Contextual Domain Classification with Temporal Representations

Tzu-Hsiang Lin, **Yipeng Shi**, **Chentao Ye**, **Fan Yang**, **Weitong Ruan**, **Emre Barut**, **Wael Hamza**, and **Chengwei Su**

Amazon Alexa AI

{tzuhsial,syipeng,ychentao,fyaamz,weiton,ebarut,waelhamz,chengwes}@amazon.com

Abstract

In commercial dialogue systems, the Spoken Language Understanding (SLU) component tends to have numerous domains thus context is needed to help resolve ambiguities. Previous works that incorporate context for SLU have mostly focused on domains where context is limited to a few minutes. However, there are domains that have related context that could span up to hours and days. In this paper, we propose temporal representations that combine wall-clock second difference and turn order offset information to utilize both recent and distant context in a novel large-scale setup. Experiments on the Contextual Domain Classification (CDC) task with various encoder architectures show that temporal representations combining both information outperforms only one of the two. We further demonstrate that our contextual Transformer is able to reduce 13.04% of classification errors compared to a non-contextual baseline. We also conduct empirical analyses to study recent versus distant context and opportunities to lower deployment costs.

1 Introduction

Voice assistants such as Amazon Alexa, Apple Siri, Google Assistant and Microsoft Cortana provide a wide range of functionalities, including listening to music, inquiring about the weather, controlling home appliances and question answering. To understand user requests, the Spoken Language Understanding (SLU) component needs to first classify an utterance into a domain, followed by identifying the domain-specific intent and entities (Tur, 2011; Su et al., 2018a), where each domain is defined for a specific application such as music or weather. In commercial systems, the number of domains tend to be large, resulting in multiple possible domain interpretations for user requests (Kim et al., 2018; Li et al., 2019). For example, *"play american pie"* can be interpreted as either playing a song or a movie. Also, *"what does your light color mean?"* can be classified as *Question Answering*, or as a complaint which does not necessarily require a meaningful response.

Multiple prior works have attempted to incorporate context in SLU to help resolve such ambiguities. However, these works often report results on datasets with limited amount of training data (Bhargava et al., 2013; Xu and Sarikaya, 2014; Shi et al., 2015; Liu et al., 2015), or resort to synthesize contextual datasets (Gupta et al., 2018, 2019) that may not reflect natural human interaction. Furthermore, the majority of these works focus on domains where session context is recent and collected within a few minutes. Though this setup works well for domains that bias towards immediate preceding context such as *Communication* (Chen et al., 2016) and *Restaurant Booking* (Henderson et al., 2014; Bapna et al., 2017), there are also domains that have useful context spanning over hours or even up to days. In the *SmartHome* domain, it is natural for users to turn on T.V., watch for a couple of hours and then ask to turn it off. In the *Notifications* domain, users setup alarms or timers which occur hours and days away. We hypothesize that distant context, if properly utilized, can improve performance in instances where recent context cannot.

In this paper, we propose temporal representations to effectively leverage both recent and distant context on the Contextual Domain Classification (CDC) task. We introduce a novel setup that contains both recent and distant context by including previous 9 turns of context within a few days, so that context not just come from minutes but can also come from hours or days ago. We then propose temporal representations to indicate the closeness of each previous turn. The key idea of our approach is to combine both wall-clock second difference (Conway and Mathias, 2019) and turn order offset (Su et al., 2018b) so that a distant previous turn can still be considered as important.

Proceedings of NAACL HLT 2021: IndustryTrack Papers, pages 41–48
June 6–11, 2021. ©2021 Association for Computational Linguistics

We conduct experiments on a large-scale dataset with utterances spoken by users to a commercial voice assistant. Results with various encoder architectures show that combining both wall-clock second difference and turn order offset outperforms using only one of them. Our best result is achieved with Transformer of 13.04% error reduction, which is a 0.35% improvement over using only wall-clock second difference and 2.26% over using only turn order offset. To understand the role of context in CDC, we conduct multiple empirical analyses that reveal the improvements from context and discuss trade-offs between efficiency and accuracy.

