching and implicit reasoning for user preference. We employ professional linguists to annotate the dataset, and end up with 5.1k dialogues and 26k turns of high-quality user utterances. Our dataset captures a wide range of phenomena naturally occurring in realistic user utterances, including specified factoid knowledge, commonsense knowledge and users’ own situations. We conduct comprehensive experiments and analyses to demonstrate the challenges. We hope NUANCED can serve as a valuable resource to bridge the gap between current researches and real-world applications.

2 Related Work

Task-oriented dialogue systems are typically divided into several sub modules, including user intent detection (Liu and Lane, 2016; Gangadharaiah and Narayanaswamy, 2019), dialogue state tracking (Rastogi et al., 2017; Heck et al., 2020), dialogue policy learning (Peng et al., 2017; Su et al., 2016), and response generation (Dusek et al., 2018; Wen et al., 2015). More recent approaches begin to build unified models that bring the pipeline together (Chen et al., 2019; Hosseini-Asl et al., 2020). *Conversational recommendation* focus on combining the recommendation system with online conversation to capture user preference (Fu et al., 2020; Sun and Zhang, 2018; Zhang et al., 2018). Previous works mostly focus on learning the agent side policy to ask the right questions and make accurate recommendations, such as (Xu et al., 2020; Lei et al., 2020; Li et al., 2020; Penha and Hauff, 2020). Chit-Chat (Adiwardana et al., 2020; Roller et al., 2020) is the most free form dialogue but almost with no knowledge grounding or state tracking. Both existing task-oriented, conversational recommendation systems have a pre-defined system ontology as a representation connected to the back-end database. The ontology defines all entity attributes as slots and the option values for each slot. In existing datasets, such as the DSTC challenges (Williams

et al., 2014), Multi-WOZ (Budzianowski et al., 2018), MGConvRex (Xu et al., 2020), etc, the utterances from the users mostly closely follow the system ontology. While in task-oriented dialogue systems, parsing the user utterances into dialogue states is more on hard matching, in conversational recommendation systems soft matching is more encouraged since the user preferences are more salient and diverse in this type of conversations.

3 The NUANCED Dataset

3.1 User Preference Modeling

Given a system ontology, denote the set of all slots as $\{S_i\}$, with the option values for each slot as $\{V_i^j\}$. Denote the current user utterance as T and dialogue context (of past turns) as C . We model the user preference as a distribution over each slot-value, namely *preference distribution*:

$$P_i^j = P(V_i^j|T, C). \quad (1)$$

Note that we expect the representation to be general, expandable, and to hold the fewest assumptions, i.e., there is no assumption on the dependency among slot-values, nor the completeness of the value set. Therefore we model the distribution as a Bernoulli distribution over each slot-value. Intuitively, P_i^j represents the probability that the user chooses an item with attributes V_i^j , under the observed condition of the dialogue up to the current turn. Note that the preference distributions may differ among individuals which causes variances, In this work, we aim to aggregate estimated distributions from large-scale data collected from multiple workers as “commonsense” distributions. We leave modeling user-specific distributions to future work.

3.2 Dataset Construction

We first simulate the dialogue flow with the preference distributions, then we ask the annotators to compose utterances that imply the distribution.

3.2.1 Dialogue Simulator

We follow the approach from the MGConvRex dataset (Xu et al., 2020) to build the user visiting histories from real-world data. For each user with its visiting history as a list of restaurants with slot-values, we sample a subset of the history and aggregate to get a value distribution for each slot. For example, in the list of restaurants of a user’s visiting history, we sampled two restaurants, restaurant 1 and restaurant 2. Restaurant 1 has the slot-values of Alcohol = full_bar, Restaurant 2 has the

slot-value of Alcohol = beer_and_wine. Then the aggregated distributions is Alcohol = (full_bar, 0.5), (beer_and_wine, 0.5), (no_serve, 0.0). As generally, for the same user, the attributes of its visited restaurants tends to follow certain trends. Therefore the aggregated distributions created this way can be more natural. Using the sampled distribution as the ground truth distribution, we simulate the dialogue skeletons of the following scenarios: 1) Straight dialogue flow: the system asks each slot, followed by the user response filled with preference distributions; 2) User updating preference: the user updates the preference distributions in a previous turn; 3) System yes/no questions: the system can choose to ask confirmation questions; For each turn, we randomly select 1 to 3 slots, corresponding to the cases that the user utterances naturally imply multiple slot-values. The system turns are composed using templates.

