

Dialogue state tracking accuracy improvement by distinguishing slot-value pairs and dialogue behaviour

Khaldoon H. Alhussayni¹, Alexander Zamyatin², S. Eman Alshamery³

¹Computer Center, Babylon University, Babylon, Iraq.

²ComputerScience Department, Tomsk State University, Tomsk, Russia

³Software Department, Babylon University, Babylon, Iraq

Article Info

Article history:

Received Apr 23, 2020

Revised Jun 15, 2020

Accepted Jul 8, 2020

Keywords:

Dialogue state tracking

Encoder-decoder

Natural language processing

Task-oriented dialogue system

ABSTRACT

Dialog state tracking (DST) plays a critical role in cycle life of a task-oriented dialogue system. DST represents the goals of the consumer at each step by dialogue and describes such objectives as a conceptual structure comprising slot-value pairs and dialogue actions that specifically improve the performance and effectiveness of dialogue systems. DST faces several challenges: diversity of linguistics, dynamic social context and the dissemination of the state of dialogue over candidate values both in slot values and in dialogue acts determined in ontology. In many turns during the dialogue, users indirectly refer to the previous utterances, and that produce a challenge to distinguishing and use of related dialogue history. Recent methods used and popular for that are ineffective. In this paper, we propose a dialogue historical context self-Attention framework for DST that recognizes relevant historical context by including previous user utterance beside current user utterances and previous system actions where specific slot-value pairs variations and uses that together with weighted system utterance to outperform existing models by recognizing the related context and the relevance of a system utterance. For the evaluation of the proposed model the WoZ dataset was used. The implementation was attempted with the prior user utterance as a dialogue encoder and second by the additional score combined with all the candidate slot-value pairs in the context of previous user utterances and current utterances. The proposed model obtained 0.8 per cent better results than all state-of-the-art methods in the combined precision of the target, but this is not the turnaround challenge for the submission.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Khaldoon H. Alhussayni

Computer Center

Babylon University, Babylon, Iraq

Email: alhussayni83@gmail.com

1. INTRODUCTION

Dialog state tracking (DST) is a crucial part of task-oriented dialogue systems that track user objectives at each turn based on dialogue history. Task-oriented dialogue systems communicate with natural-language users by talking to a voice interface or writing text to perform tasks they have in mind. Modern DST methods rely on models for deep learning.

Proposals that are more recent have been made for several neural-based DST structures. In addition to handcrafts features, where a representation based on pre-trained word embeddings is computed to agree on such vectors, the neural faith tracker (NBT) model proposed to use deep learning (DNN) and convolutionary network (CNN) to calculate these representation vectors in order to learn feature appropriate

for each state by [1]. Nevertheless, multi-valued slots do not take into account all values, although this work may predict multiple potential values [2]. Zhong et al. (2018) proposed the GLAD model with self-attention-based recurrent networks and simulated representations for each utterance and previous system behaviour, calculating a similarity to each slot-value [3]. This has improved GLAD architecture with the elimination of slot-dependent recurrent utterance networks and device intervention encoder and the use of an application vector globally conditioned encoder (GCE). In the last two approaches, however, distributors in the production system were still ineffective due to their failure to recognise and take the applicable context on board, while the work could identify the associated context. The purpose of DST is to predict the set of objectives that correspond to ontological products in user statements that are represented as slot-value pairs. This will become a complicated task when the lexical discrepancy is challenged dynamics of the context production, the distribution of the state of the dialogue, total pairs of candidates and dialogue actions described in conceptualization. We present here an improvement model used by [3], using the prior values in each spin, which form the state. To this end, the current and previous user statements have been merged and variations between them and ontology have been determined.

2. RELATED WORK

The methods of dialogue state monitoring can be divided into a rules-based approach, statistics and a deep learning system. The use rule-based heuristics and calculate the confidence scores of the N-best candidates produced to determine the correct dialogue states from the performance of the natural language comprehension module [4-7]. The purpose is monitor the details needed to track the status of the dialogue. Nevertheless, these principles are not collected automatically from actual information on a dialogue to involve careful tuning and responsive design attempts. Due to these techniques, the definition of dialogue states often leads to inaccuracy.