To summarize, this paper makes the following contributions:

- A novel large-scale setup for CDC that showcases the usefulness of distant context, comparing to previous works whose datasets are limited to thousands and context within minutes.
- Temporal representations combining wall-clock second and turn-order offset information that can be extended and applied to other tasks.
- Empirical analyses that study context from 4 different aspects to guide future development of commercial SLU.

2 Related Work

2.1 Contextual SLU

Context in commercial voice assistants may belong to widely different domains, as users expect them to understand their requests in a single utterance, which is different from the conventional dialogue state tracking task (Williams et al., 2016). Earlier works seek better representations of context, such as using recurrent neural networks (Xu and Sarikaya, 2014; Liu et al., 2015), or memory networks to store past utterances, intents, and slot values (Chen et al., 2016). Recently, Gupta et al. (2019) proposes a self-attention architecture that fuses multiple signals including intents and dialog act with a variable context window. On other aspects of contextual SLU, Naik et al. (2018) proposes a scalable slot carry over paradigm where the model decides whether a previous slot value is referred in the current utterance. For rephrased user requests, Rastogi et al. (2019) formulates rephrasing as the Query Rewriting (QR) task and uses sequence-to-sequence pointer generator networks to perform both anaphora resolution and DST. In contrast, our work proposes temporal representations to utilize both recent and distant context for domain classification.

2.2 Temporal Information

Most previous works use recurrent neural networks to model natural turn order (Shi et al., 2015; Gupta et al., 2018). Assuming context follows a decaying relationship, Su et al. (2018b) presents several hand-crafted turn-decaying functions to help the model focus on the most recent context. Kim and Lee (2019) further expands upon this idea by learning latent turn-decaying functions with deep neural networks. On the other hand, wall-clock information has not been exploited until the recent *Time Mask* module proposed in Conway and Mathias (2019). From the lens of wall-clock, they show that context importance does not strictly follow a decaying relationship, but rather occurs in certain time spans. Our work combines both wall-clock and turn order information and models their relationship.

3 Methodology

In this section, we describe our model architecture in Section 3.1 and our proposed temporal representations in Section 3.2.

3.1 Model Architecture

Our model is depicted in Figure 1 and consists of 3 components: (1) utterance encoder, (2) context encoder, and (3) output network. We next describe each component in detail.

Utterance Encoder We use a bi-directional LSTM (Hochreiter and Schmidhuber, 1997) and pre-trained word embeddings to encode the current utterance into an utterance embedding. For pre-trained word embeddings, we use FastText (Bojanowski et al., 2017) concatenated with Elmo (Peters et al., 2018) trained on an internal SLU dataset.

Context Encoder Context encoder is a hierarchical model that consists of a turn encoder and a sequence encoder. For each previous turn, turn encoder encodes 3 types of features: (1) utterance text, (2) hypothesized domain, and (3) hypothesized domain-specific intent, which are also used in Naik et al. (2018). Utterance text is encoded using the same model architecture as in utterance

