意味的類似度計算システムによるチャットボット FAQ システムの性能向上

栗原健太郎 [1,2]　二宮大空 [2]　友松祐太 [2]
[1] 早稲田大学理工学術院　[2] 株式会社 AI Shift
kkurihara@akane.waseda.jp
{kurihara_kentaro, ninomiya_hirotaka, tomomatsu_yuta}@cyberagent.co.jp

概要

カスタマーサポートや社内ヘルプデスクなどにおける問い合わせ対応に、チャットボットが適用されつつある。AI Shift では、各種サービスにおけるユーザの質問に自動回答するシステムとして Dense Retriever [1] を活用したチャットボット FAQ システムの構築を検討している。Dense Retriever の学習に自社の事業で収集している対話ペアを用いているが、正例と見做している対話ペアの中には、2 文間の内容が大きく異なる品質の悪い対話ペアが存在する。しかし、顧客の多様さとデータ量の多さ故に人手によるそれらの除去は非常に手間がかかる作業となっている。本研究では、意味的類似度計算システムを用いた学習データの自動フィルタリング手法を提案する。実験の結果、提案手法による FAQ システムの性能向上を確認することができた。

1　はじめに

多くの企業や団体が提供するサービスにおけるカスタマーサポートなどにおいて、チャットボットが適用されつつある。チャットボットが提供する機能の一つに、各種サービスにおける「よくある質問」などと呼ばれる Frequently Asked Questions (FAQ) 検索を用いたユーザ質問への回答機能が存在する。FAQ 検索では、企業が保持する FAQ のデータベースに基づき、ユーザ質問に対して最もマッチする回答を得ることができる。

我々は現在構築を検討中のチャットボットの FAQ システムにおける検索手法として、Open-Domain QA で有効とされている Dense Retriever [1] を採用する。Dense Retriever の学習には、自社のチャットボット事業で収集している < ユーザ質問, FAQ 質問 > を対話ペアと見做した対話データを用いる[1)]。本対話データにおいて、ユーザ質問とユーザが選択した FAQ 質問の対話ペアを正例として学習に用いる。

表 1　品質の悪い対話ペアの例

ユーザ質問	FAQ 質問
アンケート	最新情報を教えてください
アカウントが作れない	ログインできない
あああああああああ	定休日はいつですか？

しかし、プロダクトで収集される対話データには、ユーザが選ぶ FAQ 質問の内容が質問内容とマッチしていない品質の悪い対話ペアが含まれているという問題がある。これらを正例と見做して学習することで、Dense Retriever の学習の際にノイズとなる恐れがある。品質の悪い正例の例を表 1 に示す。いずれもユーザ質問と FAQ 質問の内容が大きく異なるため品質の悪い正例と言える。また、複数顧客のデータからランダムサンプリングした対話ペアに対して、筆者による品質の良い正例であるか否かについてのアノテーションを実施した結果、品質の { 良い正例: 455 件, 悪い正例: 328 件 } となっており、品質の悪い正例が多く含まれている。

一方で、多様な顧客から多数の対話ペアを収集していることから、人手での品質の悪い正例の除去は大変手間がかかる作業となっている。人手フィルタリング以外の品質の悪い正例を除去する手段として、FAQ 質問選択後にユーザがさらに選択するフィードバック質問の活用が考えられる。フィードバック質問とは、FAQ 選択後に表示される回答によって課題を解決することができたかを「はい」「いいえ」の 2 択でユーザが回答する質問である。ここで「はい」が選択されたデータのみを収集することで、品質の悪い正例を除去することが可能である。しかし、フィードバック質問に回答するユーザ

1)　対話データの収集方法の詳細については二宮ら [2] の 4 章に原則従う。

は少なく、収集できるデータ量も著しく減少してしまう。そのため、学習データのフィルタリング方法としてはふさわしくない。

そこで本研究では、意味的類似度計算システムを用いた対話ペアの類似度付与による品質の悪い正例の自動フィルタリング手法を提案する。実験の結果、提案手法によるフィルタリングを適用したデータで学習した FAQ システムは、フィルタリング未適用のデータで学習した FAQ システムと比較して性能が向上することを確認することができた。