Statistical methods for dialogue between the State Tracking [8-10] have been used to provide alternatives to crafted rules. Statistical approaches such as logistic regression and the Bayesian network for the accomplishment of high-performance monitoring and the implementation of a confidence evaluation of user information. These reports, however, have a common problem that every conversation, rather expensive computation, should be included in every. Recently, the use of deep learning techniques for dialogue tracking the state [1, 11-19]. In addition, others are able to understand specific user and system utterances and prior system actions to predict turnaround.

First used for state tracking dialogues was the neural network [20]. Our research is important because, in a pipeline approach, the first attempt is to use a neural network to dialogue state tracking to obtain appropriate data from the user utterances. In the absence of the necessary user interpretation dialogue framework, these schemes can be used to accumulate errors separately within the language module.

State Tracking Dialog results demonstrate the utility of learning to collectively interpret speech and monitor dialogue [14, 18, 19]. These solutions are extracted from the N-Best list developed by the automated speech recognition programme. By avoiding error build-up from the comprehension dimension of the original language. Such frameworks include the use of common tagings to override unique slot forms and values, as well as handcrafted procedural dictionaries. Nevertheless, such models rely on handcrafted features and complicated domain-specific lexicons, which are difficult to scale for each form of slot and therefore difficult to apply to new domains.

Recent state-of-the-art projections for DST predicted the condition of every transition by observing uniform consumer and machine utterances representations. Nevertheless, the efficiency of these schemes is low in the unusual and unfamiliar slot values that have recently been addressed by local slot encoders [3] and the pointer network [21].

The Global-Local Self-Attention Encoder model as [2] suggested recurrent self-attention networks with a computed representation by comparing the similarities of every slot value to each user utterance and prior device behaviour. [3] improved global-local self-care encoder structure by eliminating slot-based recurrent voice networks and system encoders and using a global-conditioned embedded slot style encoder. Nevertheless, because of their lack of activity in understanding and incorporating the relevant context, these approaches were not successful in the production system, while this research may understand the connecting sense.

3. PROPOSED MODEL

We present the proposed model in this section. Section 3.1 first explains the recently proposed architecture of GCE [3], followed by the proposed encoder in Section 3.2 then the model scour in Section 3.3.

3.1. Globally conditioned encoder (GCE)

Each encoder input is represented as a representation of the vector(C). In the GCE model there is the two-way LSTM system [22] and all slots which are shared to achieve a sequence of hidden states by encrypting the input sequence, shaded by a self attention layer [23]. The GCE approach takes into account user expression and the previous system action for model learning. However, in our work, we used a GCE approach to learn distribution of Slot Value Pairs and context, not only current user utterances and previous system actions but also previous user utterance.

3.2. Encoder model

We followed the proposed architecture for calculation of each slot value pair's encoder, user utterance and previous system actions at GCE. However, we use an additional encoder to extract the history and the context for previous user utterances. The encoder model is used to encode the previous user utterances (H_{per} , C_{per}), the current user utterances (H_{cur} , C_{cur}), the previous system actions (H_e , C_a) and slot value (H_v , C_v) for each system act. As shown in Figure 1, the slot-embedding vector for the k^{th} slot is used for context extraction.

