Improving Dialogue State Tracking by Discerning the Relevant Context

Sanuj Sharma, Prafulla Kumar Choubey, Ruihong Huang
Department of Computer Science and Engineering
Texas A&M University
(sanuj, prafulla.choubey, huangrh)@tamu.edu

Abstract

A typical conversation comprises of multiple turns between participants where they go back-and-forth between different topics. At each user turn, dialogue state tracking (DST) aims to estimate user's goal by processing the current utterance. However, in many turns, users implicitly refer to the previous goal, necessitating the use of relevant dialogue history. Nonetheless, distinguishing relevant history is challenging and a popular method of using dialogue recency for that is inefficient. We, therefore, propose a novel framework for DST that identifies relevant historical context by referring to the past utterances where a particular slot-value changes and uses that together with weighted system utterance to identify the relevant context. Specifically, we use the current user utterance and the most recent system utterance to determine the relevance of a system utterance. Empirical analyses show that our method improves joint goal accuracy by 2.75% and 2.36% on WoZ 2.0 and MultiWoZ 2.0 restaurant domain datasets respectively over the previous state-of-the-art GLAD model.

1 Introduction

Dialog state tracking (DST) is a vital component in the task-oriented dialog systems which is used to estimate user's goals and requests in order to plan next action and respond accordingly. At each turn, DST aims to identify the set of goals that a user aims to achieve and requests that are represented as slot-value pairs. Typically, this decision is made by considering user utterance in the current turn or system actions in the previous turn. However, in many cases, the considered user utterance or system actions do not present enough information and refers to a previous utterance.

As shown through an example in Figure 1, while exploring different available options, user

User: hello, i'm looking for a restaurant with fair prices
System: There are 31 places with moderate price range. Can you please tell me what kind of food you would like?
Turn Label: price range = moderate **Sys Act:** food

User: well I want to eat in the North, what's up that way?
System: I have two options that fit that description, Golden Wok chinese restaurant and Nirala which serves Indian food. Do you have a preference?
Turn Label: area = north **Sys Act:** food

User: Can I have the address and phone number for the Golden Wok chinese restaurant?
System: The phone number is 01223 350688.
Turn Label: request = address, phone; food = chinese

User: thank you. what is the address?
System: The address is 191 Histon Road Chesterton.
Turn Label: request = address

User: Okay, what about Nirala, what's the address and phone of that?
System: 7 Milton Road Chesterton and the number is 01223 360966
Turn Label: request = address, phone; food = indian

Figure 1: An example dialog from WoZ 2.0 dataset. A turn contains user utterance (blue), system utterance (red), system actions (yellow) and turn label (green). Each turn is separated by a line.

can go back-and-forth between the currently and previously discussed facts. For instance, when offered with two different restaurant options namely *Nirala* (`food=indian`) and *Golden Wok* (`food=chinese`) in the second turn, user first inquires about the details of *Golden Wok*. And after getting relevant details about the *Golden Wok* in the following two turns, user refers back to the second option provided in second turn and asks about *Nirala* restaurant. To predict the correct slot-value pair `food=indian` in the dialog state of the fifth turn, the system is required to refer back to the second turn again to find information about *Nirala*, as the context obtained from the current dialog turn is insufficient.

Identifying such implicitly referenced historical turns is challenging since implicit references are

not local and most recent turns are often not informative. Therefore, the traditional approach of modeling dialogue recency (El Asri et al., 2017) may not suffice. Instead, we propose to model implicit references by storing links to the past turn where each of the slots was modified. Then at each turn, we look up though the stored links to find the previous turn which may provide additional cues for predicting the appropriate slot-value.

Moreover, the dialogue system often asks polar questions with yes-no answers. For instance, the DST system should update the dialogue state with `food=indian` when a user replies *Yes* to a system utterance *Do you want Indian food?*. In such cases, neither the user utterance nor system acts (*food* in this example) contain any information about the actual slot-value. This makes utilization of both system and user utterance eminent for dialog state tracking. However, utilizing the previous system utterance together with the current user utterance always at each turn may add noise. Therefore, we use a gating mechanism based on both utterances to determine the relevance of the previous system utterance in the current turn.