2 関連研究

Open-Domain QA タスクにおける文書検索では、表層情報による文書検索手法として TF-IDF や BM25 [3] が用いられていた。昨今では密なベクトル表現に基づいた文書検索を行う Dense Retriever が採用されつつあり、QA 分野への適用の流れも生じている。加藤ら [4] は日本語の Open-Domain QA データセット JAQKET [5] を用いた DPR における Retriever の性能評価を実施しており、一定の性能の発揮を報告している。また、言語理解モデルの学習におけるデータの拡張による性能向上の試みも、要約タスク [6] や FAQ タスク [2] などで行われている。Talukdar ら [7] は学習に用いるデータのフィルタリングが言語理解モデルの性能向上に寄与すると報告しているが、その調査は主に SST などの分類タスクを対象としている。

本研究では、2 文間の類似度が低い対話ペアを品質の悪い対話ペアと見做し、回帰タスクである意味的類似度計算 (Sentence Textual Similarity: STS) タスクに帰着させることによるデータの自動フィルタリングを提案する。英語の STS-b [8] や日本語の JSTS [9] などはベンチマークに含まれているデータセットとして言語理解モデルの性能評価に用いられている。2 つの STS データセットの正解類似度は 0（意味が完全に異なる）から 5（意味が等価）の実数値で定義されており、一般的に STS タスク全般で同様に定義される。

3 意味的類似度計算システムを用いた低品質対話ペアのフィルタリング

検討中のチャットボット FAQ システムにおける Dense Retriever の学習に用いる対話データには、2 文間の内容が大きく異なる品質の悪い正例が多く存在している。これらのデータを学習に用いることで、

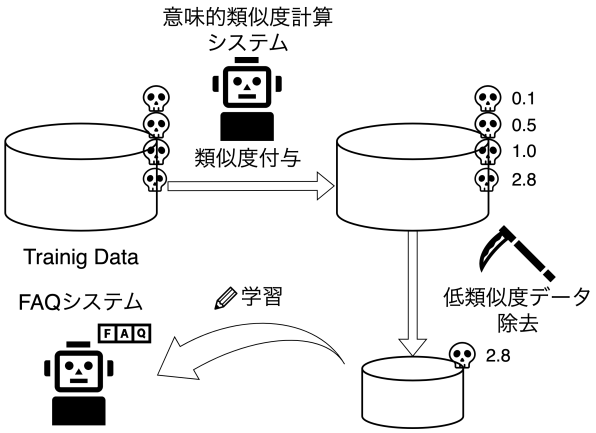

図 1 意味的類似度計算システムを用いた学習データの自動フィルタリングのフロー

モデルの学習の際にノイズとなる恐れがある。しかし、顧客の多様さとデータの多さ故に人手フィルタリングは非常に手間がかかる作業となっている。

本研究では、意味的類似度計算システムを用いた学習データの自動フィルタリング手法を提案する。提案手法のフローを図 1 に示す。意味的類似度計算システムを用いて類似度（以下 s と呼ぶ）を獲得し、s が一定の値以下である対話ペアを除去することによって自動フィルタリングを実施する。その後フィルタリングしたデータを用いて FAQ システムに用いられている Dense Retriever を訓練する。意味的類似度計算システムの構築には、言語理解モデル BERT [10] を用いる。具体的には、日本語の STS データセットである JSTS で BERT の事前学習済みモデルをファインチューニングすることで意味的類似度計算システムを構築する。システムが算出する類似度 s は、JSTS の正解類似度と同様に 0 から 5 の間の実数値となる。

4 FAQ システムの評価実験

4.1 実験設定

自社のチャットボット事業で収集した対話ペアを用いて、チャットボット FAQ システムにおける Dense Retriever の学習・評価を行う。意味的類似度計算システムを用いたデータセットの自動フィルタリングの有効性検証のため、フィルタリングを適用した対話データと適用していない対話データそれぞれで Dense Retriever の学習を行う。評価には Macro Average Top {1, 3, 5} Accuracy を使用し、各顧客データ毎に Top {1, 3, 5} Accuracy を算出した後、全体顧

