Infusing Context and Knowledge Awareness in Multi-turn Dialog Understanding

Ting-Wei Wu  
Georgia Institute of Technology  
Electrical & Computer Engineering  
waynewu@gatech.edu

Biing-Hwang Juang  
Georgia Institute of Technology  
Electrical & Computer Engineering  
juang@ece.gatech.edu

Abstract

In multi-turn dialog understanding, semantic frames are constructed by detecting intents and slots within each user utterance. However, recent works lack the capability of modeling multi-turn dynamics within a dialog in natural language understanding (NLU), instead leaving them for updating dialog states only. Moreover, humans usually associate relevant background knowledge with the current dialog contexts to better illustrate slot semantics revealed from word connotations, where previous works have explored such possibility mostly in knowledge-grounded response generation. In this paper, we propose to amend the research gap by equipping a BERT-based NLU framework with knowledge and context awareness. We first encode dialog contexts with a unidirectional context-aware transformer encoder and select relevant inter-word knowledge with the current word and previous history based on a knowledge attention mechanism. Experimental results in two complicated multi-turn dialog datasets have demonstrated significant improvements of our proposed framework. Attention visualization also demonstrates how our modules leverage knowledge across the utterance.

1 Introduction

In conventional task oriented dialog systems, natural language understanding (NLU) modules aim to transform utterances into meaningful semantic representations for dialog management (Weld et al., 2021; Zhang et al., 2020). It mainly detects associated dialog acts or intents and extracts key slot information as so-called ‘semantic frames’ (Abbeduto, 1983), shown in Table 1. Humans usually associate relevant knowledge and previous contexts with current utterance’s entities to understand an utterance. Similarly, models’ prediction of overall intent semantics and slot values can benefit from act relations such as ‘Inform’ may follow ‘Request’ acts, and background knowledge which is usually represented as triples in knowledge graphs (Wang et al., 2021a).

However such intuition has not been emphasized when automating NLU tasks. In early attempts of NLU systems, utterances were isolated and analyzed separately for user intents and semantic slots (Raymond and Riccardi, 2007; Liu et al., 2017). Models that maximize the joint distribution likelihood were proposed to allow transitions between two tasks (Liu and Lane, 2016; Wang et al., 2018; Wu et al., 2021a; Li et al., 2018a). While driven by large pretrained corpora, these methods still fall short of employing complete dynamic interactions within dialogs, especially in multiple intent cases (Qin et al., 2019; Rashmi Gangadharaiah, 2019; Qin et al., 2020). Some works have then integrated dialog contexts for more robust NLU (Wang et al., 2019; Gupta et al., 2019; Su et al., 2021; Wu et al., 2021c). However, many of them could not capture dialog flows well with RNN encoders or explain how contexts should affect the slot filling task.

Publicly available models like BERT or XLNet
provide universal contextualized representations that could be adapted for learning task-oriented contexts. However, it may not give full play to its value when tagging some rare words like *Foxtrot* together with *Tango* as *Movie* in Table 1 that may appear in a domain-specific dataset. One can pretrain these models beforehand emphasizing such phrase relationship which nevertheless tends to be time-consuming and computationally expensive. Therefore, directly integrating external knowledge like a knowledge graph (KG) becomes a more tractable solution (Liu et al., 2019; Zhang et al., 2019b; Wu and Juang, 2022b).

However, there are mainly three challenges lying in the way of such integration: (1) **Heterogeneous information fusion**: the vector space of KG entities is inconsistent with that of the pre-trained models. (2) **Knowledge noise**: overwhelming knowledge for models may adversely cause redundant noises for more ambiguity. Many works in knowledge-grounded dialog generation has applied term-level denoising (Zheng et al., 2021) or filtering techniques (Wang et al., 2021b) to refine the adopted knowledge for better semantic considerations. (3) **Inter-token knowledge sharing**: Wang et al. (2019) predicts a slot for a given word along with its own associated knowledge. However, real sentences may contain phrases where knowledge between words should be shared to probably enrich the entire utterance semantics. To overcome these challenges and ground knowledge in contextual NLU, which is less explored in the research community, we propose a Context and Knowledge Awareness NLU Framework (CKA-NLU) to effectively incorporate relevant knowledge and dialog history in dialog understanding.