The evaluation shows that identifying the relevant context is essential for dialogue state tracking. Our novel model that discerns important details in non-adjacent dialogue turns and the previous system utterance from a dialog history is able to improve the previous state-of-the-art GLAD (Zhong et al., 2018) model on all evaluation metrics for both WoZ and MultiWoZ (restaurant) datasets. Furthermore, we empirically show that a simple self-attention based biLSTM model, using only one-third of the number of parameters as GLAD, outperforms GLAD by identifying and incorporating the relevant context.

2 Related Work

Early work for DST relied on separate Spoken Language Understanding (SLU) module (Henderson et al., 2012) to extract relevant information from user utterances in a pipelined approach. Such systems are prone to error accumulation from a separate SLU module, in absence of necessary dialog context required to interpret the user utterance. Thus, later work on DST moved away from separate SLU modules and inferred the dialog state directly from user utterance and dialog history (Henderson et al., 2014b,c; Zilka and Jurcicek, 2015). These models depend on delexicalization, using generic tags to replace specific slot types and values, and handcrafted semantic dictionaries. In practice, it is difficult to scale these models for every slot type and recent state-of-the-art models for DST use deep learning based methods to learn general representations for user and system utterances and previous system actions, and predict the turn state (Henderson et al., 2013, 2014b; Mrkšić et al., 2015, 2017; Hori et al., 2016; Liu and Lane, 2017; Dernoncourt et al., 2017; Chen et al., 2016). However, these systems are found to perform poorly on rare and unknown slot-value pairs which was recently addressed through local slot-specific encoders (Zhong et al., 2018) and pointer network (Xu and Hu, 2018).

A crucial limitation to all these approaches lies in the modeling of appropriate historical context, which is simply ignored in most of the works. Since user's goal may change back-and-forth between previous values, incorporating relevant historical context is useful in monitoring implicit goal references. In a recent work, El Asri et al. (2017) discussed on similar limitations of current DST task and introduced a new task of frame tracking that explicitly tracks every slot-values that were introduced during the dialogue. However, that significantly complicates the task by maintaining multiple redundant frames that are often left unreferenced. Our proposed model, that explicitly track relevant historical user and system utterances, can be easily incorporated into any known DST or frame tracking systems such as Schulz et al. (2017) to replace the recency encoding.

3 Discerning Relevant Context for DST

Similar to previous works, we decompose the multi-label classification problem to binary classification where we score each slot-value pair and select the ones that receive a score above a threshold to be included in the current dialog state. To predict the score for a candidate slot-value pair, the model uses the relevant past user utterance (referential utterance), a fused utterance composed using the current user utterance and the system utterance of the previous turn, as well as previous system actions as evidence. Shown in Figure 2, our model comprises of:

Lookup module: retrieves a link to the turn where each of the slots changes. At each step, our system refers to the lookup module that returns the past user utterance (the "*antecedent user utterance*")

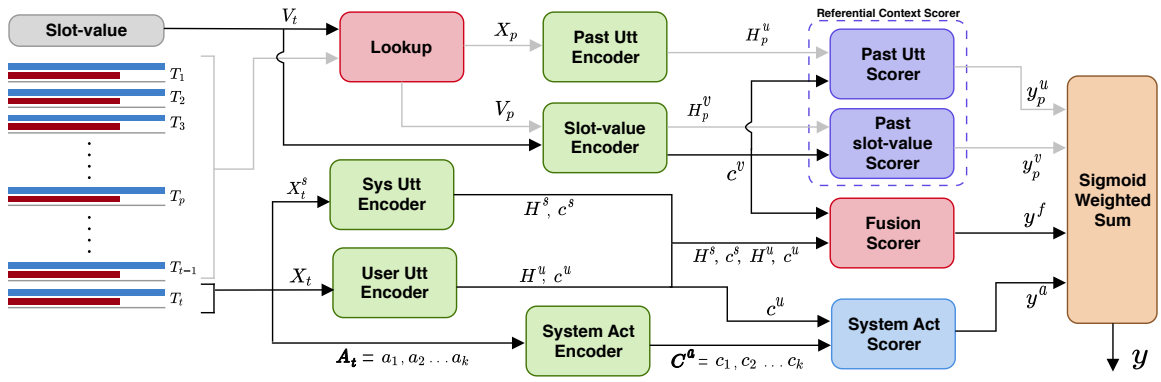

Figure 2: The Architecture of Context Aware Dialogue State Tracker.

where the candidate slot-type was modified as well as outputs the previous slot-value.

