

畳み込みニューラルネットワークを用いたコミュニティ QA における補足文選択の試み

水越 友裕 高木 友博

明治大学大学院理工学研究科情報科学専攻

{t_mizukoshi, takagi}@cs.meiji.ac.jp

概要 多くのコミュニティ QA では質問に対し、回答として複数の文を書き込むことができる。そのため質問に対する直接的な回答文に加えて、その質問または回答を補足するような文が多く見られる。そこで本稿では質問文と回答文の両方を考慮した補足文を対象に、適切な補足文を質問文と直接的な回答文の関係性から選択するモデルを提案する。手法としてはニューラルネットワークを用いた従来の回答選択モデルを拡張し、補足文を選択するタスクに適用する。また補足文の選択モデルの学習・テストを目的に、既存のデータセットから補足文を手で抽出し、データセットを構築した。この構築したデータセットを用いて、提案手法が有効であるかどうか評価実験を行った。

キーワード コミュニティ QA, 畳み込みニューラルネットワーク, 補足文選択, 分散表現

1 はじめに

近年、ユーザからの質問に対して自動で応答する質問応答システムの分野での研究が盛んに行われている。

最近の質問応答システムにおいては質問文と回答文のペアを、ニューラルネットワークを用いて推論する手法が多く提案され既存の手法より大きく性能を向上した。

一方ユーザの質問と応答から成り立っているサービスとしてコミュニティ QA が挙げられる。コミュニティ QA とはユーザが質問を投稿し、それに対して他のユーザが回答する形のサービスであり、その代表例として「Yahoo!知恵袋」が挙げられる。

このようなコミュニティ QA は一般的に一文だけではなく複数の文を書け、それによってユーザは回答の情報量を多くすることができ、回答の信頼性を高めることができる。

またコミュニティ QA の分野では主にベストアンサー推定が研究の中心であり、質問応答に関する研究は少ない。したがってコミュニティ QA においても既存の質問応答システムのような一問一答とは違い、質問に対して複数の文を選択することができれば、情報量のあるユーザにとって価値の高い回答を自動で提供できると考えられる。

さらにコミュニティ QA でのユーザの複数文の回答を分析すると、質問に対する回答だけでなく、その回答に対して補足をしている文を多く見ることができる。表 1 に具体例を示す。下線部は補足文の内容が考慮しているフレーズである。

表 1 補足文の例

質問文: 東京駅から3時間で行って帰ってこれる範囲で、下町情緒を味わえるところってありますか？
回答文: 下町なら浅草はどうでしょうか。
補足文: 浅草であれば東京駅から神田で銀座線に乗換えて約 20 分です。

ここではコミュニティ QA におけるユーザの回答全体のうち、質問に対して答えを提示している文を回答文、質問文および回答文の内容を補足している文を補足文と呼ぶことにする。このような補足文を自動で生成することができればより情報量が多く、価値の高い情報を提供できると考えられる。

本研究では、コミュニティ QA における質問文と回答文との関係に基づいた補足文の選択を行うために、既存のニューラルネットワークを用いた回答選択モデルを拡張し、適切な補足文を選択することを試みる。

2 関連研究

近年、言語処理の諸問題におけるニューラルネットワークを用いた手法は増加していて、従来の手法に比べて性能が向上している例が多く見られる。

Kim[1]は畳み込みニューラルネットワークを用いて文書の分類を行い、既存手法と同等または性能の向上を実現した。また文書分類に限らず、さまざまな言語処理のタスクに対して適用されている。本研究の元のタスクである回答選択においてもニューラルネットワークを用

いた手法が高い精度を出している. Tan ら[2]は bi-directional LSTM を用いて質問に対して適切な回答を選択するモデルを提案した. Yin ら[3]は BCNN(Bi-CNN)と呼ばれる畳み込みニューラルネットワークを用いた手法で同様のタスクを解くだけでなく, 言い換え認識・含意関係認識のタスクに適用している.