The key ingredients lie in how we can efficiently integrate relevant knowledge and previous history for understanding. We first introduce a context attention module to retrieve context-aware representations. Different from previous works of determining a given word’s slot based on its own knowledge, our objectives require models to aggregate both previous dialog contexts and all intra-sentence knowledge facts together to formulate context-attended knowledge vectors in the same space. Such vectors are a weighted combination of all knowledge facts based on the aggregated information until the current turn. We use attention masks and filtering to remove adversarial effects from redundant knowledge noises. Finally we adopt these context-attended vectors for NLU tasks with RNN decoders. Experiment results have shown superior performances of our methods that beat all competitive baselines.

Our contributions are as follows:

1. We propose a novel CKA-NLU framework that incorporates inter-word knowledge with inter-sentence contexts to fill the void of relevant knowledge exploration for important NLU tasks.
2. We demonstrate the benefits of adopting knowledge for token-level slot filling and dialog history for sentence-level intent detection.
3. Experimental and attention visualization results show that our model achieves superior performances over several competitive baselines and demonstrates how our model adopts the knowledge.

2 Problem Formulation

For each utterance \( x_n = \{w_1^n, w_2^n, \ldots, w_N^n\} \) in a task-oriented dialog \( X \) with \( N \) utterances, given the domain ontology of a dialog act set \( A \) and a slot set \( S \), we aim to find one or more acts \( \{a_i^n\} \) and a sequence of slot tags \( \{s_1^n, s_2^n, \ldots, s_T^n\} \) to construct a semantic frame. Namely, we hope to maximize the joint log likelihood of \( A \) and \( S \) in Eq 1 given a parametrized model \( \theta \), its context \( C_n = \{x_1, \ldots, x_{n-1}\} \) and associated knowledge \( K_n = \phi(K_G, x_n) \) for the current utterance \( x_n \). We deem \( K_G \) as an external large knowledge base with knowledge represented as triples (head \( h \), relation \( r \), tail \( t \)) and \( \phi(\cdot) \) helps to extract related knowledge pairs for \( x_n \) (§3.2.1). It will be critical to match correct knowledge based on current dialog history and the utterance for better dialog understanding.

\[
\mathcal{L}(A, S) \equiv \sum_n \log P(A_n, S_n | x_n, C_n, K_n; \theta)
\]

(1)

3 Methodology

3.1 Context Attention

Our overall framework is illustrated in Figure 1. To allow information flow across the dialog, we first encode the entire dialog with a token-level BERT (Devlin et al., 2019) encoder and a turn-level context-aware transformer encoder. Instead of concatenating all sentences which may cause an extreme sequence length, we first generate the token-level representations \( H = \{h_1, h_2, \ldots, h_N\} \)

\[\text{Dialog acts and intents are equivalent and interchangeably used in this paper.}\]
for each utterance $x_n$ in a dialog $X$ by taking vectors from each [CLS] token. During testing at turn $n$, we may directly reuse these calculated representations $\{h_1, h_2, \ldots, h_{n-1}\}$ until turn $n-1$.

In contrast with other contextual NLU (Wang et al., 2019; Gupta et al., 2019) with hierarchical components, we introduce a GPT-like unidirectional transformer encoder with the hidden size $H_n$ to encode $H \in \mathbb{R}^{N \times H_0}$. It consists $L$ layers of masked multi-head self-attention (MHA), point-wise feed forward network (FFN), residual sublayer and layer normalization. The future time steps are masked for training since we will not have access to future utterances during testing. We will send $H$ as the first layer input $C^1$ and iteratively encode it with two sublayers in Eq 2. Each head $C_i \in \mathbb{R}^{N \times (H_0/h)}$ will be first mapped into a query $C_i^Q$, a key $C_i^K$ and a value $C_i^V$ which participate in the multi-head self-attention. Here $f(\cdot)$ is softmax function. Finally, we will obtain the final contextual dialog representations $C^L$.

$$C^l = FFN(MHA(C^{l-1}, C^{l-1}, C^{l-1}))$$ 

(2)

$$MHA(C_i^Q, C_i^K, C_i^V) = f(\frac{C_i^Q(C_i^K)^T}{\sqrt{H_0}})C_i^V$$

(3)

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

(4)

3.2 Knowledge Attention

Humans could naturally associate contexts with relevant knowledge to predict semantics. Here we elaborate on how we can leverage current contexts $C^L = \{c^L_n\}$ and a relevant knowledge base $K_G$ to induce the intents and slots for each utterance $x_n$.

