

ノリツッコミのボケ役エージェントの実装

田嶋 優樹

1. はじめに

近年, Apple の Siri や, Google Assistant といった対話型エージェントが搭載された製品が増えてきている. 対話型エージェントは介護などのユーザとのコミュニケーションが重要な場面での活躍が期待されており, このような場でユーザがエージェントに親しみを感じられるようにすることが重要であると考え.

本研究では, ユーモアを意識したノリツッコミを行う対話型エージェント[1]を参考に, 言語処理の理解を目的に, ボケ役のエージェント実装を行った.

2. ボケエージェントとの対話[1]

本研究ではユーザが入力した文章内の名詞を他の名詞に置換してボケを生成する. ユーザとボケエージェントの対話は下記の手続きで行われる.

Step1: ユーザがエージェントに発言を行う.

Step2: ボケエージェントがユーザの発言内の最もトピックを表す単語を, ユーザの発言のトピックから意味が遠く, 置換元単語から音が近い単語に聞き間違えて聞き返すボケを行う.

3. 実装

3.1 事前準備

名詞 50 単語で構成される名詞リストを作り, これをボケ単語候補群とする. 概念の類似度 u の計算には, Word2vec の similarity メソッドを用いる. このとき算出される値は -1.0 ~ 1.0 であり, 値が大きいほど単語間の概念が類似している.

3.2 置換元単語の決定

ユーザ入力文中の各名詞が, どの程度文のトピックに近いかを表すスコア St を算出し, St が最も高い名詞を置換元単語とする. St は式(1)より求める. i は St を算出する単語の位置, n は文中の名詞の総数であり, $u(k, i)$ は k 番目の単語と i 番目の単語の概念の類似度である.

$$St(i) = \frac{1}{n-1} \sum_{k=1, k \neq i}^n u(k, i) \quad (1)$$

3.3 ボケ単語の決定

入力文中の全名詞とボケ単語候補群の意味の遠

さを表すスコア Ss は, 入力文中の全名詞との単語間の概念の類似度 u の逆数を用いて算出する. Ss は式(2)より求める. l はボケ単語候補群の中の単語の位置, n は文中の名詞の総数であり, $u(k, l)$ は k 番目の単語と l 番目のボケ単語候補の類似度である.

$$Ss(l) = \frac{1}{n} \sum_{k=1}^n \frac{1}{1+u(k, l)} \quad (2)$$

置換元単語とボケ単語候補の音の近さを表すスコア Se は, 単語間の編集距離 de をもとに算出する. Se は式(3)より求める.

$$Se = \frac{1}{1+de} \quad (3)$$

ボケ単語候補ごとの最終的なスコア Sh が最も高くなるボケ単語候補をボケ単語とする. Ss, Se へ重み付け, 合算を行い, Sh を算出する. Sh は式(4)より求める. ws, we は重み係数であり, それぞれ 1.5, 2.0 とする.

$$Sh = ws \times Ss + we \times Se \quad (4)$$

4. 実行例

ユーザ: 写真を撮りたいのに雨が降ってきて悲しい
ボケエージェント: 家臣を撮りたいのに雨が降ってきて悲しい?

ユーザ: 味噌汁がすごく美味しかった

ボケエージェント: 本がすごく美味しかった?

5. おわりに

今回の実験で, 音が近い言葉に置き換わる場合もあれば音が遠い言葉に置き換わる場合もあった. 原因は, ボケ単語の候補の数の少なさだと考える. ユーザに更に親しみを感じてもらうには, ボケ単語の候補を増大し, ツッコミ役エージェントも実装し更なる検討が必要である.

[参考文献]

- [1] 呉健朗, 大西俊輝, 大串旭, 宮田章裕: ノリツッコミを行う対話型エージェント, 情報処理学会インタラクシオン 2022 年出版社