2022

Assisting End-Users in Creating Chatbots by Improving Training Data

Aparna Roy
University of Delaware

Chris Egersdoerfer
University of North Carolina at Charlotte

Assisting End-Users in Creating Chatbots by Improving Training Data

Aparna Roy, Chris Egersdoerfer, Kostadin Damevski
Department of Computer Science, Virginia Commonwealth University, Richmond, VA

Introduction

As the number of open-source chatbot frameworks continues to grow, there is an ever-increasing need for tools to automatically measure and improve upon domain-specific chatbots.

The training dataset being one of the most impactful pieces to overall chatbot performance, it is the foundational component we aim to optimize.

Method

Part 1: Set Up

1. Randomly split entire training data set into 80% (training set) and 20% (test set)

Part 2: KNN-Based Removal
(testing if removing worst examples improves chatbot)

1. Embed training examples with Sentence-BERT [1]
2. Reduce 384-dimensional embeddings to 2 dimensions with Principal Component Analysis
2. Find 7% of all points that are closest to each example using k-nearest neighbors (KNN) algorithm
3. For each example, X, calculate:

$$\frac{\text{Number of examples of same intent as X}}{\text{Total number of examples in closest 7\%}}$$

4. If the calculated ratio equates to 0, remove the example from the training set
   - If the intent has less than 7 total examples, no examples are removed
5. Retrain chatbot with updated training set and test chatbot with test set

Part 3: Confidence-Based Removal
(setting a baseline using Rasa's NLU system)

1. Remove training examples with lowest confidence based on Rasa's intent prediction model
   - Remove same number of examples using this method as removed with KNN-based removal
2. Retrain chatbot with updated training set and test chatbot with test set

Method

Part 4: Paraphrase-Based Training Data Addition
(testing if increasing training set improves chatbot)

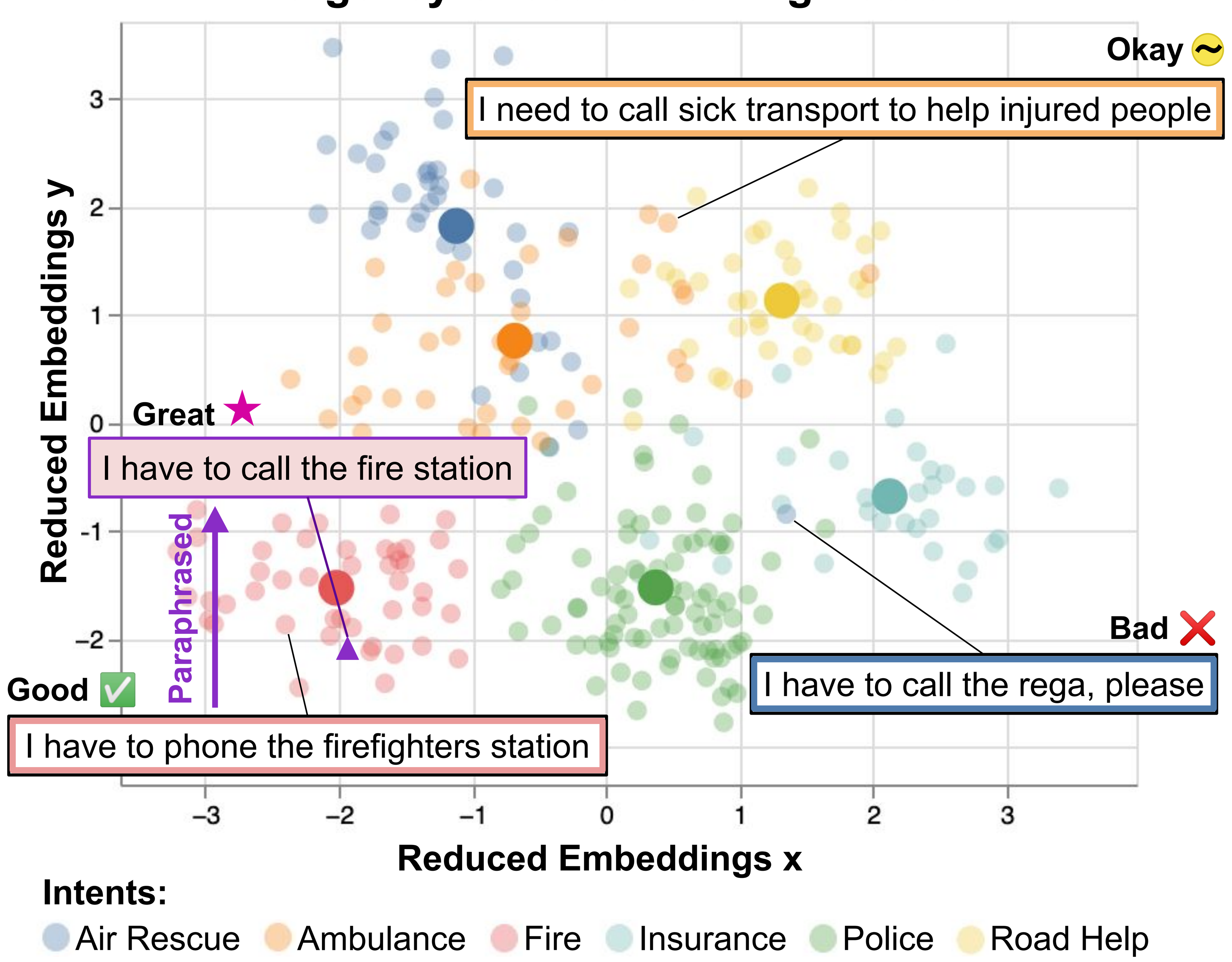
1. Paraphrase each example 3 times [2]
2. Retrain chatbot using larger training set and then test chatbot with test set

Part 5: Repeat

1. Repeat previous steps 5 times and average results

Results

Emergency Chatbot Training Dataset



Chatbot Performance: Original vs After Removal

Emergency Chatbot	Precision	Recall	F1-Score	Accuracy
Original Chatbot (No Removal)	0.8852	0.8348	0.8593	0.8492
Confidence-Based Removal	0.8822	0.8392	0.8602	0.8571
KNN-Based Removal	0.9049	0.8392	0.8708	0.8572

Discussion

Significance of Results

- After testing the confidence-based removal and KNN-based removal on 6 different chatbots, our results showed:
  - Removing training data with lowest confidence (based on Rasa's NLU system) increases average F1-score and accuracy.
  - However, using KNN-based removal method further increases average F1-score and accuracy.
  - Not every chatbot may benefit from this approach, but most will.
- This shows that KNN-based training data removal helps improve chatbot performance by enhancing chatbots' ability to correctly classify user input.

Limitations

- The chatbots that were evaluated were mostly amateur projects, not production-ready chatbots.
  - As a result, the quality of the training data was not always the best, though it is likely close to real-world data.

Future Directions

- We will continue to test if paraphrase-based training data addition improves chatbot performance.
- Does paraphrase-based addition in combination with KNN-based removal improve chatbots more than just removing or adding alone?

References

[1] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using Siamese bert-networks," in *EMNLP 2019*, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1908.10084. [Accessed: 27-Jul-2022].

[2] P. Damodaran, "Parrot: Paraphrase generation for NLU." *GitHub*, 2021. [Online]. Available: https://github.com/PrithivirajDamodaran/Parrot_Paraphraser. [Accessed: 27-Jul-2022].