GLE modules: Each of the five green modules in Figure 2 is a global-locally self-attentive encoder (GLE module) (Zhong et al., 2018) that encodes each type of evidence into a vector representation (c). Each input is represented as a sequence of words which is encoded to a vector representation via global-local self-attentive encoder (GLE) module (Zhong et al., 2018). Specifically, GLE employs local slot-specific bidirectional LSTMs and a global bidirectional LSTM (Hochreiter and Schmidhuber, 1997) that is shared across all slots for encoding the input sequence into a sequence of hidden states (H), followed by a self-attention layer (Lin et al., 2016) to obtain a fixed dimension vector representation (c).

The GLE modules are used to encode the antecedent user utterance (H^u_p, c^u_p), the current user utterance (H^u, c^u), the previous system utterance (H^s, c^s), each of the system acts (H^{a_i}, c^{a_i}), as well as the previous slot-value (H^v_p, c^v_p) and the candidate (H^v, c^v) slot-value.

Referential Context Scorer: uses the candidate slot value (c^v), the antecedent user utterance as well as the previous slot-value to determine if the candidate slot value was referenced in the antecedent utterance. Specifically, the scorer uses the representation of the candidate slot value c^v to attend over hidden states of the antecedent user utterance and the previous slot-value, H^u_p and H^v_p, and then computes attention weights for each of the hidden states. Next, the scorer sums up the hidden states weighed with the calculated attentions to get the summary context (Equation 1). Finally, the scorer applies a linear neural layer to calculate the scores y^v_p and y^u_p representing the likelihoods that the candidate slot-value is different from the previous slot-value and the candidate slot-value was unreferenced in the antecedent utterance (Equation 2).

$$Q(H, c): \; a_j = (H_j)^\top c \; ; \; p = softmax(a)$$
$$Q(H, c) = \sum_i p_i H_i \quad (1)$$

$$y^u_p = W^u_p \, Q(H^u_p, c^v) + b^u_p$$
$$y^v_p = W^v_p \, Q(H^v_p, c^v) + b^v_p \quad (2)$$

Fusion Scorer: leverages necessary details in the previous system utterance to enrich the current user utterance. First, we use a gating mechanism based on c^s and c^u that determines the relevance of the previous system utterance in the current turn. We concatenate c^s and c^u and use a linear layer with sigmoid activation to calculate the score α (Equation 3). Then, we use attention from c^v over H^s and H^u to calculate context summaries (l^s, l^u), and combine the summary vectors by taking their normalized weighted sum based on α. We finally apply a single linear layer to calculate the score y^f that determines the likelihood of the candidate slot-value based on both the current user utterance and the previous system utterance (Equation 4).

$$f_c = W_{fc}(c^s \oplus c^u) + b_{fc}$$
$$\alpha = \sigma(W_\alpha tanh(f_c) + b_\alpha) \quad (3)$$

$$l^s = Q(H^s, c^v) \; ; \; l^u = Q(H^u, c^v)$$
$$l^f = \alpha l^s + (1 - \alpha) l^u \; ; \; y^f = W_{lf} l^f + b_{lf} \quad (4)$$

System Act Scorer: is the same as the action scorer proposed by (Zhong et al., 2018). Specifically, The scorer uses attention from c^u over C^a to calculate action summary followed by a linear layer with sigmoid activation to calculate the score y^a that determines the relevance of the candidate

slot-value based on the previous system actions (Equation 5).

$$l^a = Q(C^a, c^u)\,; \quad y^a = (l^a)^\top c^v \qquad (5)$$

It then calculates the final score of the candidate slot-value by taking weighted sum of the four scores (y_p^u, y_p^v, y^f, y^a) followed by a sigmoid layer, where weights are learned in the network.

4 Evaluations

4.1 Experimental Setup

We primarily use WoZ 2.0 (Wen et al., 2017) restaurant reservation task dataset that consists of 1200 dialogues for training and evaluation. Each dialogue has an average of eight turns, where each turn contains *system utterance transcript*, *user utterance transcript*, *turn label* and *belief state*. All the dialogue states and actions are based on a task ontology that supports three different informable slot-types namely *price range* with 4 values, *food* with 72 values, *area* with 7 values, and *requests* of 7 different types like *address* and *phone*. Following the standard settings, we use 600 dialogues for training, 200 for validation and the remaining 400 for testing.