表 2 学習データのサンプルサイズ

学習データ	件数
Raw Data	178,152
JSTS Data ($\mathcal{S} < 1.0$)	160,684
JSTS Data ($\mathcal{S} < 1.5$)	114,130
vanilla-BERT Data ($\mathcal{S} < 3.2$)	145,937
vanilla-BERT Data ($\mathcal{S} < 3.9$)	21,984

表 3 評価データのサンプルサイズ

評価データ	Known-Domain		Unknown-Domain	
	dev	test	dev	test
Normal-Data	22,270	22,279	684	684
Filtered-Data	6,763	6,777	251	251

客数で平均することで最終的なスコアを得る。

ベースライン手法 提案手法における JSTS 活用の妥当性検証のため、ベースラインとして事前学習済みモデル BERT をファインチューニングせずに意味的類似度計算システムを構築する。本システムは、対話ペアの各文を独立にモデルに入力し、最終層の出力の [CLS] トークンのベクトルのコサイン類似度を算出して類似度を獲得する。ただしコサイン類似度は 0 から 1 の実数値で算出される一方、提案手法の類似度計算システムは 0 から 5 の実数値で類似度を算出する。そこで、ベースライン手法では、コサイン類似度を 5 倍した値を類似度とする。

フィルタリング未適用の Dense Retriever の学習データ（以下 Raw Data と呼ぶ）と、ベースライン、及び提案手法それぞれによるフィルタリング後の学習データ（それぞれ vanilla-BERT Data, JSTS Data と呼ぶ）のサンプルサイズを表 2 に示す。

評価データの設計 ドメインに依らない汎化性能を評価するため、Dense Retriever の学習に用いた顧客データで構成した既知ドメイン（以下 Known-Domain と呼ぶ）と、学習に用いていない顧客データで構成した未知ドメイン（以下 Unknown-Domain と呼ぶ）の 2 種類の評価データを用意する。評価データも、学習データと同様にユーザ質問とユーザが選択した FAQ 質問の対話ペアを正例として収集したデータを用いる。しかし、この収集方法では学習データと同様に評価データにも品質の悪い対話ペアを含んでしまう。そのため、1 節で述べたフィードバック質問でユーザが「はい」と回答した対話ペアのみを抽出したデータでも評価を行う。学習データと同様に収集した評価データを Normal-Data、フィードバック質問を用いてフィルタリングを実施している評価データを Feedback-Data と呼ぶ。

各評価データのサンプルサイズを表 3 に示す。

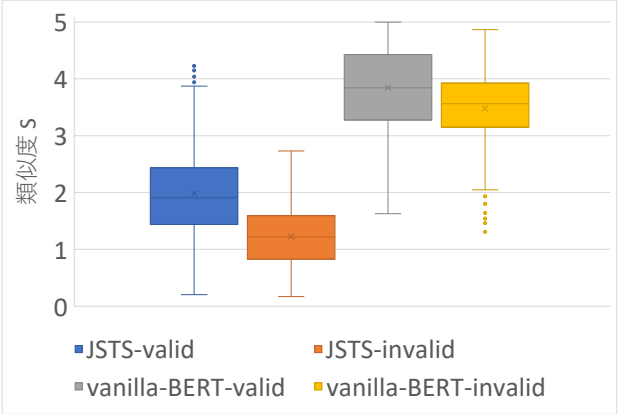

図 2 ランダムサンプリングした対話ペアの類似度の分布を示す箱ひげ図（valid: 品質の良い正例データ, invalid: 品質の悪い正例データに対応している）