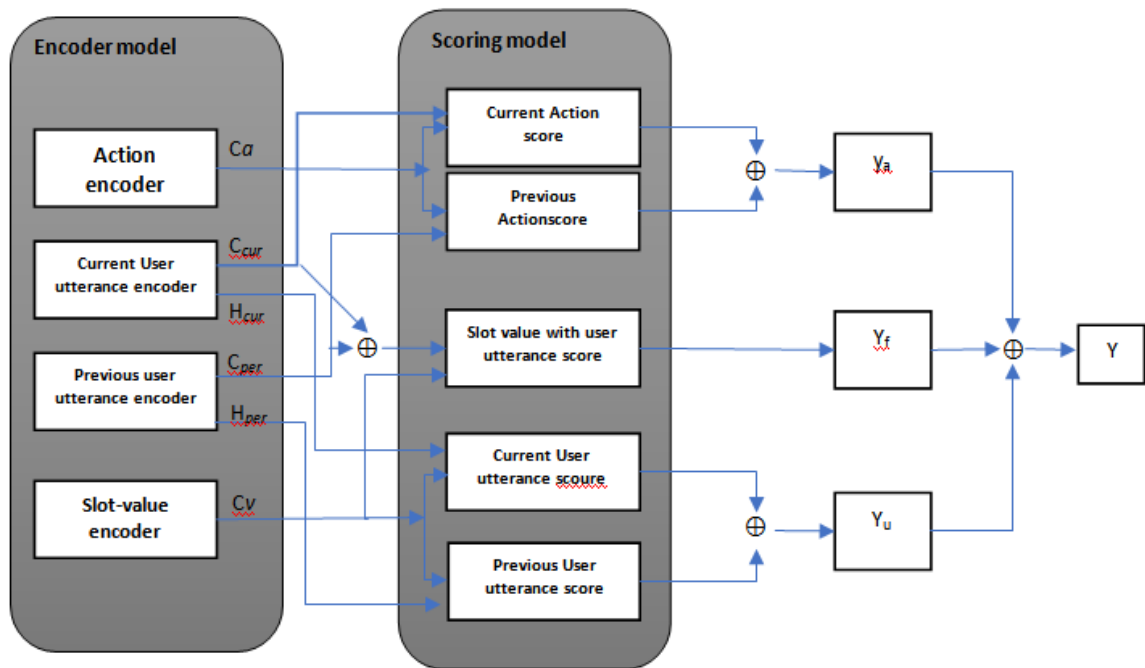


Figure 1. Dialogue historical context self-attention model for dialogue state tracker

To calculate representation H^k for each slot k^{th} as shown in (1), we concatenated the slot embedding s_k with input sequence X , i.e., current user utterance, previous user utterance, or previous system actions, as input to the encoder, where concatenation is denoted as $f(X, s_k)$.

$$H^k = \text{BiLSTM}(f(X, s_k)) \in \mathbb{R}^{n \times d_r} \tag{1}$$

Where d_r is the dimension of the LSTM state. Then we calculate the attention score a_i^k of the slot for each token hidden representation H_i^k as shown in (2), by concatenating them to the slot embedding s_k and transitory to a linear layer, then applying a softmax p^k in (3) to normalize the distribution. In (4) compute similarly of the context c^k .

$$a_i^k = Wf(H_i^k, s_k) + b \in \mathbb{R} \tag{2}$$

$$p^k = \text{softmax}(a^k) \in \mathbb{R}^n \tag{3}$$

$$c^k = \sum_i p_i^k H_i^k \in \mathbb{R}^{d_r} \tag{4}$$

Each of the four encoders in the encoder model, as shown in Figure 1, can be represented as follows:

- U, s_k as inputs and H_u^k, c_u^k as outputs, where U denotes word embeddings of the user utterance.
- P, s_k as inputs and H_p^k, p as outputs, where P denotes word embeddings of the previous user utterance.
- A_j, s_k as inputs and H_{aj}^k, C_{aj}^k as outputs, where A_j th is the previous system action.
- V, s_k as inputs and H_v^k, c_v^k as outputs, where V denotes current slot-value pair.

3.3. Scoring model

In GLAD [2], we adopted the suggested method to calculate the performance of each slot-value pair in current and previous consumer utterances and previous system steps. They however use the additional point to improve the meaning and delivery over the history of dialogue.