We also use dialogues from restaurant domain in MultiWoZ 2.0 dataset (Budzianowski et al., 2018) for secondary evaluation. It banks on a significantly complex ontology covering seven informable slot types with 276 different values (*food, price range, restaurant name, area, book time, book day* and *book people* with 97, 6, 105, 8, 43, 8 and 9 values respectively). We use standard training, validation and test splits of 1199, 50 and 61 dialogues respectively.

All the models on WoZ 2.0 are evaluated on the two standard metrics introduced in Henderson et al. (2014a). First, **Joint Goal Accuracy** is the percentage of turns in a dialogue where the user's informed joint goals are identified correctly. Joint goals are accumulated turn goals up to the current dialog turn. Second, **Turn Request Accuracy** calculates the percentage of turns in a dialogue where the user's requests were correctly identified. Models on MultiWoZ 2.0 dataset are evaluated using joint goal and turn inform accuracies, as used by Nouri and Hosseini-Asl (2018).

4.2 Implementation Details

We use pretrained GloVe word embeddings (Pennington et al., 2014) concatenated with character n-gram embeddings (Hashimoto et al., 2017) which are kept fixed during the training. Each of bi-LSTMs use 200 hidden dimensions. All the models are trained using ADAM optimizer (Kingma and Ba, 2014) with the initial learning rate of 0.001. Dropout rate (Srivastava et al., 2014) is set to 0.2 for all biLSTM modules and the embedding layer. The models are trained for a maximum of 100 epochs with a batch size of 50. The validation data was used for early stopping and hyperparameter tuning.

Model	WoZ 2.0 Joint Goal	WoZ 2.0 Turn Request
Delexalisation-Based Model + SD	83.7%	87.6%
NBT - DNN	84.4%	91.2%
NBT - CNN	84.2%	91.6%
GLAD †	**86.4%**	**97.1%**
Global biLSTM based GLE	85.0%	96.8%
Global biLSTM based GLE + RC	87.4%	**97.0%**
Global biLSTM based GLE + RC + FS	**88.4%**	**97.0%**
GLAD + RC + FS	**89.2%**	**97.4%**

Table 1: Test accuracy of baselines and proposed models on WoZ 2.0 restaurant reservation dataset. †Retrained using docker container provided by the authors with exactly same hyper-parameters. We also experimented with different versions of PyTorch and cuDNN and found that results had high variance. Therefore, we report the average performance over 5 runs with different initializations for GLAD and all our models.

Model	MultiWoZ 2.0 (Restaurant) Joint Goal	MultiWoZ 2.0 (Restaurant) Turn Inform
GLAD	43.95%	76.99%
GLAD + RC	**45.72%**	**77.87%**
GLAD + RC + FS	**46.31%**	**78.76%**

Table 2: Test accuracy of GLAD and proposed models on MultiWoZ 2.0 restaurant domain dataset. Note that we considered all 276 slot-values for evaluating models. Budzianowski et al. (2018) reported joint goal accuracy of 80.9 on MultiWoZ 2.0 (restaurant) dataset. We believe they didn't include *restaurant name* slot in their evaluation and only considered presence of three slot-types—*book time, book day* and *book people*—and not their values.

4.3 Results

Table 1 compares the performance of our proposed models with different baselines, including **delexalisation-based model + SD** (Wen et al., 2017), DNN and CNN variants of **neural belief tracker** (Mrkšić et al., 2017) and the previous state-of-the-art GLAD systems (Zhong et al., 2018) on WoZ 2.0 dataset. We also implement a simplified variant of GLAD, **Global BiLSTM**

Model	Approx. # of parameters
Global biLSTM based GLE	1.2 million
Global biLSTM based GLE + RC + FS	**6 million**
GLAD	17 million
GLAD + RC + FS	28 million

Table 3: Number of learnable parameters for different models on WoZ 2.0 dataset

based GLE, by removing slot-specific local biLSTMs from the GLE encoder. We then successively combine it with referential context (**Global biLSTM based GLE + RC**) and the fused previous system utterance (**Global biLSTM based GLE + RC + FS**). Finally, we directly incorporate the referential context and gate selected system utterance into the GLAD system (**GLAD + RC + FS**).