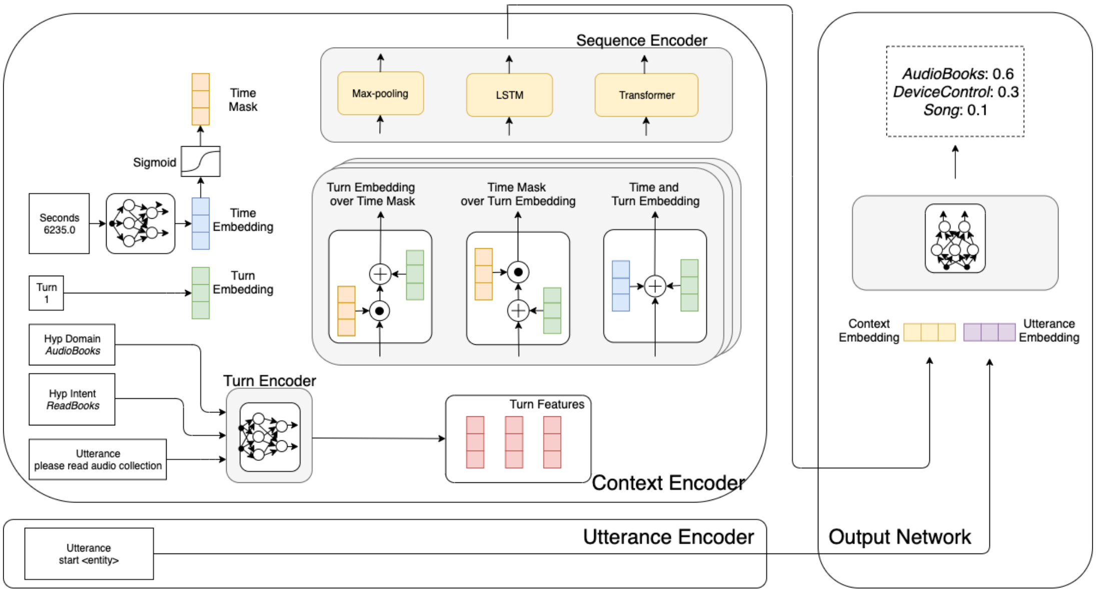


Figure 1: Overview of our model and proposed temporal representations.

encoder. Hypothesized domain and intent are first represented using one-hot encoding then projected into embeddings. We stack the 3 representations, perform max-pooling then feed into a 2 layer fully connected neural network to produce a turn representation. Temporal representations (Section 3.2) are then applied to indicate their closeness. Finally, sequence encoder encodes the sequence of temporal encoded turn representations into a single context embedding that is fed to the output network.

Output Network Output network concatenates utterance embedding and context embedding as input and feeds into a 2 layer fully-connected network to produce classification logits.

Response Time Considerations State-of-the-art contextual models encode the entire context and utterance to learn coarse and fine relationships with attention mechanisms (Gupta et al., 2019; Heck et al., 2020). Since commercial voice assistants need to provide immediate responses to users, encoding context and utterance is computationally expensive such that the system would not respond in-time at industrial-scale (Kleppmann, 2017). We separate context encoder from utterance encoder so that we can encode context when user is idle or when the voice assistant is responding. Moreover, the hierarchical design allows us to cache previously encoded turn representations to avoid re-computation.

3.2 Temporal Representations

In this section, we present the temporal representations used in our experiments. For the following, given previous turn t and its turn features $h^t(c)$ from turn encoder, we denote its wall-clock second difference and turn order offset as $d_{\Delta sec}$, $d_{\Delta turn}$. For operators, we denote $\odot$ and $\oplus$ as element-wise multiplication and summation.

Time Mask (TM) (Conway and Mathias, 2019) feeds $d_{\Delta sec}$ into a 2 layer network and sigmoid function to produce a masking vector $m_{\Delta sec}$ that is multiplied with the context feature h_c^T, and show that important features occur in certain time spans. The equations are given as follows.

$$e_{\Delta sec} = W_{s2} \cdot \phi(W_{s1} \cdot d_{\Delta sec} + b_{s1}) + b_{s2}, \tag{1}$$

$$m_{\Delta sec} = \sigma(e_{\Delta sec}), \tag{2}$$

$$h^t_{TM}(c) = m_{\Delta sec} \odot h^t(c), \tag{3}$$

Here $W_{s1}, W_{s2}, b_{s1}, b_{s2}$ are weight matrices and bias vectors, ϕ and σ are ReLU activation and sigmoid functions, and $h^t_{TM}(c)$ denotes the time masked features. We also considered binning second differences instead of working with $d_{\Delta sec}$. However, we find that binning significantly underperforms compared to the latter.