3.2.2 User Utterances Composition

After simulating the dialogue skeletons, we employ professional linguists to do the composition to ensure high quality. We provide two composing strategies: **Implicit Reasoning**: do not mention the slot-value terms explicitly. This is the focus of this work because we expect that users are unaware of the system ontology and to depict their requests naturally. **Explicitly Mention**: use the slot-value terms (or synonyms), as a backup option when the first one is not applicable. We also emphasize the following aspects: 1) Read the whole dialogue first to have an overall “story” in mind before composing each utterance to ensure consistency; 2) Try to compose utterances as diverse as possible; 3) Reject any cases with invalid or unnatural preference distributions. We provide learning sessions to linguists to ensure they all master the tasks.

3.3 Dataset Statistics and Analysis

With an average of 5.39 user turns per dialogue, we have 5,100 dialogues consisting of 25,757 user turns. The user utterances have an average length of 19.43 tokens. 84.7% of the utterances are composed using *implicit reasoning*; 6.5% of the utterances explicitly mention the ontology terms, and the rest use mixed strategies. The train / valid / test split is 3,600 / 500 / 1,000 in the number of dialogues, and 18,182 / 2,529 / 5,046 in the number of user turns. To evaluate the quality of our dataset, we randomly sample 500 examples and ask the linguistics whether a preference distribution is rea-

sonable based on the corresponding utterance. We end up with a turn-level correctness rate of 90.2%.

Among the utterances involving implicit reasoning, we summarize 3 basic reasoning types. The examples are shown in Table 1. **Type I (Factoid Knowledge)** is largely agreed on by people and is relatively stable. **Type II (Commonsense Knowledge or User Situations)** may not be formally defined. For example, a food item less than \$10 is considered cheap. In many cases, such knowledge needs to be inferred from a situation described by users. **Type III (Mix of Type I and II)** may appear in a single utterance.

3.3.1 NUANCED-reduced

We also provide a *reduced* variant called **NUANCED-reduced**, by discretizing the distributions for preference into binary numbers. For all slot-values with a positive preference distribution² we label them as 1.0, otherwise 0.0. This reduced variant does not have continuous probabilities to tell the nuanced differences but it still needs to map free form utterances to binary labels. We conduct human evaluation by asking the annotators to decide which representation can better capture more fine-grained user preferences. As Table 2 shows, NUANCED can better capture the nuanced information. Note that in real applications, which version of the data to use may depend on requirements of the system, i.e., level of granularity for state representation.

4 Experiments

In this section, we conduct experiments on both versions of the datasets in §4.1 and §4.2, respectively.

4.1 NUANCED-reduced

4.1.1 Baselines

Exact match & Random guess We follow the preceding system query to make slot prediction; we then use an exact match to predict the slot-values; if no match is found, we apply a random guess.

BERT (Devlin et al., 2019), The input is the concatenation of the slot name, current turn system question and user utterance, and the dialogue context of past turns. We add two types of prediction heads on the [CLS] token of BERT, one for slot prediction (whether the input slot is updated or not), and the other for the value prediction of each slot.

²In practice we set a threshold of 10%, because in the utterance composition stage a preference distribution lower than 10% is generally considered ignorable.

Reasoning types	Example user utterances	Example preference distributions
Type I Factoid Knowledge (37.3%)	I really want a G&T or a Riesling , but I could also have a tonic water .	Slot: Alcohol = (full_bar, 0.7), (beer_and_wine, 0.2), (don't_serve, 0.1)
Type II Commonsense knowledge or User Situations (43.8%)	five to ten dollars , I don't want a place with people wearing ties , you know?	Slot: Price = (cheap, 0.6), (affordable, 0.4), (moderately_priced, 0.0), (expensive, 0.0) Slot: Attire = (casual, 1.0), (dressy, 0.0), (formal, 0.0)
Type III Mixed Type I & II (19.0%)	I want to update blog on my laptop , with a dry martini on side.	Slot: Wifi = (free, 0.7), (paid, 0.3), (no, 0.0) Slot: Alcohol = (full_bar, 1.0), (beer_and_wine, 0.0), (don't_serve, 0.0)

Table 1: Examples of reasoning types. Type I utterance: G&T is only served in a full bar, while Riesling is a kind of wine and tonic water does not require alcohol options. Type II utterance, ‘place without people wearing ties’ indicates casual attire, and ‘five to ten dollars’ indicates a price range of cheap or affordable. Type III utterance, we need both kinds of reasonings.

