Cooking Up A Conversation: Fine-tuning SOLOIST With Recipe Recommendation Dialogues

Chih-Yi Lin Thi Ngoc Yen Tran Quy Nguyen

Institute for Natural Language Processing (IMS)

Introduction

- Soloist: An end-to-end, Transformer-based auto-regressive model that subsumes different dialog modules. Pre-trained with GPT-2 using the Schema-Guided Dialogue (SGD) corpora.
- Domain: Recipe recommendation

Research Questions

1. Could Soloist adapt to new domain after **fine-tuning** with *a handful* of dialogues?
2. Could we improve the performance of the system?
   - **Remove negative information** as they were not annotated in pretrained data
   - Feed the model **more training examples**
   - Make the model aware of **database query results (KB)** before generating responses
   - Ask the model to **predict dialogue policy (DP)** before generating responses
   - Let the model learn from mistakes by feeding **Machine Teaching (MT)** dialogues (correcting failed dialogues obtained by chatting with the system)

Implementation

Domain Ontology and Database

	Sys. Req.	User Req.	User Inf.
name		✓	✓
type	✓	✓	✓
ingredients	✓	✓	✓
instructions		✓	
substitutions		✓	
equipment		✓	✓
time	✓	✓	✓
is_vegan	✓	✓	✓
difficulty	✓	✓	✓

Table 1. Domain Ontology. The database includes 22 recipe instances with the left mentioned attributes.

Corpus Creation

- **Annotations:** Each dialogue turn is annotated with the database query results (kb), the belief state (belief), the delexicalized system's response (reply) and the system's action (dp).
- **Example**:
  history: [user : Do you have any good vegan recipe?, system : Are you looking for a main dish recipe?, user : I actually want a soup recipe],
  kb: one,
  belief: is_vegan = yes ; type = soup ; ingredients = not mentioned,
  reply: system : I would recommend [recipe_name_1],
  dp: recommend (name = [recipe_name_1])
  - **Negative inform:** user: I don't like mushrooms. →belief: ingredients = negative mushrooms
  - **Alternative recommendation:** system: How about [recipe_name_1]?, user : Do you have other recipe?→reply: Then I would suggest [recipe_name_2].

Experiments

Settings		Train Ex. (Dia.)
1. Use pre-trained model to generate responses	No fine-tuning	0 (0)
*2. Use all training instances (**baseline**)*	Fine-tuning pre-trained	200 (40)
3. Remove turns with negative values		171 (40)
4. Add more training examples		366 (50)
5. Add database states to context		200 (40)
6. Predict dialogue policies before generate responses		200 (40)
7. Use Machine Teaching dialogues	Fine-tuning baseline	46 (10)
8. Use non-machine-teaching dialogues		31 (10)

Table 2. Experiment settings. Train ex.: number of training examples. **Each turn** serves as a training example. Dia.: number of training dialogues. All models are trained and tested using a segment of 20 dialogues (including negative values).