Turn Embedding (TE) We first represent $d_{\Delta turn}$ as a one-hot encoding then project it into a fixed-size embedding $e_{\Delta turn}$. We then sum the turn embedding with context features as in positional

Temporal Representations	Max-pooling	LSTM	Transformer
–	4.41	11.02	10.18
Time Mask	7.62	11.91	12.69
Turn Embedding	7.09	11.44	10.78
Turn Embedding over Time Mask	4.59	**12.51**	**13.04**
Time Mask over Turn Embedding	7.56	12.21	12.75
Time and Turn Embedding	**10.13**	11.31	11.79

Table 1: ARER % (↑) results computed against an utterance-only baseline with different temporal representations and sequence encoders. "–" indicates that no temporal representation is applied. Best results are boldfaced.

encoding in Transformer (Vaswani et al., 2017).

$$h^t_{TE}(c) = e_{\Delta turn} \oplus h^t(c), \tag{4}$$

It is natural and intuitive to assume that a closer context is more likely to correlate with the current user request. Assuming we are given user requests *"Where is Cambridge?"* and *"How is the weather there?"*. It is more likely that the user is inquiring about weather in Cambridge if the second request immediately follows the first, compared to the case where these two requests are hours or multiple turns apart. For a proper comprehension of closeness, both wall-clock and turn order information are needed, as having the same wall-clock difference would require us to know the turn order difference, and vice versa. Here we propose 3 representations that combines the two information based on different hypotheses.

Turn Embedding over Time Mask (TEoTM) provides turn order information on top of seconds. We do so by first masking the context features using *Time Mask* then mark the relative order with *Turn Embedding*. This variant assumes that the past context is important despite the fact that they might be distant in seconds.

$$h^t_{TEoTM}(c) = e_{\Delta turn} \oplus (m_{\Delta sec} \odot h^t(c)), \tag{5}$$

Time Mask over Turn Embedding (TMoTE) applies wall-clock second and turn offset information in reverse order of *TEoTM* by first summing *Turn Embedding* and then multiplying it with *Time Mask*. This assumes that second is more important than turn order as it can overrule by masking when needed.

$$h^t_{TMoTE}(c) = m_{\Delta sec} \odot (e_{\Delta turn} \oplus h^t(c)), \tag{6}$$

Time and Turn Embedding (TaTE) Our third variant assumes wall-clock second and turn offset have equal importance by removing the masking sigmoid of *Time Mask* in Equation (1) and sum with *Turn Embedding*.

$$h^t_{TaTE}(c) = e_{\Delta sec} \oplus e_{\Delta turn} \oplus h^t(c), \tag{7}$$

4 Results

In this section, we first describe our experimental setup in Section 4.1, present our main results in Section 4.2, followed by our analyses in Section 4.3.

4.1 Experimental Setup

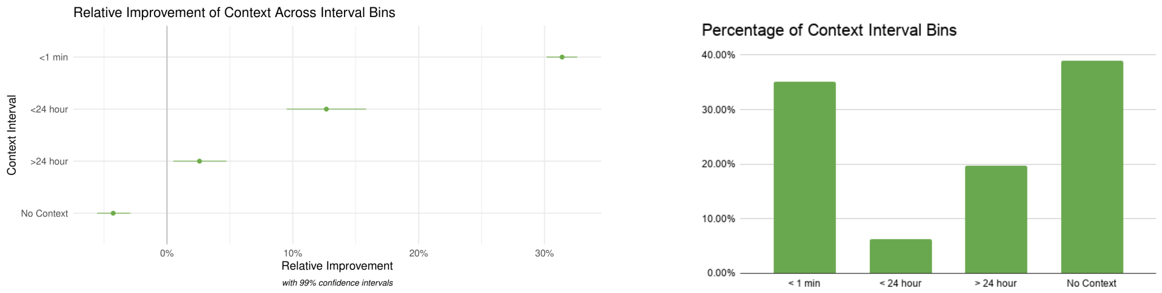
Dataset We use an internal SLU dataset that is privatized so that users are not identifiable. Our training, validation and test set contains on the order of several million, several hundred thousand, and one million utterances, respectively. For each utterance, we collect the previous 9 turns within a few days as context. Our dataset has a total of 24 domains that includes common voice assistant use cases (Liu et al., 2019).