Irrespective of the underlying system, utilizing appropriate context from the previous turns improves the overall performance of a dialogue state tracker on both joint goal and turn request accuracies on WoZ 2.0 dataset. First, incorporating relevant referential utterances to identify implicitly mentioned slot-value improves the accuracy of global biLSTM based GLE model on joint goal task by 2.4%. Then, gating based mechanism to augment user utterance with relevant information from the previous system utterance further improves the joint goal accuracy by 1.0%. Together, they improve joint goal and request accuracy of the global biLSTM based GLE model by 3.4% and 0.2% respectively. Furthermore, as evident from the results in Table 2, both referential context and fused system utterance proportionally improve performance on MultiWoZ 2.0 dataset as well with overall improvement of 2.36% and 1.77% on joint goal and turn inform accuracies respectively. Performances of all models on MultiWoZ 2.0 are significantly inferior compared to WoZ 2.0 owing to higher complexity, with richer and longer utterances and considerably more slot-values in the former dataset.

5 Analysis

The utilization of relevant context results in significant reduction in the number of learnable parameters in the model as shown in Table 3. Relevant context with the baseline model is able to outperform GLAD while using only one third of the number of learnable parameters. The parameters added due to using relevant context are the parameters for encoding the antecedent referential user utterance and the previous system utterance as well as the past utterance and past slot-value scorers. However, we also observe high variance in the joint goal accuracy. Since joint goal is calculated by accumulating turn goals, an error in predicting a turn goal is propagated to all the downstream turns.

6 Conclusion

We have presented a novel method for identifying the relevant historical user utterance as well as determining the relevance of the system utterance from the last turn to enrich the current user utterance and improve goal tracking in dialogue systems. The experimental results show that discerning relevant context from the dialog history is crucial for tracking dialog states.

Acknowledgments

We want to thank our anonymous reviewers for providing insightful review comments.

References

Paweł 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*, pages 5016–5026.

Yun-Nung Chen, Dilek Hakkani-Tür, Gökhan Tür, Jianfeng Gao, and Li Deng. 2016. End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *Interspeech*, pages 3245–3249.

Franck Dernoncourt, Ji Young Lee, Trung H Bui, and Hung H Bui. 2017. Robust dialog state tracking for large ontologies. In *Dialogues with Social Robots*, pages 475–485. Springer.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 207–219. Association for Computational Linguistics.

Kazuma Hashimoto, caiming xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple nlp tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933. Association for Computational Linguistics.

Matthew Henderson, Milica Gašić, Blaise Thomson, Pirros Tsiakoulis, Kai Yu, and Steve Young. 2012. Discriminative spoken language understanding using word confusion networks. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 176–181. IEEE.

Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014a. The third dialog state tracking challenge. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 324–329. IEEE.

Matthew Henderson, Blaise Thomson, and Steve Young. 2013. Deep neural network approach for the dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 467–471. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014b. Word-based dialog state tracking with recurrent neural networks. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 292–299. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014c. Word-based dialog state tracking with recurrent neural networks. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 292–299.

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

Takaaki Hori, Hai Wang, Chiori Hori, Shinji Watanabe, Bret Harsham, Jonathan Le Roux, John R Hershey, Yusuke Koji, Yi Jing, Zhaocheng Zhu, et al. 2016. Dialog state tracking with attention-based sequence-to-sequence learning. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 552–558. IEEE.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2016. A structured self-attentive sentence embedding.

Bing Liu and Ian Lane. 2017. An end-to-end trainable neural network model with belief tracking for task-oriented dialog. *arXiv preprint arXiv:1708.05956*.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gasic, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 794–799. Association for Computational Linguistics.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2017. Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788. Association for Computational Linguistics.

Elnaz Nouri and Ehsan Hosseini-Asl. 2018. Toward scalable neural dialogue state tracking model. *arXiv preprint arXiv:1812.00899*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Hannes Schulz, Jeremie Zumer, Layla El Asri, and Shikhar Sharma. 2017. A frame tracking model for memory-enhanced dialogue systems. *arXiv preprint arXiv:1706.01690*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M. Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449. Association for Computational Linguistics.

Puyang Xu and Qi Hu. 2018. An end-to-end approach for handling unknown slot values in dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1448–1457. Association for Computational Linguistics.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Global-locally self-attentive encoder for dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1458–1467. Association for Computational Linguistics.

Lukas Zilka and Filip Jurcicek. 2015. Incremental lstm-based dialog state tracker. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 757–762. IEEE.