The scores model is used to measure slot k for its slot values in order to evaluate the slot value that the users name. This was also done using five examples. The first score $a_{cur_i}^k$ as shown in (5) is the current user utterance H_{cur} , taking into account the slot-value pair being considered c_v and using the resulting attention context q_{cur} as shown in (6) to score the slot-value pair.

$$a_{cur_i}^k = \text{softmax}((H_{cur_i}^k)^T c_v^k) \in \mathbb{R}^m \quad (5)$$

$$q_{cur}^k = \sum_i a_{cur_i}^k H_{cur_i}^k \in \mathbb{R}^{d_r} \quad (6)$$

$$y_{cur}^k = W q_{cur}^k + b \in \mathbb{R} \quad (7)$$

where m indicates a number of words in the input sequence. The score y_{cur}^k as shown in (7) denotes the predicted values of the user utterance.

The second score $a_{pre_i}^k$ as shown in (8) is similar to the first score, but uses previous user utterance H_{pre} instead of the current user utterance, taking into account the slot-value pair being considered c_v and using the resulting attention context q_{pre}^k as shown in (9) to score the slot-value pair. The score y_{pre}^k as shown in (10) denotes the predicted values of the previous utterance.

$$a_{pre_i}^k = \text{softmax}((H_{pre_i}^k)^T c_v^k) \in \mathbb{R}^m \quad (8)$$

$$q_{pre}^k = \sum_i a_{pre_i}^k H_{pre_i}^k \in \mathbb{R}^{d_r} \quad (9)$$

$$y_{pre}^k = W q_{pre}^k + b \in \mathbb{R} \quad (10)$$

Then the predicted values of both current and previous user utterances are added as shown in the following as shown in (11):

$$y_u^k = y_{pre}^k + y_{cur}^k \quad (11)$$

Similarly, this is used to determine the mentioned previous system actions in the current or previous user utterance separately to reach sufficient information about user utterance when this is not informative. The third score $a_{acur_i}^k$ as shown in (12), the context of current user utterance C_{cur} over the previous action representations $C_a = [C_{a1} \cdot \cdot \cdot C_{al}]$. Here, l is the number of previous system actions. Then we use the similarity between the attention context q_{acur} as shown in (13) and the slot-value pair c_v to score the slot-value pair. The score y_{acur}^k as shown in (14) denotes the predicted values of the previous system actions in the current user utterance separately.

$$a_{acur_i}^k = \text{softmax}((C_{cur_i}^k)^T c_v^k) \in \mathbb{R}^{l+1} \quad (12)$$

$$q_{acur}^k = \sum_i a_{acur_i}^k C_{cur_i}^k \in \mathbb{R}^{d_r} \quad (13)$$

$$y_{acur}^k = W q_{acur}^k + b \in \mathbb{R} \quad (14)$$

The fourth source $a_{apre_i}^k$ as shown in (15), similar to the third score, but uses the context of previous user utterance C_{pre} inserted into the context of a current user utterance.

$$a_{aprei}^k = softmax((C_{pre_i}^k)^T c_v^k) \in \mathbb{R}^{l+1} \tag{15}$$

$$q_{apre}^k = \sum_i a_{aprei}^k C_{pre_i}^k \in \mathbb{R}^{d_r} \tag{16}$$

$$y_{apre}^k = W q_{apre}^k + b \in \mathbb{R} \tag{17}$$

Then the predicted values of both the context of the current and previous user utterances are added:

$$y_a^k = y_{apre}^k + y_{acur}^k \tag{18}$$

In the last score, we followed the proposed approach in [24] to determine the relevance of the slot-value pair in the current turn. We concatenate the context of both current user utterance C_{cur}^k and previous user utterance C_{pre}^k and use a sigmoid activation of the linear layer to compute the score.

$$fc = W_{fc}(C_{cur}^k \oplus C_{pre}^k) + b_{fc} \tag{19}$$

$$\alpha = \sigma(W_\alpha tanh(fc) + b_\alpha) \tag{20}$$

Then we compute context summaries l_u of attention from C_v over H_{cur} and l_s of attention from C_v over H_{per} .

$$l_u^k = Q(H_{cur}^k, c_v^k) \tag{21}$$

$$l_s^k = Q(H_{per}^k, c_v^k) \tag{22}$$

To compute the additional score y_f that establishes the probability of the candidate slot-value based on both the current and previous user utterances and the previous system utterance, we use:

$$y_f^k = \alpha l_u^k + (1 - \alpha) l_s^k \tag{23}$$

Finally, we add the weight of all scores of slot k, i.e., y_u^k , y_a^k , and y_f^k , which are normalized by the sigmoid function:

$$Y = \sigma(y_u^k + w y_a^k + y_f^k) \in \mathbb{R} \tag{24}$$

where w is a learned parameter.