3.2.1 Knowledge extraction

The first step is to gather all necessary knowledge triples $\gamma = \{h, r, t\}$, which are head $h$ and tail $t$ entities with their relation $r$, related to the current utterance $x_n = \{w_{n}^1, w_{n}^2, \ldots, w_{n}^T\}$. For each word $w_{n}^i$, we first retrieve a list of triples with the exactly same head entity being $w_{n}^i$ from a knowledge base $K_G$. If no head entities are matched, we instead seek entities that has a substring of $w_{n}^i$. Each triple in the pretrained $K_G$ (Bordes et al., 2013) has a pre-given relation weight $w_{r} \in [0, 1]$. For each $w_{n}^i$, we select $|K|$ triples that have the largest $|K|$ weights as the final word-level knowledge $k_{n}^i$. We will finally obtain a $T$ length knowledge sequence $K_n = \{k_{n}^1, k_{n}^2, \ldots, k_{n}^T\}$ gathered from each word $w_{n}^i$. In case of non-alphabetic or out-of-vocabulary (OOV) words with no match in $K_G$, we instead replace their $K_n$ as zero vectors to represent agnosticism of knowledge.

3.2.2 Global awareness

To improve the heterogeneous information fusion between contexts and knowledge, after obtaining the knowledge sequence $K_n = \{k_{n}^i\}$ (i.e. total $T \times |K|$ triples $\gamma = \{h, r, t\}$), we aim to obtain the context-attended knowledge sequence $V_K = \{v_{n}^i\}$ by selecting the most appropriate knowledge (i.e., removing redundant knowledge noise) within the entire sentence, given each word $w_{n}^i$ and its previous dialog history $c_{n}^l$. Different from the term-level denoising like Zheng et al. (2021) and Wang et al. (2019), to allow phrase-level knowledge sharing,
Figure 2: Knowledge Attention Diagram. (a) Context Attention module will first process the dialog history and produce context-aware vectors for each utterance. (b) Token-level representation will be concatenated with the context-aware vector to calculate the attention weights for every knowledge vector in the utterance. (d) The final context-attended knowledge vector will be the weighted combination of all knowledge vectors.

for each word, we aim to globally select all related knowledge in the sentence after seeing previous turns \(c_L^n\). This will allow us to possibly consider knowledge of words in the same phrase.

Shown in Figure 2, we calculate the vector \(v^n_i\) where \(r^n_{ij}\), \(t^n_{ij}\) are \(j\)-th relation and \(j\)-th tail entity vectors for the word \(w^n_i\). \(W^H, W^R, W^T\) are learnable matrices during training. \([:]\) is the concatenation of two vectors:

\[
v^n_i = \sum_{j=1}^{|K|} \sum_{i=1}^T \alpha_{ij}[r^n_{ij}; t^n_{ij}]
\]

\[
\alpha_{ij} = \exp(\beta_{ij}) / \sum_{j'=1}^T \sum_{i'=1}^{|K|} \exp(\beta_{ij'})
\]

\[
\beta_{ij} = (\tilde{h}^n_i W^H)(\tanh(r^n_{ij} W^R + t^n_{ij} W^T))^T
\]

\[
\tilde{h}^n_i = [h^n_i; c^n_L]
\]

to calculate the attention weight \(\alpha_{ij}\) with any of the knowledge \((r^n_{ij}, t^n_{ij})\) related to this utterance (Eq 6, 7). Eventually, we linearly combine all knowledge vectors together to formalize the context-attended knowledge vector \(v^n_i\) (Eq 5). Additionally, to avert the noise from zero-vectors of non-alphabetic word knowledge, we introduce an attention mask to calculate \(\alpha_{ij}\) only on the non-zero knowledge vectors.