Metric For evaluation, we report Accuracy Relative Error Reduction Percentage (ARER %). ARER % is computed with the following equation.

$$ARER_{ctx} = \frac{(1 - ACC_{utt}) - (1 - ACC_{ctx})}{1 - ACC_{utt}}, \tag{8}$$

Here ACC_{utt} is the accuracy of an utterance-only baseline that masks context information, and ACC_{ctx} is the accuracy of a contextual model.

Implementation Details We set both FastText and Elmo embedding dimensions to 300 and hidden dimension to 256 for all neural network layers, hypothesized domain and intent, time and turn embeddings. We used a bi-directional LSTM for turn encoder, uni-directional LSTM for sequence encoder and set both to 2 layers. For Transformer,



(a) (b)

Figure 2: (a) Left figure plots the ARER % (↑) with confidence intervals of our best model on different time interval bins. (b) Right figure depicts the percentage of each bin within our dataset.

we used 1 layer with 4 heads. Dropout rate is set to 0.2 for all fully-connected layers, and we used Adam (Kingma and Ba, 2015) as optimizer with learning rate set to 0.001. For utterances that do not have context, we use a special <PAD> token to pad the turn features. For consistency, we report results averaging 3 random seeds. We use the MXNet (Chen et al., 2015) framework to develop our models.

4.2 Main Results

In Table 1, we report performance of temporal representations with sequence encoders (1) Max-pooling, (2) LSTM, and (3) Transformer, computed with respect to an utterance-only baseline. For all sequence encoders, temporal representations combining both wall-clock second difference and turn order offset achieved best results. Specifically, *Time and Turn Embedding* works best for Max-pooling, and *Turn Embedding over Time Mask* works best for LSTM and Transformer. Transformer achieved the best results of 13.04%, improving 0.35% over using wall-clock and 2.26% using turn offset. Similar trends are observed with LSTM and Max-pooling, with both information outperforming using only one. In general, having *Time Mask* performs better than *Turn Embedding*, suggesting that wall-clock is more important than turn offset in CDC. Also, despite being a natural time series encoder, temporal representations further improve LSTM performance by up to an additional 1.49%.

4.3 Analysis

In this section, we conduct analyses to better understand the role of context in CDC.

Utt	Hyp-Domain	Hyp-Intent	ARER % (↑)
✓	✓	✓	13.04
✗	✓	✓	12.70
✓	✗	✓	10.20
✓	✓	✗	12.70
✓	✗	✗	5.37
✗	✓	✗	11.27
✗	✗	✓	10.73
✗	✗	✗	0.00

Table 2: Analysis on turn features used in Context Encoder. ✓ indicates the feature is used. ✗ indicates the feature is masked.

Recent & Distant Context To understand whether distant context actually improves SLU, we use the second difference of the first previous turn $d^1_{\Delta sec}$ to indicate absolute closeness and divide the test set into 3 non-overlapping interval bins: (1) $<$ *1 min*, (2) $<$ *24 hr*, (3) $>$ *24 hour*, where (1) represents recent context and (2), (3) are the more distant context. We also include a fourth bin (4) *No Context* for utterances that do not have context. Figure 2 depicts performance of our best model from Section 4.2 on each bin. While improvements are largest for (1), there are still statistically significant improvements for the more distant (2) and (3), suggesting that distant context is indeed helpful, albeit decreases with distance and at a smaller scale. Interestingly, our best model performed worse on (4), suggesting that models trained with context exhibit certain biases when evaluating without context.