一方コミュニティ QA における回答選択に関する研究は少数である. TREC LiveQA ではコミュニティ QA の自動解答タスクを取り扱っている. 一例として Wang ら[4]は Bi-directional LSTM と BM25 を用いてコミュニティ QA である Yahoo! Answers における回答選択を行っている. しかしながら LiveQA で提案された手法は前述の回答選択と同じであり, 文の役割には注目していない.

3 BCNN(Bi-CNN)[3]

この研究では Yin らの提案した BCNN を拡張し補足文選択タスクに適用するため, まず BCNN の動作機構を説明する.

BCNN は主に入力層, 畳み込み層, プーリング層, 出力層の 4 つに分けることができる. 図 1 に入力層, 畳み込み層, プーリング層を, 図 2 に出力層を示す.

畳み込み層とプーリング層はスタックすることができる.

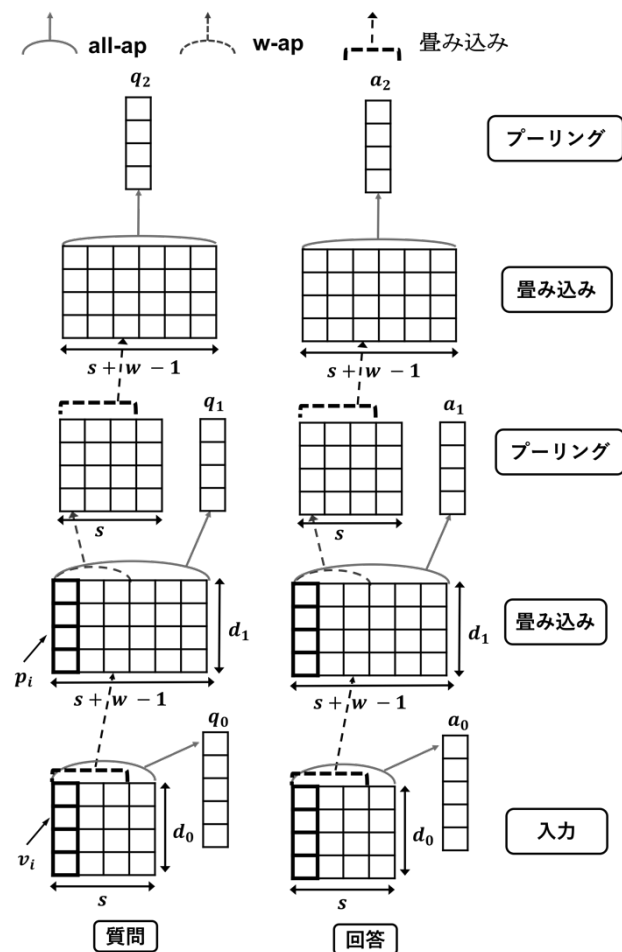


図 1 BCNN の入力層・畳み込み層・プーリング

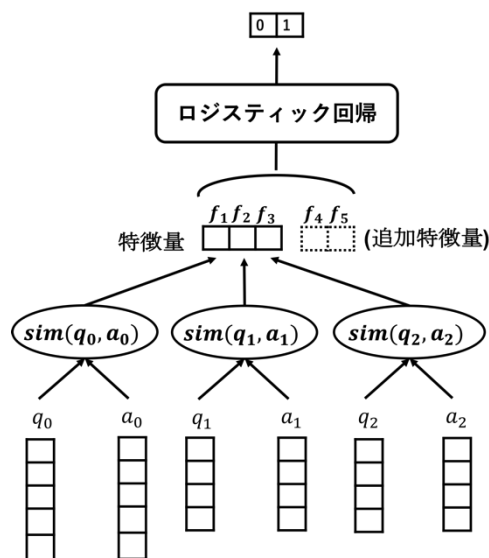


図 2 BCNN の出力層

入力層では, 入力文中の各単語を d_0 次元の分散表現に変換する. ここで変換した各単語の分散表現を v_0, v_1, \dots, v_{s-1} とする.