### 3.3 Semantic Decoder

After obtaining the context-attended knowledge \(V_K = \{v^n_i\}\), context vectors \(C^L\) and initial token-level vectors \(H\), we adopt two BiLSTMs to predict multiple dialog acts and slots which exhibit the sequential information in BIO scheme.

\[
H_{act} = \text{BiLSTM}([\tilde{H}; V_K])
\]

\[
H_{slot} = \text{BiLSTM}([H; V_K])
\]

For dialog act detection, we concatenate \(V_K\) with the fused context \(\tilde{H} = ([H; C^L])W^H\) from the attention mechanism and serve as the inputs of BiLSTM. For slot filling, since the task focuses more on token-level information for decision, we only concatenate raw token-level representations and \(V_K\) to be inputs of another BiLSTM, which empirically works better. Finally, we can generate logits \(\hat{y}_{act} = \sigma(H_{act} w_{act})\) by transforming \(H_{act}\) with \(W_{act} \in \mathbb{R}^{H_L \times |V^n|}\) and a sigmoid function \(\sigma\). \(H_L\) is LSTM hidden size and \(|V^n|\) is the size of dialog act set. Likewise, we compute \(\hat{y}_{slot} = \text{softmax}(H_{slot} W_{slot})\). Total loss will be the combination between the binary cross entropy loss based on \(\hat{y}_{act}\) and the cross entropy loss based on \(\hat{y}_{slot}\) as shown in Eq 11, 12. Finally, the joint objective is formulated as the sum of \(L_a\) and \(L_s\).

\[
L_a \triangleq - \sum_{n=1}^N \sum_{a=1}^{|V^n|} (y^n_a \log(\hat{y}^n_a) + (1 - y^n_a)\log(1 - (\hat{y}^n_a)))
\]

\[
L_s \triangleq - \sum_{n=1}^N \sum_{t=1}^T \sum_{s=1}^{|V^n|} (y^{n,t}_s \log(\hat{y}^{n,t}_s))
\]

### 4 Experiment Setting

#### 4.1 Experimental setup

We evaluate our proposed framework on two large-scale dialog datasets, i.e. Microsoft Dialog Challenge dataset (MDC) (Li et al., 2018b) and Schema-Guided Dialog dataset (SGD) (Rastogi et al., 2019). MDC contains human-annotated conversations in
three task-completion domains (movie, restaurant, taxi) with total 11 dialog acts and 50 slots. SGD entails large-scale task-oriented dialogs over 20 domains ranging from travel, weather to banks, etc. It has total 18 dialog acts and 89 slots. To compare the relevant knowledge usage in different domains and save computational resources, we randomly select 1k dialogs for each domain in MDC and two restaurant and flights domains from SGD for total 5k dialogs in 6:1:3 train, validation, test ratio. For SGD, Restaurant domain is chosen to compare with that of MDC and Flights domain is the one not existing in MDC. Each utterance is labeled with one or more dialog acts and several slots.

4.2 Baselines

We compare our models with several competitive baselines which sequentially include more features:
- **MID-SF** (Rashmi Gangadharaiah, 2019) considers joint multi-intent and slot detection in use of BiLSTMs.
- **ECA** (Chauhan A., 2020) encodes the dialog context with LSTM for joint tasks.
- **KANLUM** (Wang et al., 2019) extracts knowledge from the knowledge base and incorporates dialog history for joint tasks.
- **ERNIE** (Zhang et al., 2019b): We take ERNIE backbone to integrate knowledge entities and take the token and entity outputs for intent detection and slot filling directly.
- **LABAN** (Wu et al., 2021b) leverages label information to construct a latent semantic space for utterance projection. It is mainly for the multiple intent detection task only.
- **CASA-BERT** (Gupta et al., 2019) encodes the context with sentence2token and DiSAN which we replace with BERT for fair comparison with other BERT-based models.

We also perform several variations of our proposed framework to conduct the ablation study with the following detailed descriptions.