NUANCED win	NUANCED-reduced win	Tied
54.7%	16.7%	28.6%

Table 2: Human evaluation results of comparing two versions.

The loss is a combination of cross-entropy loss for slot prediction and mean squared error (MSE) loss for value prediction. During inference, we set up a threshold to decide positive or negative predictions.

Transformer (Vaswani et al., 2017) We use the similar architecture as the BERT baseline but train the weights from scratch.

Train-ConvRex As MGConvRex dataset (Xu et al., 2020) has similar domain and ontology, we compare the BERT model trained on MGConvRex³ with that tested on NUANCED-reduced. We use this baseline to demonstrate the open challenges caused by users’ free-form speaking.

We refer the readers to Appendix A for more details. For all baselines, we evaluate on the turn level slot prediction accuracy and joint accuracy.

4.1.2 Results for NUANCED-reduced

As shown in Table 3, the BERT model achieves the best performance as the external knowledge obtained from pre-training helps draw a better relevance between unrecognized entities from the user and entities from the agent. Train-ConvRex limits such mapping to system ontology, indicating that existing dialogue datasets may limit what an agent can understand from users. Lastly, by comparing with BERT without dialogue context (or past turns), we notice that context may help in learning better values but yields more noise for slot prediction.

4.2 NUANCED

4.2.1 Baselines

Exact match & Random guess Similar to NUANCED-reduced, we assign a probability of 1.0

³We contacted the first author to obtain the dataset.

Baselines	Slot Accuracy (%)	Joint Accuracy (%)
Exact match & Random guess	48.83	4.84
Train-ConvRex	38.70	4.02
Transformer	74.14	21.52
BERT	88.21	36.56
BERT w/o context	88.78	34.99

Table 3: Results on NUANCED-reduced. *Slot Accuracy*: percentage of turns that all slots are correct; *Joint Accuracy*: percentage of turns that all slots and values are correct.

for matched values or random value otherwise.

BERT, Transformer Similar to NUANCED-reduced, we use MSE loss between the ground truth and the predicted distribution.

Train-reduced-X We train the model on NUANCED-reduced and test on NUANCED to see how data with binary states can infer states in the continuous space. We define a fixed number of X as the continuous number for all positive predictions. We experiment with X = 0.5 and 1.0.

We keep the same evaluation for slot prediction. For value predictions, we evaluate the *soft* average mean absolute error (MAE) between the ground truth distribution and the predictions.

4.2.2 Results for NUANCED

As in Table 4, BERT reaches the best performance. One interesting observation is that using the same model BERT, the slot prediction accuracy increases (from 88.21% to 89.62%) compared with training on the reduced version. NUANCED helps to reduce the noise of sparse entities in context (past turns). This is probably because numbers in continuous space can draw more relevance among different entities. As we can see, Train-reduced-X has a much larger error. This indicates that simply adapting the results from the reduced state labels suffers from information loss, i.e., the nuanced differences in continuous distributions.

Baselines	Slot Accuracy (%)	Correct slots mean MAE (1e-2)
Exact match & Random guess	48.83	46.84
Train-reduced-1.0	88.21	40.72
Train-reduced-0.5	88.21	21.62
Transformer	78.42	16.78
BERT	89.62	14.20
BERT w/o context	88.08	14.49

Table 4: Evaluation results on NUANCED. *Correct slots mean MAE* (lower the better): mean absolute error of predicted distribution for all correctly predicted updated slots;

4.2.3 Analysis on Slots

We study how the models perform on different kinds of turns, shown in Table 5. Generally speaking, the turns with more slots are relatively harder to learn. The turns that update the preference in previous turns have the highest error, the preference distribution needs to be jointly inferred from the previous mention and the current turn. We also study the performance on each slot in Appendix B, and provide some case studies in Appendix C.