Amount of Context Next, we analyze the number of previous turns needed for CDC. We trained and evaluated our best model from Section 4.2

Previous Turn		Current Turn		
Utterance	buy stuff	Utterance	t.v.	
Hyp-Domain	*Shopping*	Baseline	*SmartHome*	✗
Seconds	6.0	Best Model	*Shopping*	✓
Utterance	play <entity1>	Utterance	<entity1> by <entity2>	
Hyp-Domain	*Song*	Baseline	*AudioBooks*	✗
Seconds	54.0	Best Model	*Song*	✓
Utterance	please read audio collection	Utterance	start <entity>	
Hyp-Domain	*AudioBooks*	Baseline	*DeviceControl*	✗
Seconds	6235.0 (1 hr, 43 mins)	Best Model	*AudioBooks*	✓
Utterance	turn on <entity>	Utterance	turn off <entity>	
Hyp-Domain	*SmartHome*	Baseline	*DeviceControl*	✗
Seconds	212421.0 (2 days, 11hrs)	Best Model	*SmartHome*	✓

Table 3: Examples showing predictions of an utterance-only baseline and our best model from Section 4.2 with context from the first previous turn. Our best model is able to make correct predictions by utilizing context from recent and distant time ranges when the current turn utterance is ambiguous. We anonymize entities and modify certain utterances for user privacy. Hypothesized domain-specific intents and additional previous turns are not included for clarity.

using 1 and 5 previous turns, which resulted in ARER% of 10.00%, and 12.86%, respectively. Compared to 13.04% of using 9 previous turns, this suggests that while more than 1 previous turn is needed for performance, using 5 turns is comparable as using 9 turns and can potentially save caching costs.

Where Does Context Improve SLU Most CSLU works are motivated by rephrases and reference resolution (Chen et al., 2016; Rastogi et al., 2019). Noticing that in both phenomena users follow up their requests within the same domain, we split our test set based on whether the previous turn's hypothesized domain (PTHD) is same as or different from the target domain. Our model largely improved ARER % by 22.82% on the *PTHD Same* set, and has comparable performance of -0.03% on the *PTHD Different* set. This suggests that our model learns to carryover previous domain prediction when the current utterance is ambiguous and not over rely on them. We also include several examples with recent and distant context in Table 3 that exhibits this behavior.

Types of Context Information Last, we conducted an ablation study of turn features used in the context encoder. We mask 1 or retain 1 of the 3 features and show results in Table 2. The most effective feature we observed is the previously hypothesized domain, as masking domain yielded the worst results, and keeping domain yielded the best results. Since domain is a crude label, we hypothesize that previous domain predictions are sufficient for CDC, and utterance text will be more useful for more fine-grained tasks such as intent classification or slot labeling.

The upside of this analysis comes from deployment costs. Since pre-trained Elmo embeddings are computation heavy and may require GPU machines, using only hypothesized domain as turn features can largely lower the costs as we can inference using CPUs while sacrificing little accuracy.

5 Conclusions

We presented a novel large-scale industrial CDC setup and show that distant context also improves SLU. Our proposed temporal representations combining both wall-clock and turn order information achieved best results for various encoder architectures in a hierarchical model and outperforms using only one of the two. Our empirical analyses revealed how previous turn helps disambiguation and showed opportunities on reducing deployment costs.

For future work, we plan to explore more turn features such as responses, speaker and device information. We also plan to apply temporal representations on other tasks, such as intent classification, slot labeling, and dialogue response generation.

6 Ethics Statement

Our dataset is annotated by in-house workers who are compensated with above minimum wages. Annotations were acquired for individual utterances and not for aggregated sets of utterances. To protect user privacy, user requests that leak personally-identifiable information (e.g., *address*, *credit card number*) were removed during dataset collection. As our model is a classification based which output is within a finite label set, incorrect predictions will not cause harm to the user besides an unsatisfactory experience.

Acknowledgements

We would like to thank the anonymous reviewer's for their useful suggestions, Yuval Merhav and Bo Xiao for their helpful feedback on the early versions of this paper, and Alan Packer for proofreading during internal review process.