- **Less-Relevant knowledge triples (LR-KA)**: We replace the top $|K| \setminus 2|K|$ knowledge triples with the less related knowledge triples ranked from $|K| \sim 2|K|$ (from relation weights in $K_G$) to perform sensitivity analysis on the quality of knowledge.
- **Word-Level knowledge attention (WL-KA)**: We use the attention-based filter (AF) (Wang et al., 2021b) to perform token-level knowledge attention instead of sentence-level attention in our framework.
- **Transformer decoder (Trans)**: We replace the semantic decoder (§ 3.3) with a transformer decoder to both predict dialog acts and slots.

4.3 Implementation details

We adopt the pretrained BERT$_{base}$ (Devlin et al., 2019) as our utterance encoder. Context attention transformer has $L = 6$-layer attention blocks with 768 head size and 4 attention heads. The max sequence length is 60. We use ConceptNet knowledge base (Speer et al., 2018) to obtain relevant knowledge for attention. It involves many crowdsourced and expert-created resources like DBPedia, OpenCyc and WordNet with 1.5M word entities connected with weighted edges (relation). Each word or relation is represented as a dense 100-dim vectors by adopting TransE (Bordes et al., 2013) learning mode. Each knowledge also contains an ExternalURL to represent the external source. We retrieve $|K| = 5$ most related knowledge from each word based on weights assigned on the edges. Both LSTMs have 256 hidden units. We use the batch size of 2 dialogs for MDC and 1 for SGD. In all training, we use Adam optimizer with learning rate as $5e-5$. The best performance on validation set is obtained after training 30 epochs on each model. For metrics, we report the dialog act accuracy (exact match) and slot filling F1 score. Here we only consider a true positive when all BIO values for a slot is correct and forfeit ‘O’ tags.

5 Main Results

5.1 Main results

Table 2 shows our main results on the joint task performance. MID-SF with only LSTMs has relatively inferior performance on both datasets especially in SGD. ECA by taking dialog contexts into consideration has much greater increase in SGD than in MDC. ERNIE and KANLUM have better slot filling performance which suggests the importance of further knowledge induction. Leveraging BERT-based encoder seems to substantially increase semantic visibility in ERNIE, CASA-BERT and our proposed framework, while introducing dialog contexts additionally gives better dialog act detection performance in CASA-BERT and our model. Eventually, our proposed framework beats all baselines both in MDC and substantially in SGD, by more
efficiently incorporating external knowledge and dialog contexts with the proposed global awareness attention mechanism.

5.2 Ablation analysis

To better estimate the effectiveness of each module of our best model, we conduct ablation experiments in Table 3. We ablate or replace each component from CKA to observe the performance drops. First, we could see knowledge quality may affect the performance of joint tasks where most performance drops are observed with LR-KA, while we found that slot accuracy may increase if the overall extracted knowledge is less relevant to utterances. To note, the word matching accuracies in the knowledge base are 78.12% (MDC) and 80.97% (SGD), which indicates that there is still about 20% of zero vectors introduced as redundant noises. Second, considering global knowledge across the entire sentence has overall better performance than only word-level knowledge, where knowledge of some phrases should be treated jointly. Finally, we see a single transformer decoder may still entangle the act and slot information by updating gradients simultaneously with poorer performance.

By removing the entire knowledge attention module, we could see a larger accuracy decrease in slot filling tasks, denoting the necessity of external knowledge in enriching the current word representations. By substituting a LSTM on top of BERT for our context attention module (CA), we obtain poorer performance in dialog act detection. By replacing two LSTMs with fully connected layers after knowledge attention, the performance drops especially in SGD. Overall, we observe dialog act detection relies more on contexts while slot filling tasks may concentrate on inter-utterance relations where external knowledge benefits more instead.

5.3 Further Discussion

Could knowledge amend the data scarcity? We also study how knowledge could contribute to the joint tasks when resources are scarce. Figure 3 shows the performance changes with different numbers of training data. We found that inducing the knowledge will have the positive effect on both tasks. In the few-shot setting, we see the performance difference enlarges where knowledge becomes beneficial to enrich the external information aside from data itself. However, knowledge becomes less useful when we have extreme low dataset particularly for slot detection in MDC. Introducing more MDC data at a certain point may contradict with the external knowledge data base that possibly makes models hard to generalize, while it helps dialog act detection that amends the training instability from data scarcity.