畳み込み層では入力層で得られた分散表現の畳み込みを行う. フィルタサイズを w とすると, v_i, \dots, v_{i+w-1} を連結した c_i とフィルタ W , バイアス項 b を用いて, 以下の式(1)を用いて特徴 p_i を得る.

$$p_i = \tanh(W \cdot c_i + b) \quad (1)$$

なお特徴 p_i の次元は畳み込みフィルタ枚数 d_1 となる. ここではゼロパディングを行い, 入力層の分散表現の列のサイズを s とすると, 畳み込み後の列のサイズは $s + w - 1$ となる.

プーリング層では畳み込み層で得られた特徴に対して, プーリングを行う. ここでは平均プーリングを用いる. BCNN では all-ap (all-average-pooling)と w-ap(w-average-pooling)と呼ばれるフィルタ幅の違うプーリングを行う. all-ap では入力および畳み込みで得られた特徴を出力層で利用するため, 特徴 p のすべての列の平均をとる. これにより得られた質問ベクトルを q_0, \dots, q_2 , 回答ベクトルを a_0, \dots, a_2 とする. 一方 w-ap では次の畳み込み層で利用するために, フィルタ幅 w でプーリングを行う. w-ap を行った後の列のサイズは s となる.

出力層ではロジスティック回帰により回答候補が質問の回答として適切かどうか 2 値分類を行う. 回帰に用いる特徴量は同じ層における質問・回答の all-ap で得られるベクトルの類似度 f_1, \dots, f_3 とする. BCNN の利点としてロジスティック回帰のため, 特徴量の追加が容易で拡張性が高いことが挙げられる. 回答選択タスクにおいては, 質問文と回答文で共起している単語の数および共起している単語の idf の和が特徴量として追加されている(それぞれ f_4, f_5 とする).

4 提案手法

本章では, BCNN を補足文選択に適用するための拡張方法について述べる.

提案するモデルを図 3 と図 4 に示す.