References

Ankur Bapna, G. Tür, Dilek Z. Hakkani-Tür, and Larry Heck. 2017. Sequential dialogue context modeling for spoken language understanding. In *SIGDIAL Conference*.

A. Bhargava, A. Çelikyilmaz, Dilek Z. Hakkani-Tür, and R. Sarikaya. 2013. Easy contextual intent prediction and slot detection. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8337–8341.

P. Bojanowski, E. Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. 2015. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274*.

Yun-Nung Chen, Dilek Hakkani-Tur, Gokhan Tur, Jianfeng Gao, and Li Deng. 2016. End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *Proceedings of Interspeech*.

Rylan T. Conway and Lambert Mathias. 2019. Time masking: Leveraging temporal information in spoken dialogue systems. In *SIGdial*.

Arshit Gupta, Peng Zhang, Garima Lalwani, and Mona Diab. 2019. CASA-NLU: Context-aware self-attentive natural language understanding for task-oriented chatbots. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1285–1290, Hong Kong, China. Association for Computational Linguistics.

Raghav Gupta, Abhinav Rastogi, and Dilek Hakkani-Tür. 2018. An efficient approach to encoding context for spoken language understanding. *Proc. Interspeech 2018*, pages 3469–3473.

Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, 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*, pages 35–44, 1st virtual meeting. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Jason Williams. 2014. The second dialog state tracking challenge. In *15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 263. Citeseer.

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

Jonggu Kim and Jong-Hyeok Lee. 2019. Decay-function-free time-aware attention to context and speaker indicator for spoken language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3718–3726, Minneapolis, Minnesota. Association for Computational Linguistics.

Young-Bum Kim, Dongchan Kim, Joo-Kyung Kim, and Ruhi Sarikaya. 2018. A scalable neural shortlisting-reranking approach for large-scale domain classification in natural language understanding. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 16–24.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Martin Kleppmann. 2017. *Designing Data-Intensive Applications*. O'Reilly, Beijing.

Han Li, Jihwan Lee, Sidharth Mudgal, Ruhi Sarikaya, and Young-Bum Kim. 2019. Continuous learning for large-scale personalized domain classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3784–3794, Minneapolis, Minnesota. Association for Computational Linguistics.

Chunxi Liu, Puyang Xu, and Ruhi Sarikaya. 2015. Deep contextual language understanding in spoken dialogue systems. In *INTERSPEECH*.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. In *10th International Workshop on Spoken Dialogue Systems Technology 2019*.

Chetan Naik, Arpit Gupta, Hancheng Ge, Lambert Mathias, and Ruhi Sarikaya. 2018. Contextual slot carryover for disparate schemas. In *INTERSPEECH*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Pushpendre Rastogi, Arpit Gupta, Tongfei Chen, and Lambert Mathias. 2019. Scaling multi-domain dialogue state tracking via query reformulation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Yangyang Shi, Kaisheng Yao, Hu Chen, Yi-Cheng Pan, Mei-Yuh Hwang, and Baolin Peng. 2015. Contextual spoken language understanding using recurrent neural networks. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5271–5275.

Chengwei Su, Rahul Gupta, Shankar Ananthakrishnan, and Spyridon Matsoukas. 2018a. A reranker scheme for integrating large scale nlu models. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 670–676.

Shang-Yu Su, Pei-Chieh Yuan, and Yun-Nung Chen. 2018b. How time matters: Learning time-decay attention for contextual spoken language understanding in dialogues. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2133–2142, New Orleans, Louisiana. Association for Computational Linguistics.

Gokhan Tur. 2011. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Jason D. Williams, Antoine Raux, and Matthew Henderson. 2016. The dialog state tracking challenge series: A review. *Dialogue Discourse*, 7:4–33.

Puyang Xu and Ruhi Sarikaya. 2014. Contextual domain classification in spoken language understanding systems using recurrent neural network. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 136–140.