Does global knowledge help non-alphabetic slots? We are interested if knowledge for other words would also help with the slot prediction of
non-alphabetic words. Table 4 shows the results for each non-alphabetic slot for our global and local attention models. Since there is no knowledge for the non-alphabetic words, we observe an overall 2% increase by inducing global attention. Contexts are beneficial especially for slots associated with rating, money and address, which should be likely inferred by other keywords near them. However, introducing more knowledge noises may not help to predict time and zip code since they are rather independent to contexts.

5.4 Knowledge Attention

In Figure 4, we visualize the attention heatmap of tokens with their slot labels vs. all knowledge triples from each token. First, we focus on the rows of the heat map. Without attached knowledge for the words like numbers or punctuations, their attention weights are perceived blank across all tokens in the utterance. Second, for valid attention weights, we found the knowledge corresponding to keywords like ‘you’, ‘with’, ‘restaurant’ and ‘Antioch’ are most adopted for overall knowledge representations across all the utterance. It substantiates the belief that the overall semantics of the utterance may be driven by these valued words.

In Table 5, we further show an utterance example with some highlighted words including ‘you’, ‘restaurant’ and ‘Antioch’ with their extracted knowledge and weights for semantic detection. We take the average of all attention weights across all tokens for that knowledge triple; then normalized across the knowledge triples in the same word (head). We could see ‘you’ as an object is most adopted to clarify the user being offered and informed counts. Then we observe that the knowledge triple (restaurant, atl, city) where restaurant is at a location of the city is most recognized to illustrate the relations of restaurant and city tags. Finally, knowledge for ‘Antioch’ keyword is mostly relevant to a country which is conducive when the system seldom sees this word during training. But without further contexts, our model believes ‘Antioch’ is more of a part of Turkey.

6 Related Work

Intent detection and slot filling are two main NLU tasks (Weld et al., 2021). Many classification or clustering approaches (Sarikaya et al., 2011; Raymond and Riccardi, 2007; Liu et al., 2017; Wu and Juang, 2022a) had been proposed for single intent detection. However, treating two tasks separately may experience error propagation. Liu and Lane

---

**Table 4**: F1 scores of non-alphabetic slots in overall SGD dataset when using all (CKA-NLU) or word-level (WL-KA) knowledge.
(2016) first proposed an attention-based LSTM network to model the correlations between intents and slots. Li et al. (2018a) proposed the gating mechanism for better self-attention on joint tasks, which is not scalable for longer sequences. Wang et al. (2018) instead proposed the bi-model to directly model the cross impacts and Zhang et al. (2019a) utilized capsule neural networks. Memory networks are also popular choices to model long-range dependency (Wu et al., 2021a). However, a single utterance may have many intents. Qin et al. (2019) proposed a stack-propagation networks to predict intents on each token. Rashmi Gangadhariah (2019) and Qin et al. (2020) considered the dynamic interactions between two tasks by jointly detecting multiple intents. Wu et al. (2021b) extended the multiple intent scenario with zero-shot cases. These methods nevertheless restrict their resources to current utterances for prediction where we consider the multi-turn dialogs jointly where dialog acts could be context-sensitive (Bothe et al., 2018).

**Contexts and knowledge**

Contexts are also critical for dialog understanding. Bertomeu et al. (2006) first studied the contextual phenomena in words. Bhargava et al. (2013) and Shi et al. (2015) then introduced contextual signals to the joint intent-slot tasks. Advanced hierarchical structures are also emphasized to encode multi-turn dialog contexts efficiently (Chauhan A., 2020; Wang et al., 2019; Gupta et al., 2019; Wu et al., 2021c). Knowledge is also another important resource to induce commonsense for understanding. It is widely adopted for knowledge-enhanced pretraining to enrich representations (Liu et al., 2019; Zhang et al., 2019b). In task-oriented dialogs, main emphasis lies in the interaction with task-related knowledge bases (Madotto et al., 2020; Yang et al., 2020). Most of works also focus on open-domain dialog response generation (Zhao et al., 2020; Wang et al., 2021b; Rashkin et al., 2021; Zheng et al., 2021) or task-specific responses (Wang et al., 2021a). However, commonsense knowledge is seldom adopted in NLU. Wang et al. (2019) tried to apply knowledge in NLU but it is not suitable for complex dialog modeling. To amend the gap in modeling such knowledge and context interactions, we follow these previous works’ paradigms and explore the mechanisms of characterizing their mutual effects.