Type of turn	all	1 slot	2 slots	3 slots	updating preferences
Slot Accuracy(%)	89.62	96.67	78.91	67.65	90.61
Mean MAE(1e-2)	14.12	14.06	13.55	14.20	15.63

Table 5: Performance for different kinds of slots: *all*: all kinds of turns; *n slots*: turns that the user utterance jointly implies *n* slots; *updating preferences*: turns that the user utterance updates the preference in previous turns.

4.2.4 Human Evaluation

We further conduct a human evaluation on baseline models. We first evaluate the model outputs of Transformer, BERT, and BERT w/o context, through pairwise comparison between the model predictions and the gold labels. The results on 200 samples are shown in Table 6. There is a large gap between the best-performing baseline and the gold reference, which indicates significant room for improvement for future research. Further, we study the breakdown of predictions of BERT on 3 different types of reasoning. As shown in Table 7, the type 1 utterances, that involve factoid knowledge, are relatively harder to learn. This is close to our intuition because factoid knowledge is huge (and keeps increasing) and the limited utterances in the dataset may not cover all of the knowledge.

5 Conclusion and Open Problems

Starting from our dataset, we believe the user-centric dialogue system is an open-ended problem and the following directions are worth pursuing:

Methods	Model output win(%)	Tied(%)	Gold win(%)
Transformer	10	9.5	80.5
Bert	23.6	20.9	55.4
Bert w/o context	19.5	9.6	70.9

Table 6: Human evaluation results for the model predictions.

Methods	Model output win(%)	Tied(%)	Gold win(%)
Type I	22.5	19.9	57.6
Type II	27.4	24.1	48.5
Type III	21.1	11.2	67.7

Table 7: Human evaluation results for different reasoning types. Type I: factoid knowledge; Type II: commonsense knowledge or user situations; Type III: Mixed Type I & II.

1) Preliminary experimental results indicate that to improve performance, it is promising to incorporate external domain texts into pre-trained models, for example, pre-training the model on domain corpora like restaurant descriptions and reviews. 2) Although our dataset collects a large set of domain entity knowledge, we still cannot guarantee that it will cover the vast amount of unknown entities in the future. One idea is to incorporate a knowledge base (KB) in the form of data augmentation or modeling. 3) Through our large-scale dataset, although one can learn a general agreement of estimated distributions from the crowds, a more user-specific distribution would be more desirable. We believe providing a personalized solution is another proper next step to consider.

6 Ethical Considerations

For our data annotation, our annotators were hired as full-time employees through a leading annotation services vendor, and were paid in accordance with a fair wage rate.

Acknowledgment

We thank Becka Silvert, Gerald Demeunynck, and Linnea Ross for helping with the data annotation process. We thank the anonymous reviewers for their thoughtful comments.

References

Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*.

- Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. [Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 5016–5026. Association for Computational Linguistics.
- Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. [Semantically conditioned dialog response generation via hierarchical disentangled self-attention](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3696–3709. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.
- Ondrej Dusek, Jekaterina Novikova, and Verena Rieser. 2018. [Findings of the E2E NLG challenge](#). In *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pages 322–328. Association for Computational Linguistics.
- Zuohui Fu, Yikun Xian, Yongfeng Zhang, and Yi Zhang. 2020. [Tutorial on conversational recommendation systems](#). In *RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020*, pages 751–753. ACM.
- Rashmi Gangadharaiah and Balakrishnan Narayanaswamy. 2019. [Joint multiple intent detection and slot labeling for goal-oriented dialog](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 564–569. Association for Computational Linguistics.
- Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishhauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. [Trippy: A triple copy strategy for value independent neural dialog state tracking](#). In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGdial 2020, 1st virtual meeting, July 1-3, 2020*, pages 35–44. Association for Computational Linguistics.
- Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. [A simple language model for task-oriented dialogue](#). *CoRR*, abs/2005.00796.
- Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. [Interactive path reasoning on graph for conversational recommendation](#). In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 2073–2083. ACM.
- Shijun Li, Wenqiang Lei, Qingyun Wu, Xiangnan He, Peng Jiang, and Tat-Seng Chua. 2020. [Seamlessly unifying attributes and items: Conversational recommendation for cold-start users](#). *CoRR*, abs/2005.12979.
- Bing Liu and Ian Lane. 2016. [Attention-based recurrent neural network models for joint intent detection and slot filling](#). In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 685–689. ISCA.
- Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Çelikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. [Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2231–2240. Association for Computational Linguistics.
- Gustavo Penha and Claudia Hauff. 2020. [What does BERT know about books, movies and music? probing BERT for conversational recommendation](#). In *RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020*, pages 388–397. ACM.
- Abhinav Rastogi, Dilek Hakkani-Tür, and Larry P. Heck. 2017. [Scalable multi-domain dialogue state tracking](#). In *2017 IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2017, Okinawa, Japan, December 16-20, 2017*, pages 561–568. IEEE.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. 2020. [Recipes for building an open-domain chatbot](#). *arXiv preprint arXiv:2004.13637*.
- Pei-Hao Su, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Stefan Ultes, David Vandyke, Tsung-Hsien Wen, and Steve J. Young. 2016. [Online active reward learning for policy optimisation in spoken dialogue systems](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Yueming Sun and Yi Zhang. 2018. [Conversational recommender system](#). In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 235–244. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Peihao Su, David Vandyke, and Steve J. Young. 2015. [Semantically conditioned lstm-based natural language generation for spoken dialogue systems](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1711–1721. The Association for Computational Linguistics.

Tsung-Hsien Wen, Yishu Miao, Phil Blunsom, and Steve J. Young. 2017. [Latent intention dialogue models](#). In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3732–3741. PMLR.

Jason D. Williams, Matthew Henderson, Antoine Raux, Blaise Thomson, Alan W. Black, and Deepak Ramachandran. 2014. [The dialog state tracking challenge series](#). *AI Mag.*, 35(4):121–124.

Hu Xu, Seungwhan Moon, Honglei Liu, Bing Liu, Pararth Shah, Bing Liu, and Philip S. Yu. 2020. [User memory reasoning for conversational recommendation](#). *CoRR*, abs/2006.00184.

Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. [Towards conversational search and recommendation: System ask, user respond](#). In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 177–186. ACM.

Appendix

A Model Implementation and Training Details

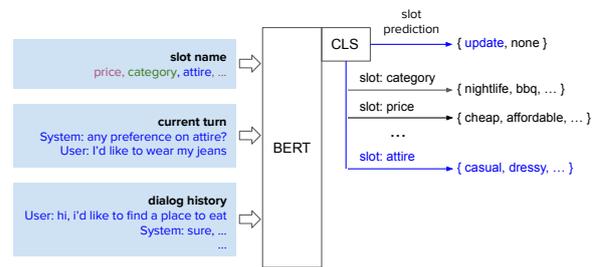


Figure 2: Illustration of the BERT baseline

Figure 2 presents the architecture of the BERT baseline. For each turn, we concatenate each slot with the current turn and the dialogue context as the input. On the [CLS] output, we add one head for slot prediction as binary classification, i.e., whether the input slot is updated in the current turn. For each slot, we add a specific head for value prediction. We use cross entropy loss for slot prediction, and mean squared loss for value distribution prediction. The overall loss is a weighted combination of the two losses. We set the weight for value prediction as 20.0. The threshold for value prediction in NUANCED-reduced is set as 0.5. We use BERT-base uncased model from the official release⁴ with 110M parameters; The learning rate is set as 3e-5, batch size as 32. We take the results based on the performance on validation set. For NUANCED-reduced, the training takes around 25,000 gradient steps; For NUANCED, the training takes around 40,000 steps. For the transformer model, to achieve best performance we use 6 layers and hidden size 300. All training is done on a single NVIDIA TESLA M40 card with 11G memory.

Note that for the slot “food category”, some values are commonly observed in the dataset such as “American food”, “nightlife”, while some others are less frequently such as “Thai”. During training we employ up-sampling for the less frequent ones.