除去する対話ペアの類似度の閾値 本実験では、除去する対話ペアの類似度 $\mathcal{S}$ の閾値による FAQ システムの性能差を比較するため、各フィルタリング手法で複数の閾値を設定する。効果的な閾値設定のため、1 節で述べた人手アノテーションした対話データの意味的類似度を、ベースライン、提案手法それぞれで獲得した。アノテーション済みデータの類似度分布を図 2 に示す。提案手法について、$\mathcal{S}$ の閾値を品質の良い正例 (valid データ) の類似度分布の第一四分位、および品質の悪い正例 (invalid データ) の類似度分布の第三四分位に相当する 1.5 程度に設定することで、アノテーション済みデータにおいて、valid データを 1/4 程除去しつつも、invalid データを 3/4 程除去することができる。以上より、提案手法において $\mathcal{S} < 1.5$ を除去する対話ペアの類似度の閾値の 1 つに設定する。しかし、valid データを除去しすぎることによる意味的類似度計算システムの性能低下が懸念される。そこで、valid データを極力除去することなく invalid データを除去するために $\mathcal{S} < 1.0$ も閾値に設定する。ベースライン手法については valid, invalid データの類似度分布の重なりが大きく、効果的な閾値設定が困難である。本実験では、提案手法における閾値設定と同様に、valid データの第一四分位および invalid データの第三四分位に相当する $\mathcal{S} < 3.2, \mathcal{S} < 3.9$ をベースライン手法の閾値に設定する。

その他のハイパーパラメータ 意味的類似度計算システムの構築に用いる BERT、および FAQ システムの構築に用いる Dense Retriever のファインチューニングにおけるハイパーパラメータを表 4 に示す。

表 4 意味的類似度計算システム、および FAQ システムの fine-tuning 時のハイパーパラメータ

	意味的類似度計算システム	FAQ システム
pretrained-model	cl-tohoku/bert-base-japanese-v2	cl-tohoku/bert-base-japanese-whole-word-masking
batch size	8	64
learning rate	5e-5	1e-5
epoch	4	10
max-seq-length	512	64

表 5 各評価手法による FAQ システムの性能評価結果 (3 つのスコアは左から順に Top{1, 3, 5} Accuracy を表す)

Normal-Data				
学習データ	Known-Domain		Unknown-Domain	
	dev	test	dev	test
Raw Data	0.284 / 0.472 / 0.550	0.290 / 0.477 / 0.564	0.302 / 0.489 / 0.570	0.291 / 0.463 / 0.577
JSTS Data ($\mathcal{S} < 1.0$)	**0.296 / 0.500 / 0.584**	**0.305 / 0.504 / 0.589**	**0.365 / 0.574 / 0.638**	0.323 / **0.564 / 0.628**
JSTS Data ($\mathcal{S} < 1.5$)	0.278 / 0.468 / 0.566	0.285 / 0.474 / 0.566	0.352 / 0.534 / 0.586	**0.364** / 0.544 / 0.610
vanilla-BERT Data ($\mathcal{S} < 3.2$)	0.285 / 0.472 / 0.556	0.290 / 0.487 / 0.571	0.344 / 0.534 / 0.606	0.303 / 0.524 / 0.616
vanilla-BERT Data ($\mathcal{S} < 3.9$)	0.231 / 0.402 / 0.483	0.235 / 0.413 / 0.495	0.308 / 0.496 / 0.575	0.325 / 0.511 / 0.564
Feedback-Data				
学習データ	Known-Domain		Unknown-Domain	
	dev	test	dev	test
Raw Data	0.296 / 0.511 / 0.589	0.310 / 0.505 / 0.602	0.232 / 0.369 / 0.531	0.246 / **0.432** / 0.477
JSTS Data ($\mathcal{S} < 1.0$)	0.328 / **0.544 / 0.624**	0.326 / **0.545 / 0.635**	0.281 / **0.546 / 0.592**	**0.303** / 0.430 / 0.492
JSTS Data ($\mathcal{S} < 1.5$)	**0.333** / 0.525 / 0.620	**0.327** / 0.516 / 0.595	**0.289** / 0.530 / 0.567	0.289 / 0.401 / 0.450
vanilla-BERT Data ($\mathcal{S} < 3.2$)	0.305 / 0.501 / 0.579	0.311 / 0.506 / 0.578	0.267 / 0.443 / 0.581	0.299 / 0.412 / 0.479
vanilla-BERT Data ($\mathcal{S} < 3.9$)	0.248 / 0.410 / 0.479	0.246 / 0.412 / 0.489	0.212 / 0.473 / 0.521	0.266 / 0.388 / **0.523**