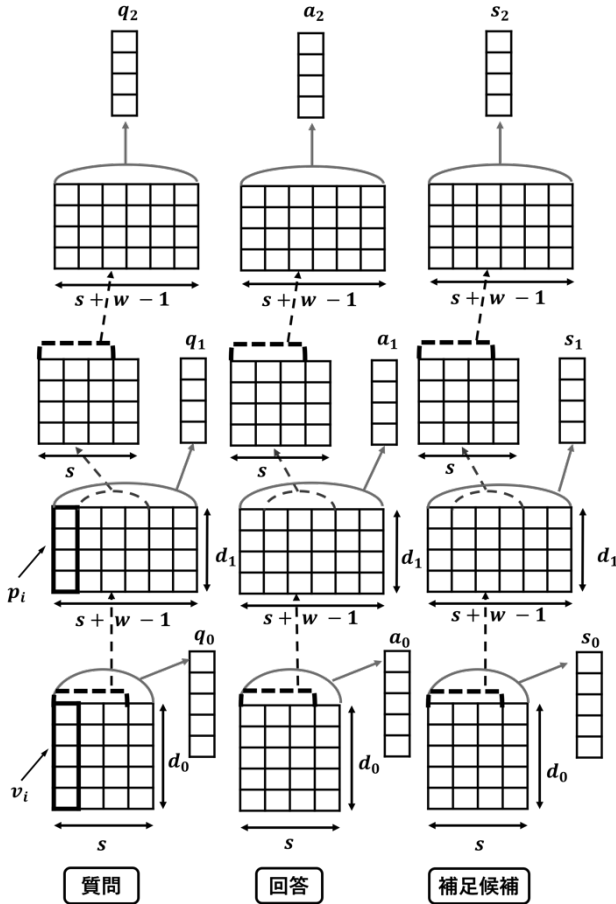


図 3 提案モデルの入力層・畳み込み層・プーリング

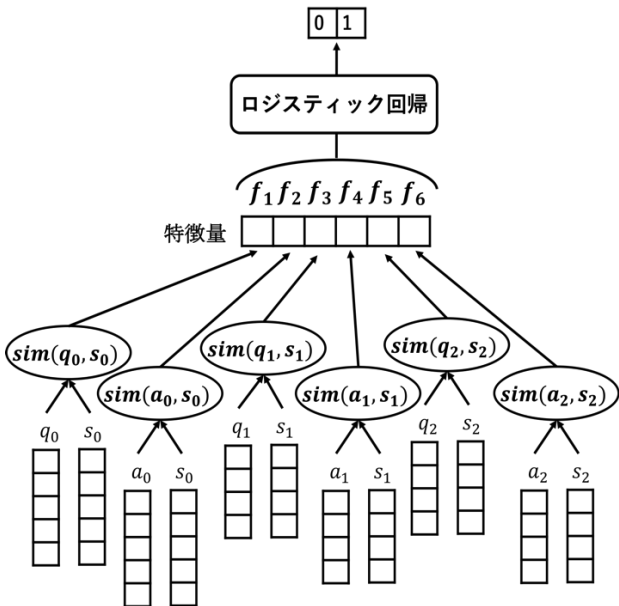


図 4 提案モデルの出力層

ただし all-ap によって得られる補足候補のベクトルを

s_0, \dots, s_2 とする.

モデルへの入力には質問文と回答文と候補の補足文の 3 つであり, 出力として候補の補足が適切であることをロジスティック回帰による 2 値分類で判定する. 質問・回答に加え候補の補足も BCNN と同様の処理を行う. ただしロジスティック回帰に入力する all-ap の類似度の組み合わせを質問と補足, 回答と補足の 2 つに拡張する.

なお, 提案する補足文選択のタスクでは元論文で提案しているアテンションは採用していない. 畳み込みやプーリング時に質問に対する補足のアテンションと回答に対する補足のアテンションの両方が補足文にかかることにより, 類似度を測るために必要な情報を失う可能性があるためである.

5 データセットについて

補足の選択モデルの学習・テストを行うために Yahoo! 知恵袋データ(第 2 版)からデータセットを構築する. ただしカテゴリを地域, 旅行, お出かけに限定している. 補足文が質問と回答の単語やフレーズを考慮している部分が明確であるため, このカテゴリに限定した.

5.1 構築方法

「ユーザの回答全体のうち, 回答文を除く質問と回答の内容の両方を考慮している」という性質を満たす文をここでは正しい補足と定義する. 目的の補足文がある文章から(質問文・回答文・補足の候補文)の組の形で抽出した. 抽出する質問文・回答文・補足文の組は主に上記カテゴリのうち, 質問文の内容が場所を尋ねるものとした. また質問文・回答文・補足の候補文の組には正しい補足/正しくない補足の 2 値のラベルを付与した.

5.2 概要

5.1 節の方法で得られたデータセットの詳細を表 2 に示す. 同じ質問文・回答文ペアに対して, 正しい補足と正しくない補足の両方を含んだ複数の補足候補文が関連付けられている構造となっている.

表 2 構築したデータセットの概要

	学習	テスト
質問・回答ペア数	868	193
質問・回答・補足ペア数	12794	1967
正しい補足数	962	210
正しい補足が存在しない質問・回答ペアの数	442	0
1 文あたりの平均の単語数	14.43	13.05

6 実験

構築したデータセットを用いて提案手法による補足の選択を行う. 評価時には補足候補の文をモデルの出力

の確率の高い順に並べる. 評価指標として MAP(mean average precision)と MRR(mean reciprocal rank)を用いて, 正しい補足をモデルがどの程度選択することができるかを評価する.