In the construction of NUANCED, we sample a subset of the user history and aggregate to get the ground truth preference distributions. Because the number of viable values of each slot is different, for those slots with relatively more values the distribution generally presents ‘long tail’, we only take the top 3 value distributions for each slot. Correspondingly, during the model evaluation, we also take the

⁴<https://github.com/google-research/bert>

top 3 predicted value distributions to calculate the MAE.

B Analysis on Slots

We also study how the model performs on each slot in the domain, shown in Table 8. Generally, slots that may involve more factoid knowledge or more choices of values are harder to learn, such as `food category`, `parking`. These may require learning long-tailed knowledge from external data.

Slot	<code>food category</code>	<code>price</code>	<code>parking</code>	<code>noise</code>
Mean MAE(1e-2)	15.48	15.29	16.94	13.34

Slot	<code>ambience</code>	<code>alcohol</code>	<code>wifi</code>	<code>attire</code>
Mean MAE(1e-2)	15.04	13.88	12.30	8.95

Table 8: Performance for each slot of our dataset.

C Case Studies

Table 9 provides some case studies with ground truth and the BERT model predictions.

Dialogue Turns	NUANCED-reduced	NUANCED
Assistant: any preference on attire? User: I like shorts and a loose tee shirt in this heat.	Gold labels: Attire (casual= 1, dressy= 0, formal= 0) BERT predictions: Attire (casual= 1, dressy= 0, formal= 0)	Gold Distributions: Attire (casual= 1.00, dressy= 0.00, formal= 0.00) BERT predictions: Attire (casual= 0.99, dressy= 0.01, formal= 0)
Assistant: what type of food would you like? User: Ribs would be perfect.	Gold labels: Category (traditional_american= 1.0, bbq= 1.0, nightlife= 0.0) BERT predictions: category (traditional_american= 1.0, nightlife= 1.0, new_american= 0.0)	Gold Distributions: Category (traditional_american= 0.50, bbq= 0.50, nightlife= 0.00) BERT predictions: Category (traditional_american= 0.20, nightlife= 0.08, new_american= 0.09)
Assistant: any preference on alcohol? User: I really want a G&T or a Riesling, but I could also have a tonic water.	Gold labels: alcohol (full_bar= 1.0, beer_and_wine= 1.0, don't_serve= 1.0) BERT predictions: alcohol (full_bar= 1.0, beer_and_wine= 1.0, don't_serve= 1.0)	Gold Distributions: alcohol (full_bar= 0.78, beer_and_wine= 0.33, don't_serve= 0.11) BERT predictions: alcohol (full_bar= 0.55, beer_and_wine= 0.47, don't_serve= 0.09)
Assistant: what parking option would you like? User: I need something fuss-free and out of the rain for my car. Also, I really want a gin and tonic, but it's not a complete deal-breaker if I can't have it. (after some turns)	Gold labels: parking (garage= 1.0, valet= 0.0, validated= 0.0) alcohol (full_bar= 1.0, beer_and_wine= 1.0, don't_serve= 1.0) BERT predictions: parking (garage= 1.0, valet= 1.0, lot= 1.0) alcohol (full_bar= 1.0, beer_and_wine= 1.0, don't_serve= 1.0)	Gold Distributions: parking (garage= 0.86, valet= 0.00, validated= 0.00) alcohol (full_bar= 0.93, beer_and_wine= 0.21, don't_serve= 0.14) BERT predictions: parking (garage= 0.78, valet= 0.41, lot= 0.34) alcohol (full_bar= 0.79, beer_and_wine= 0.17, don't_serve= 0.12)
Assistant: here're the recommendations. User: You know what, if it's going to be a fancier place then I don't mind dealing with more complicated parking after all.	Gold labels: parking (garage= 1.0, valet= 1.0, validated= 1.0) BERT predictions: parking (garage= 1.0, lot= 1.0, validated= 1.0)	Gold Distributions: parking (garage= 0.86, valet= 0.64, validated= 0.21) BERT predictions: parking (garage= 0.67, valet= 0.48, lot= 0.40)

Table 9: Some case studies. the last example shows two turns in a dialogue and corresponding distributions for each turn. The user updates the preference in a later turn based on a previous turn.