4.2 結果・考察

Dense Retreiver の推論性能の評価結果を表 5 に示す。全般的に JSTS Data で学習した Dense Retriever が、Raw Data で学習した場合や、vanilla-BERT Data で学習した場合のモデルの結果と比較して Accuracy が高くなっている。JSTS Data ($\mathcal{S} < 1.0$) と Raw Data それぞれで学習した結果を比較した場合、Normal-Data, Known-Domain, test において Top {1, 3, 5} Accuracy はそれぞれ 1.5%, 2.7%, 2.5%上回っており、Normal-Data, Unknown-Domain, test では 3.2%, 10.1%, 5.1%上回っている。一方で、Known-Domain における性能について、vanilla-BERT Data で学習した場合の Accuracy は、Raw Data で学習した場合と同等またはそれ以下の値という結果になっている。この結果は、JSTS によるファインチューニングを行なっていない vanilla-BERT を用いた invalid データの効果的な除去は困難であること、及び JSTS を用いたファインチューニングによる文ペアの意味的類似度の学習の有用性を示している。

Normal-Data の Known-Domain における評価において、JSTS Data ($\mathcal{S} < 1.5$) で学習したモデルの Top 1 Accuracy が Raw Data で学習したモデルと比べて低い。これは、Raw Data で学習したモデルの Accuracy を上回っている JSTS Data($\mathcal{S} < 1.0$) や vanilla-BERT Data ($\mathcal{S} < 3.2$) と比較して、JSTS Data($\mathcal{S} < 1.5$) はデータ数が少なく、valid データが比較的多く除去されたことが原因と考えられる。

一方で、Feedback-Data の Known-Domain における評価では、JSTS Data($\mathcal{S} < 1.5$) で学習したモデルの Top 1 Accuracy が Raw Data で学習したモデルと比べ高い。この結果は、Normal-Data に含まれていてモデルが正解できなかった invalid データが、Feedback-Data ではフィルタリングされていることが原因と考えられる。

5 おわりに

本論文では、Dense Retriever ベースの FAQ システムの学習データの自動フィルタリング手法として、JSTS でファインチューニングした BERT ベースの意味的類似度計算システムの活用を提案した。実験結果は、提案手法による FAQ システムの性能向上を確認することができたことで、人手フィルタリングによる負担の削減に貢献できる可能性を示した。

今後は、T5 [11] や BART [12], などの生成モデルを用いて、FAQ に紐づいた回答文章と答えから質問文を生成することなどによって、データセットの拡張をすることを検討する。さらに、データセット拡張とフィルタリングを相互に実施することで FAQ システムの更なる性能向上を目指す。

参考文献

[1] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 6769–6781, Online, November 2020. Association for Computational Linguistics.

[2] 二宮大空, 邊土名朝飛, 杉山雅和, 戸田隆道, 友松祐太. チャットボット事業における dense retriever を用いた zero-shot faq 検索. 第 96 回言語・音声理解と対話処理研究会 (第 13 回対話システムシンポジウム), 2022.

[3] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at trec-3. In **TREC**, 1994.

[4] 加藤拓真, 宮脇峻平, 西田京介, 鈴木潤. オープンドメイン qa における dpr の有効性検証. 言語処理学会 第 26 回年次大会, pp. 1403 – 1407, 2021.

[5] 鈴木正敏, 鈴木潤, 松田耕史, 西田京介, 井之上直也. Jaqket: クイズを題材にした日本語 qa データセットの構築. 言語処理学会 第 26 回年次大会, pp. 1403 – 1407, 2020.

[6] Bharath Chintagunta, Namit Katariya, Xavier Amatriain, and Anitha Kannan. Medically aware GPT-3 as a data generator for medical dialogue summarization. In **Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations**, pp. 66–76, Online, June 2021. Association for Computational Linguistics.

[7] Arka Talukdar, Monika Dagar, Prachi Gupta, and Varun Menon. Training dynamic based data filtering may not work for NLP datasets. In **Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP**, pp. 296–302, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[8] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In **Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)**, pp. 1–14, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[9] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese general language understanding evaluation. In **Proceedings of the Thirteenth Language Resources and Evaluation Conference**, pp. 2957–2966, Marseille, France, June 2022. European Language Resources Association.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[11] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

[12] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 7871–7880, Online, July 2020. Association for Computational Linguistics.