4. EXPERIMENT

The Wizard of Oz (WoZ) is a single domain for a monitoring platform dialogue booking (Wen et al. 2016). The emotion dataset consists of 600 instruction dialogues, 200 appraisal dialogues and 400 assessments. -dialogue has a total of eight turns, with a machine transcript, a user utterance transcript, a turn mark and dialogue status for each switch. The ontology consists of three separate slot forms: food with 72 values, 7 values, 4 values, and 7 different kind of slots, including telephone number and address. The ontology consists of three specific slot styles. In order to assess the metric of mutual target monitoring precision, we observed the aggregation of turns targets [2].

4.1. Implementation details

The pre-trained embedding of GLoVe word [25] is used, which is paired with n-gram embedding characteristics, and is retained throughout the training. The units for BI-LSTMs are described in 200 hidden dimensions and the initial learning rate is trained using an ADAM optimizer [26]. For the embedding layer, we defined the drop-out rate. The average number of times is 50 to train the 50 batch models.

4.2. Results and analysis

The previous DST consumer declaration is used to gain the necessary information to increase the likelihood of allocation among all applicant slot-value pairs and action behaviour. Using a model to practise and check the whole cycle is streamlined and pace benefits. The encoder side of the model is determined for both activities only once. In a segment scoring model, we determine the outcomes independently from the previous user statement and current user statement over all applicant slot pairs and intervention steps to

increase coverage and distribution over dialogue history, as shown through Figure 1. In fact, both the current and previous consumer utterances are fused in the sense of each applicant slot meaning pair, as shown in (the additional score of) Figure 1. Compared with all state-of-the-art strategies in the shared goal phase, the proposed model obtained superior results, which did not include the request turning challenge.

Table 1 provides a summary of the results of our new models with those of previous state-of-the-art projects. The NBT [1], which utilises common marks instead of slots and meanings in the utterance, uses CNN with pre-training term embeddings for representational learning instead of delexicalised [15]. In the design of GLAD [2], the utterance, previous system operation, and all slot values are segregated in encoders. For all encoders, a common design is used. The writers used two ratings, each measuring the relation between the slot meaning pairs and the consumer utterance or previous system operation. The researchers followed a similar method to GLAD in the GCE model. The Encoder has however been updated to increase latency and speed of inference by removing inadequate recurrent layers and self-attention layers.

The WoZ dataset has been used to test our proposed model and introduce two models. In the dialogue of identical design [3], previous user definitions were used as additional encoders with the same accuracies as GCE layout. The second was by integrating the meaning of previous user utterances and current user utterances with all candidate slot value pairs. Increasing existing user utterance with the correct user information and the previous system utterance also improves the standard target accuracy by 0.8%, as shown by the findings in Table 1 and request accuracy.

Table 1. Comparison of our model to previously published WoZ restaurant reservation dataset

Model	Joint Goal	Turn Request
Neural Belief Tracker (NBT)—DNN	84.4%	91.2%
Neural Belief Tracker (NBT)—CNN	84.2%	91.6%
Global-Locally Self-Attentive (GLDA)	88.1%	97.1%
Globally Conditioned Encoder (GCE)	88.5%	97.38%
GCE + previous utterance	88.5%	96.98%
GCE + previous utterance + additional score	89.3%	96.98%

5. CONCLUSION

DST's objective is to define two goals: the current state of each rotation of dialogue slot value pairs and the user dialogue act to sum up all the user's objectives. In this paper, we propose a different framework for DST as a dialogue on the historical context self-Attention to distinguish and use the related dialogue history by including the previous user's utterance in two separate ways and to fuse in order to provide the model with the necessary information. Relative to all state-of-the-art methods, the proposed model obtained outcomes of 0.8% in the shared aim precision with the WoZ data set used to test our proposed model. Our model uses a smaller number of learning parameters applied to interpret current and previous user statements using the appropriate context. They also have a high disparity in the precision of the joint aim because the joint objective is decided by piling turn targets and errors when estimating a turn destination are comparatively smaller.

