**Motivation**

**Goal:** Modify the semantic parser’s behavior at test time.

**Example:** The model is trained on training data:

- make a call to Jay Brown’s mom
- send a text message to Jack

At test time, a developer wants to add a new domain to the parser. The parser should produce MESSAGING parses given a few examples:

- MESSAGING
  - [IN send message = [SL recipient = Jack]]

Additionally, we want to modify the parser’s behavior without additional training:

- Less computation resources.
- Stability: Avoid model churn.
- Faster development: Update the parser and immediately see the result.
- Customization: Clients can modify the parser without touching the model’s params on the server.

**Procedure:** Given a query:

- The retriever retrieves related exemplars (e.g., training examples with similar queries) from a retrieval index.
  - The index initially contains training examples, but can be modified at test time.
  - Retrieval score = dot product of embeddings from Universal Sentence Encoder.
  - The seq2seq generator takes the exemplar-augmented query and produces a parse.
  - We fine-tune T5 on (augmented input, output) pairs.

**Intuition:**

- The generator learns to use (or ignore) additional information from the exemplars.
- We can modify the parser’s behavior by manipulating the retrieval process (e.g., augment the index). This can be done at test time without additional model training.

**Increasing faithfulness toward exemplars:** For the modified index to have effects, we want to parser to lean toward using instead of ignoring the exemplars. We propose:

**Method 1:** Mix in anonymized training data. This teaches the generator to rely on the exemplars when generating semantic labels.

**Method 2:** Teach the model to be extra faithful to the exemplars when guiding tags are present.

**Experiments**

**(1) Domain bootstrapping:** Remove 1 domain from the training data. At test time, add 100 examples of that domain to the index.

<table>
<thead>
<tr>
<th>Method</th>
<th>Test Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>seq2seq (T5)</td>
<td>85.1</td>
</tr>
<tr>
<td>CASPER (+anon)</td>
<td>86.4</td>
</tr>
<tr>
<td>CASPER (+anon+guide)</td>
<td>85.5</td>
</tr>
</tbody>
</table>

**(2) Parse guiding:** Override the retriever by manually supplying oracle exemplars (= override common model errors / sensitive queries)

<table>
<thead>
<tr>
<th>Method</th>
<th>retrieved exs</th>
<th>oracle exs</th>
<th>oracle exs + guiding tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASPER (+anon)</td>
<td>84.3</td>
<td>88.2</td>
<td>88.3</td>
</tr>
<tr>
<td>CASPER (+anon+guide)</td>
<td>83.9</td>
<td>89.3</td>
<td>93.0</td>
</tr>
</tbody>
</table>

**(3) Schema refactoring:** Split 10 semantic labels into 2 each at test time.

<table>
<thead>
<tr>
<th>Method</th>
<th>retrieved exs</th>
<th>oracle exs</th>
<th>oracle exs + guiding tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASPER (+anon)</td>
<td>84.3</td>
<td>88.2</td>
<td>88.3</td>
</tr>
<tr>
<td>CASPER (+anon+guide)</td>
<td>83.9</td>
<td>89.3</td>
<td>93.0</td>
</tr>
</tbody>
</table>

**CASPER can parse queries in a new domain while maintaining the accuracy on other domains.**

(Result averaged over 5 bootstrapped domains)

**Conclusion:**

- We fine-tune T5 on (augmented input, output) pairs.
- The seq2seq generator can utilize the oracle exemplars better when the tags are on.

**References:**

Ice Pasupat, Yuan Zhang, Kelvin Guu