Neural Semantic Parsing

Michael Staniek

Department of Computational Linguistics
Heidelberg University

October 25, 2021
1. Organisation
Grading

- 30% Presentation
- 30% Participation
- 40% Report/Project
Presentation

**Presenters work**
- Read the paper, understand it, present it
- Either:
  - Prepare 2 questions for your own paper that the others have to answer
  - Prepare the answers to the questions that others send to you
- Direct the discussion at the end

**Participants work**
- Read the paper, understand it,
- Either:
  - Answer 2 questions for that paper
  - Send some questions to the corresponding papers
- Participate
• Please send me 3 paper preferences until the 01.11.2021 so that I can puzzle out who gets which paper
• Its not first come first served, so take your time. If more people want a paper, I start rolling in Python
• I planned with less than 14 people that want ECTS. If we are more, I will select more papers, or if you are feeling adventurous, you can send me paper proposals
• Deadline: 31.03.2022 23:59:59:999
• But: You can already start now
• Project does not have to depend on your chosen paper, same with term paper
• If we have much fewer people participating, I’m also fine with second presentations.
The Task

Query:
area[name="Paris"]-> a;
node(area.a)[name="Gare du Nord"]-> b;
node(around.b;1000)[tourism="hotel"][wheelchair="yes"]; out;

Black-Box Model

Answer:

Figure: The general task that has to be solved
Neural Semantic Parsing

- We want to "translate" natural language queries into their corresponding structured query language
- Such languages can be SQL (Text-to-SQL), Overpass QL (Text-to-Overpass), NLMaps (Text-to-MRL), Hearthstone (Text-to-Python)
The Task

NAME: Darkscale Healer
ATK: 4
DEF: 5
COST: 5
DUR: -1
TYPE: Minion
PLAYER: Neutral
RACE: NIL
RARITY: Common
DESCRIPTION:

```python
class DarkscaleHealer(MinionCard):
    def __init__(self):
        super(), __init__="Darkscale Healer", 5,
        CHARACTER_CLASS.ALL, CARD_RARITY.COMMON,
        battlecry = Battlecry(Heal(2),
        CharacterSelector())

    def create_minion(self, player):
        return Minion(4, 5)
```

**Figure:** Example from the Hearthstone dataset
Papers can be roughly sorted into the following fields:

- Dataset
- State of the Art
The Problem

Figure: Example from the NLMaps task dataset
The Goal: Interactive Data Annotation to Resolve Ambiguities

"Where bars in bradford?"

I am now looking for pubs, is that okay?

No, i meant bars!

query(area(keyval('name','Bradford')),nwr(keyval('amenity','pub')),qtype(latlong))

query(area(keyval('name','Bradford')),nwr(keyval('amenity','bar')),qtype(latlong))
The End