REFERENCES

- [1] N. Mrkšić, D. Ó. Séaghdha, T.-H. Wen, *et al.*, "Neural belief tracker: Data-driven dialogue state tracking," *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1777-1788, 2016.
- [2] V. Zhong, C. Xiong, and R. Socher, "Global-Locally Self-Attentive Dialogue State Tracker," *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 1458-1467, 2018.
- [3] E. Nouri and E. Hosseini-Asl, "Toward scalable neural dialogue state tracking model," *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), 2nd Conversational AI workshop Montréal, Canada*, 2018.
- [4] R. Higashinaka, M. Nakano, and K. Aikawa, "Corpus-based discourse understanding in spoken dialogue systems," *Proc. 41st Annu. Meet. Assoc. Comput. Linguist.*, pp. 240-247, 2003.
- [5] J. Williams, A. Raux, D. Ramachandran, A. Black, "The Dialog State Tracking Challenge with Bayesian Approach," *Proc. of 14th Annu. Meet. Spec. Interes. Gr. Discourse Dialogue*, no. August, pp. 404-413, 2017.
- [6] K. Sun, L. Chen, S. Zhu, K. Yu. "A generalized rule based tracker for dialogue state tracking," *Proc. Spok. Lang. Technol. Work.*, pp. 330-335, 2014.
- [7] S. L. and D. R. Traum, "Information state and dialogue management in the trindi dialogue move engine toolkit," vol. 6, no. 3, pp. 323-340, 2000.
- [8] D. Bohus and A. Rudnicky, "K hypotheses + other" belief updating model," *Proc. AAI Work. Stat. Empir. Approaches to Spok. Dialogue Syst.*, pp. 13-18, 2006.
- [9] Y. Ma, A. Raux, D. Ramachandran, and R. Gupta, "Landmark-based location belief tracking in a spoken dialog system," *SIGDIAL 2012 - 13th Annu. Meet. Spec. Interes. Gr. Discourse Dialogue, Proc. Conf.*, pp. 169-178, 2012.

- [10] B. Thomson and S. Young, "Bayesian update of dialogue state: A POMDP framework for spoken dialogue systems," *Comput. Speech Lang.*, vol. 24, no. 4, pp. 562-588, 2010.
- [11] Abhinav Rastogi, Dilek Hakkani-Tur, L. Heck, "Scalable multi-domain dialogue state tracking," *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 561-568, 2017.
- [12] I. Casanueva *et al.*, "A benchmarking environment for reinforcement learning based task oriented dialogue management," *CoRR*, vol. abs/1711.1, 2017.
- [13] M. Henderson, B. Thomson, and S. Young, "Deep neural network approach for the dialog state tracking challenge," *SIGDIAL 2013 - 14th Annu. Meet. Spec. Interes. Gr. Discourse Dialogue, Proc. Conf.*, pp. 467-471, 2013.
- [14] M. Henderson, *et al.*, "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)*, 2014, pp. 292-299.
- [15] M. Henderson, B. Thomson, and S. Young, "Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation," *2014 IEEE Spoken Language Technology Workshop*, pp. 360-365, 2014.
- [16] N. Mrkšić, *et al.*, "Multi-domain dialog state tracking using recurrent neural networks," *ACL-IJCNLP 2015 - 53rd Annu. Meet. Assoc. Comput. Linguist. 7th Int. Jt. Conf. Nat. Lang. Process. Asian Fed. Nat. Lang. Process. Proc. Conf.*, vol. 2, pp. 794-799, 2015.
- [17] J. Perez and F. Liu, "Dialog state tracking, a machine reading approach using Memory Network," *CoRR*, vol. abs/1606.04052, 2016.
- [18] T. Wen, *et al.*, "A network-based end-to-end trainable task-oriented dialogue system," *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, vol. 1, pp. 438-449, 2017.
- [19] L. Zilka and F. Jurcicek, "Incremental LSTM-based dialog state tracker," *2015 IEEE Work. Autom. Speech Recognit. Understanding, ASRU 2015 - Proc.*, vol. 1, no. L, pp. 757-762, 2016.
- [20] M. Henderson, B. Thomson, and J. Williams, "Dialog State Tracking Challenge 2 & 3," *Dialogues with Soc. Robot.*, no. September, pp. 1-22, 2013.
- [21] P. Xu and Q. Hu, "An end-to-end approach for handling unknown slot values in dialogue state tracking," *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018.
- [22] H. Sepp and S. Jurgen, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735-1780, 1997.
- [23] Z. Lin *et al.*, "A structured self-attentive sentence embedding," *5th International Conference on Learning Representations (ICLR 2017)*, pp. 1-15, 2017.
- [24] S. Sharma, P. K. Choubey, and R. Huang, "Improving Dialogue State Tracking by Discerning the Relevant Context," *Proceedings of NAACL-HLT 2019*, pp. 576-581, 2019.
- [25] C. D. M. Jeffrey Pennington, Richard Socher, "GloVe: Global Vectors for Word Representation," *Proc. 2014 Conference Empir. Methods Nat. Lang. Process.*, pp. 1532-1543, 2014.
- [26] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *International Conference on Learning Representations*, pp. 1-15, 2014.