## 7 Conclusion

In this paper, we propose a novel BERT-based knowledge-augmented network to effectively incorporate dialog history and external knowledge in the joint NLU tasks. Compared to recent works

<table>
<thead>
<tr>
<th>Utterance Example in Figure 4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Utterance</strong></td>
</tr>
<tr>
<td><strong>Dialog acts</strong></td>
</tr>
<tr>
<td><strong>Slots</strong></td>
</tr>
<tr>
<td><strong>Knowledge</strong></td>
</tr>
<tr>
<td>you</td>
</tr>
<tr>
<td>restaurant</td>
</tr>
<tr>
<td>Antioch</td>
</tr>
</tbody>
</table>

Table 5: The utterance example in Figure 4 for joint task prediction. Knowledge (Relation, Tail) related to three keywords as head are presented with their attention weights (number after the knowledge). We only show the top four knowledge adopted for each keyword based on the attention weights. ‘hc’ represents ‘has context’, ‘rel’ represents ‘related to’, ‘atl’ represents ‘at location’ and ‘ptof’ represents ‘part of’.

Figure 4: Attention visualization of a single utterance example with respect to all knowledge related to each word. We denote an utterance with tokens followed by their predicted tag in x-axis. For y-axis, each word will have five knowledge triples with each as a single tick. The blank area is where attention weights are zero.
which consider only intra-word knowledge, we instead raise the knowledge awareness by selecting all relevant knowledge triples in an utterance with the current dialog contexts. We found that our framework is verified to be effective in two complex multi-turn dialog datasets where contexts and knowledge are crucial in dialog act detection and slot filling respectively. The visualization shows that our models adopt some key knowledge in particular words and learn to grasp useful information for better interpretability. These context-attended knowledge vectors could be easily applied to downstream dialog state tracking or management tasks.

Limitations

The possible limitations for our works are two-folds. First, the scalability of our method is subject to the size of the knowledge base and the number of incorporated knowledge since selecting from larger knowledge candidates may require more computational memory and training time but with higher performance. Exact string matching between context words and knowledge entities is relatively simple and could be replaced with more advanced semantic matching techniques, which nevertheless may increase model complexity. Second, depending on the domains of datasets to apply, too many out-of-vocabulary words (OOV) with no match in the knowledge base may affect the model performance and our future works will investigate a better solution to replace zero-vectors that are associated with non-alphabetic words.

References


Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling.

Ting Liu, Xiao DING, Yue QIAN, and Yiheng CHEN. 2017. Identification method of user’s travel consumption intention in chatting robot. SCIENTIA SINICA Informationis, 47:997.


Qingyue Wang, Yanan Cao, Junyan Jiang, Yafang Wang, Lingling Tong, and Li Guo. 2021a. Incorporating specific knowledge into end-to-end task-oriented dialogue systems. In 2021 International Joint Conference on Neural Networks (IJCNN), pages 1–8.


Zheng Zhang, Ryuichi Takanobu, Qi Zhu, Minlie Huang, and Xiaoyan Zhu. 2020. Recent advances and challenges in task-oriented dialog system.
A Additional Experimental Setting

We use huggingface transformers (Wolf et al., 2020) to implement our framework and we use two Nvidia 2080Ti GPUs for all model training. The number of model parameters is around 146M. It takes 30 minutes to train 30 epochs for a single model.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019b. Ernie: Enhanced language representation with informative entities.

Xueliang Zhao, Wei Wu, Can Xu, Chongyang Tao, Dongyan Zhao, and Rui Yan. 2020. Knowledge-grounded dialogue generation with pre-trained language models.