Evaluation

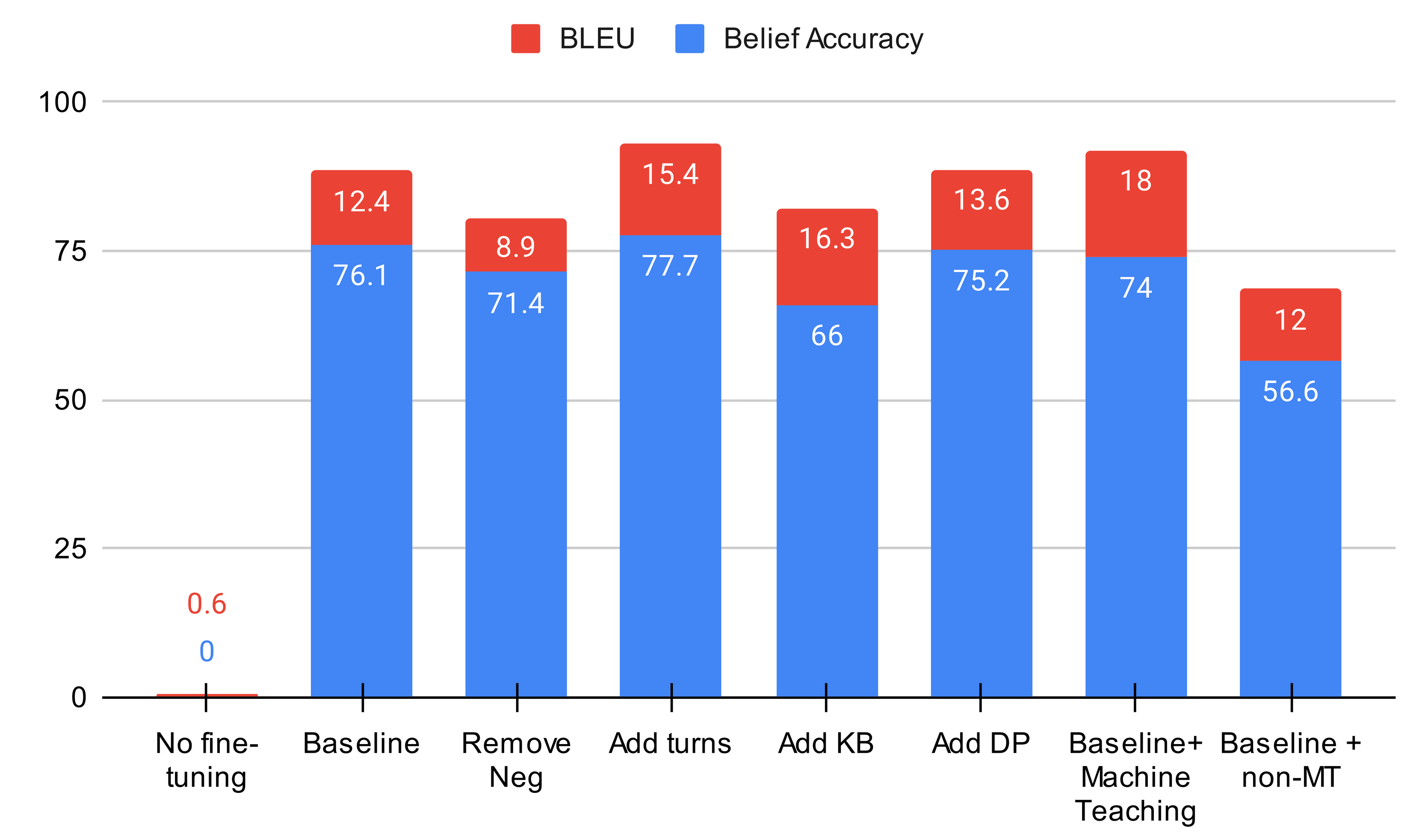

Figure 1. Evaluation results for different settings. Belief Acc.: metric for belief state, BLEU: metric for system response.

- **No Fine-tuning:** Cannot generate belief states, and responses are unrelated to the recipe domain.
- **Baseline, Removing negative values, Adding more turns:** Removing negative turns decreases both Belief State accuracy and BLEU. Training with more training dialogues achieves the best performance.
- **KB, DP:** KB produces more sensible responses (higher BLEU, but surprsingly lower Belief State accuracy). DP could also improve generated responses's BLEU.
- **Fine-tuning the Baseline with Machine Teaching vs. Non-MT:** Machine Teaching enhances BLEU (highest BLEU). However, if non-MT segment is used, the two metrics decrease significantly.

Analysis

Incorrect Belief State and Response Prediction from the Baseline

- Occurs when there is an unseen word or rarely seen scenario in the user utterance, such as negation or substitution of an ingredient.
- An incorrect belief state does **not** necessarily lead to unreasonable system response.

Challenging predictions

- **Negative information in Belief state:** All models struggle if the user gives negative information. Removing negative information in training leads to more inaccurate predictions.
- **Alternative Recommendation in Response:** All models generate partially reasonable responses when the user rejects the first recommendation, but only the KB model successfully recommends an alternative recipe.

Conclusion

- Soloist adapts to new domain with 40 recipe recommendation dialogues.
- Eliminating negative examples reduces performance, adding 10 dialogues improves it.
- Fine-tuning with machine teaching and DP surpasses baseline. KB results in coherent responses but reduces belief state accuracy.
- Future Work: 1. Better metrics for evaluating system response, e.g., human evaluation for goal completion. 2. Improve model performance on negative inform examples.