BIOGRAPHIES OF AUTHORS



Khaldoon Hasan Alhussayni received the BSc degrees in Computer Science from the University of Babylon, Iraq, in 2008. After completing his BSc, he worked as a programmer at the Department of Computer Center, the University of Babylon. In 2009, received the MSc degrees in Computer Science and Computer Engineering from the Tula State University, Russian, in 2014. Since Nov 2016 he entered the Tomsk state university (TSU), Department of computer science theory as a Ph.D. student in specialty Mathematical and software of computers, complexes and computer networks. His main research interests are machine learning, neural networks, deep learning, dialogue systems and natural language processing.



Zamyatin Alexander Vladimirovich, D. Sc., associate professor, Head of Department of computer science theory in the Tomsk State University (TSU). In December 2002 he entered the full-time postgraduate study of Tomsk Polytechnic University (TPU) in specialty 05.13.11-Mathematical and software of computers, complexes and computer networks, which he graduated ahead of schedule due to with the defense of his thesis. As an associate Professor of the Department of computer science (part-time) continued to carry out educational and research activities. From January 2019 he was appointed acting Director of the Institute of applied mathematics and computer science. Zamyatin A.V. completed a study and was entitled the teacher of the high school IGIP International Engineering Educator "ING-PAED IGIP", No ENG-255, is actively involved in scientific research, participant research as the head (Russian foundation for basic research grants No 11-07-00027, 03-07-06024, 14-07-00127, GK No 14.515.11.0047 on the instructions of Ministry of education and science of the Russian Federation) and responsible officer (grants RFBR №№ 00-07-90124, 03-07-90124, 06-05-78056). Executor for the TSU for the mega-project "Creating of local high-tech software and instrumental complex for the implementation of process control systems based on free software" carried out in collaboration with "EleSy" company and Tomsk state university. According to the results of scientific activity Zamyatin published more than 70 scientific papers, including 2 monographs. Zamyatin A.V. is a developer and head of the master's educational program "Data science and bioinformatics".



Eman Al-Shamery received the BSc and MSc degrees in Computer Science from the University of Babylon, Iraq, in 1998 and 2001, respectively. After completing her MSc, she worked as an assistant lecturer at the Department of Computer Science, the University of Babylon. In 2013, she received her PhD in Computer Science from the University of Babylon. Currently, she holds a professor position at Software Department, University of Babylon. Her current research interests include artificial intelligence, bioinformatics, machine learning, neural networks, deep learning and data mining.