Towards a Dialogue System with Long-term, Episodic Memory

Dan Kondratyuk and **Casey Kennington**
Boise State University
caseykennington@boisestate.edu and dankondratyuk@u.boisestate.edu

Abstract

Intelligent personal assistants lack long-term memory. We propose graph databases as a extensible solution to this problem by representing relevant knowledge as entities, properties, and relations between them. We demonstrate through two experiments that our approach lends itself to a system that can improve natural language understanding by updating its knowledge dynamically in a generalizable and interpretable fashion.

1 Introduction

Despite an increase in popularity, usefulness, and user adoption, intelligent personal assistants (PAs) have significant areas for improvement. The following interactions between a known PA system (S) and a user (U) illustrates one of those areas:

(1) a. U: Hey S, call my mom
b. S: I don't know "mom"
c. U: My mom is Martha
d. S: OK, calling Martha.

(2) a. U: Hey S, call my mom
b. S: I don't know "mom"

By uttering *OK*, the system makes the user think that system signaled understanding. However, the user was later surprised that the system misunderstood, as evidenced in (2-a). This example illustrates a system that lacks grounding (i.e., establishing and building mutual understanding) between the user and the system (Clark, 1996); more specifically in this case the system had no *memory* to store and recall facts about users and interactions in order to build mutual understanding. For this kind of grounding to be accomplished, system memory should dynamically update with each interaction. In this paper, we explore the role of memory in dialogue systems as step towards systems that can better ground conversationally with users.

2 System Overview

The experiments below demonstrate the dynamic nature of graph databases and their ability to express high-level natural language constructs. Our approach involves representing tokens (e.g., "Mary") as nodes in the graph and drawing edges between them when there is a temporal or syntactic relation. The semantics can then be interpreted as traversing the graph to find the desired answer to a question.

3 Experiments

We provisioned a Neo4j graph database[1] to store and retrieve graph-related information and produced a Python implementation[2] to evaluate its usefulness.

Experiment 1: NLU with Dynamic Memory The goal of this experiment is to determine the usefulness of a slot-filling natural language understanding system (NLU) using a pre-filled knowledge base. To this end, we apply a basic NLU which, incrementally for each word, performs a query against the graph database; the returned result informs the NLU about which entities are known.

Data & Task We applied the dialogue state tracking challenge (Williams et al., 2013) data from the Facebook bAbI dataset (task 6) (Kim et al., 2017). Below is an example dialogue between system S and user U:

(3) a. S: hello what can i help you with today

[1] https://neo4j.com

[2] https://github.com/hyperparticle/graph-nlu

b. U: can you make a restaurant reservation with indian cuisine for six people in a cheap price range
c. S: where should it be
d. U: in the west part of town please

By the end of this dialogue, the system has filled in a 3-slot frame such as the following:

$$\begin{bmatrix} \texttt{cuisine} & \text{indian} \\ \texttt{location} & \text{west} \\ \texttt{price} & \text{cheap} \end{bmatrix}$$

We constructed the knowledge base by creating restaurant nodes, pointing them to their three corresponding property nodes, and merging them all together. The task reduces to finding the restaurant nodes that most closely match the frame.

Results The baseline slot accuracy is 45% over 405 frames. By performing direct lookups of words with potential property nodes in the database and filling them when a match was found, we improved this to a 97% accuracy on the evaluation set. Compare this to Zilka and Jurcicek (2015) which used an RNN achieving a 98% accuracy.

Experiment 2: Interactivity using Dynamic Memory Experiment 1 showcased the usefulness of the graph database for NLU (i.e., a retrieval only task), this experiment showcases updating knowledge as well as retrieving knowledge as would be required in an interactive system.

Data & Task For this experiment, we apply our approach to the Facebook bAbI data (tasks 1-3), a synthetic dataset for testing a model's ability to store facts and reason over them (Kim et al., 2017). The following shows an example of Task 2: *Two Supporting Facts*:

(4) a. Mary moved to the bathroom.
b. Mary picked up a football.
c. Mary went to the hallway.
d. Mary put down the football.
e. Mary moved back to the bathroom.
f. Where is the football?

The task in this experiment is to correctly answer the questions. Each task has between 2-5k statements and 1k questions.

As opposed to Experiment 1, once the semantic meaning is known, the system must update its state of the world. We accomplish this by encoding each statement as nodes and edges in a graph, merging them into the graph database, and performing a traversal on the graph to achieve an answer for a given question.

By extracting each statement into a $(subject, relation, object)$ triple, we represent each component as a node, connecting them via edges, and merging the subject and object nodes together. Then we construct a linked list of relation nodes via edges in the order the dialogue presents them, so that we can dynamically update the state of the graph and traverse it to find the answer to any question given.

Results We achieved a 100% accuracy for Task 1 and Task 2, and 80% for Task 3 on the evaluation set. We compare this to to Kumar et al. (2015), which used a gated recurrent neural network to achieve 100% accuracy on all three tasks.

4 Conclusion & Future Work

We conclude that graph databases show promise in representing relationships between relevant entities and their properties for the purposes of incremental NLU and interactive dialogue. They not only enable quick lookup due to index-free adjacency, but can also update their knowledge dynamically without forgetting previous facts. This behavior is crucial for constructing an interactive dialogue system that can remember relevant pieces of information over the short to very long term. For future work, more experiments are necessary to capture the types of scenarios for which the system can be most suitable.

References

Herbert H Clark. 1996. *Using Language*. Cambridge University Press.

Seokhwan Kim, Luis Fernando D'Haro, Rafael E Banchs, Jason D Williams, and Matthew Henderson. 2017. The fourth dialog state tracking challenge. In *Dialogues with Social Robots*, pages 435–449. Springer.

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, and Richard Socher. 2015. Ask Me Anything: Dynamic Memory Networks for Natural Language Processing. *Proceedings of ICML*, 48.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The Dialog State Tracking Challenge. In *Sigdial*, number August, pages 404–413, Metz, France, aug. Association for Computational Linguistics.

Lukás Zilka and Filip Jurcicek. 2015. Incremental LSTM-based Dialog State Tracker. *CoRR*, abs/1507.0.