6 Related Works

Our work is most closely related to previous works in dialogue state tracking and question answering. Early models of dialogue state tracking (Thomson & Young, 2010; Wang & Lemon, 2013; Henderson et al., 2014) rely on handcrafted features to extract utterance semantics, and then use these features to predict dialogue states. Recently Mrkšić et al. (2017) propose to use convolutional neural network to learn utterance n-gram representation, and achieve better performance than handcrafted features-based model. However, their model maintains a separate set of parameters for each slot and does not scale well. Models that handles scalable multi-domain DST have then been proposed (Ramadan et al., 2018; Rastogi et al., 2017). Zhong et al. (2018) and Nouri & Hosseini-Asl (2018) propose a global-local architecture. The global module is shared by all slots to transfer knowledge between them. Ren et al. (2018) propose to share all parameters between slots and fix the word embeddings during training, so that they can handle new slots and values during inference. However, These models do not scale when the sizes of value sets are large or infinite, because they have to evaluate every (domain, slot, tuple) during the training. Xu & Hu (2018) propose to use a pointer network with a Seq2Seq architecture to handle unseen slot values. Lee et al. (2019) encode slots and utterances with a pre-trained BERT model, and then use a slot utterance matching module, which is a multi-head attention layer, to compute the similarity between slot values and utterances. Rastogi et al. (2019) release a schema-guided DST dataset which contains natural language description of domains and slots. They also propose to use BERT to encode these natural language description as embeddings of domains and slots. Wu et al. (2019) propose to use an encoder-decoder architecture with a pointer network. The source sentences are dialogue contexts and the target sentences are annotated value labels. The model shares parameters across domains and does not require pre-defined domain ontology, so it can adapt to unseen domains, slots and values. Our work differs in that we formulate multi-domain DST as a question answering problem and use reading comprehension methods to provide answers. There have already been a few recent works focusing on using reading comprehension models for dialogue state tracking. For example, Perez & Liu (2017) formulate slot tracking as four different types of questions (Factoid, Yes/No, Indefinite knowledge, Counting and Lists/Sets), and use memory network to do reasoning and to predict answers. Gao et al. (2019) construct a question for each slot, which basically asks *what is the value of slot i*, then they predict the span of the value/answer in the dialogue history. Our model is different from these two models in question representation. We not only use domains and slots but also use lists of candidate values to construct questions. Values can be viewed as descriptions to domains and slots, so that the questions we formulate have richer information about domains and slots, and can better generalize to new domains, slots, and values. Moreover, our model can do both span and value prediction, depending on whether the corresponding value lists exists or not. Finally, our model uses a dynamically-involving knowledge graph to explicitly capture interactions between domains and slots.

In a reading comprehension (Rajpurkar et al., 2016) task, there is one or more context paragraphs and a set of questions. The task is to answer questions based on the context paragraphs. Usually, an answer is a text span in a context paragraph. Many reading comprehension models have been proposed (Seo et al., 2017; Yu et al., 2018; Devlin et al., 2019; Clark & Gardner, 2018; Chen et al., 2017). These models encode questions and contexts with multiple layers of attention-based blocks and predict answer spans based on the learned question and context embeddings. Some works also explore to further improve model performance by knowledge graph. For example Sun et al. (2018) propose to build a heterogeneous graph in which the nodes are knowledge base entities and context paragraphs, and nodes are linked by entity relationships and entity mentions in the contexts. Zhang et al. (2018) propose to use Open IE to extract relation triples from context paragraphs and build a contextual knowledge graph with respect to the question and context paragraphs. We would expect many of these technical innovations to apply given our QA-based formulation.

7 Conclusion

In this paper, we model multi-domain dialogue state tracking as question answering with a dynamically-evolving knowledge graph. Such formulation enables the model to generalize to new domains, slots and values by simply constructing new questions. Our model achieves state-of-the-art results on MultiWOZ 2.0 and MultiWOZ 2.1 dataset with a 5.80% and a 12.21% relative improvement, respectively. Also, our domain expansion experiments show that our model can better adapt to unseen domains, slots and values compared with the previous state-of-the-art model.