6.1 実験設定

文は MeCab で形態素解析を行う. 入力単語の分散表現への変換には word2vec[5]を利用し, 分散表現の次元 d_0 は 200 として, 分散表現が存在しない単語は $[-0.01, 0.01]$ の一様分布で初期化した.

出力層へ入力するベクトルの類似度の算出方法はコサイン類似度とした. 学習率の調整には Adagrad を用いていて, L2 正則化を行った. ハイパーパラメータとして, フィルタ枚数 d_1 は 50, フィルタ幅 w は 4, 正則化項は 0.0004 とした.

畳み込み層が 1 つであるモデルと 2 つにスタックしたモデルの 2 種類の設定で実験を行った.

7 結果と考察

実験の結果を表 3 に示す. MAP, MRR がともに 0.6 近辺の値であることから, 提案手法が良好な判定ができたことがわかる. 畳み込み層の数が 1 つであるモデルより 2 つにスタックしたモデルのほうが MAP, MRR ともにより結果となっていることから, 繰り返し畳み込み・プーリングを行うことでよりよい補足選択ができると考えられる.

表 3 提案手法の MAP・MRR による評価

畳み込み層の数	MAP	MRR
1 層	0.6142	0.6228
2 層	0.6458	0.6581

また構築したデータセットは, 文の単語数が少ないため, 容易に単語およびフレーズを検知できたと考えられる. したがって今後は単語数を増やして様々な文に対する実験が必要であるといえる.

本稿では評価手法として, MAPとMRRを使用した, それだけでなく得られた補足を人手による評価をすることでより正確に性能を測ることができると考えられる.

今回は 1 つの補足文を選択するというタスクとしたが, 実際のコミュニティQAのユーザの回答には複数の文が存在することもある. 1 つの補足文だけでなく 2 つ以上の文を選択し, その場合の精度についても検証していきたい.

本稿では質問と回答を考慮している文を補足文としたが, 実際の文章には質問のみまたは回答のみを考慮している文も多く存在する. このような文も同様に選択することができればさらに文章としての多様性や網羅性を高め, 情報量が多く, より価値の高い情報を提供できると考えている.

また人手でデータセットを構築する場合, 時間的コストが必要となり, さらに再現性を失いやすい. 特にニュー

ラルネットを用いた手法には大量のデータセットを学習に必要とする. そのため paragraph2vec[6]に代表される教師なし学習によって質問文・回答文・補足文をベクトル化して関係性を導くことができれば人手による構築を行う必要はなく, この問題を解決できると考えられる.

8 おわりに

本稿ではコミュニティQAにおける質問と回答を考慮した補足文の選択を行った. そのために既存のデータセットから補足文を含む文を抽出し, 新たなデータセットを構築した. 加えて既存の回答選択モデルを拡張し, 補足文の選択に適用した. 今後は提案手法が有効かどうかをより正確に検証するためにより厳密なモデルの評価と多様な文を含んだデータセットへの改良と, 性能向上のため提案モデルの改良が必要である.

謝辞

本研究において, ヤフー株式会社提供の「Yahoo!知恵袋データ (第 2 版)」を利用させて頂きました. 深く感謝いたします.

参考文献

- [1] Yoon Kim: Convolutional Neural Networks for Sentence Classification, In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751, 2014
- [2] Ming Tan, Cicero dos Santos, Bing Xiang and Bowen Zhou: Improved Representation Learning for Question Answer Matching, In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 464–473, 2016
- [3] Wenpeng Yin, Hinrich Schütze, Bing Xiang, Bowen Zhou: ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs, Transactions of the Association for Computational Linguistics, vol. 4, pp. 259–272, 2016
- [4] Di Wang and Eric Nyberg. CMU OAQA at TREC 2015 LiveQA: Discovering the right answer with clues. In Proceedings of The Twenty-Fourth Text REtrieval Conference, 2015
- [5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean: Distributed representations of words and phrases and their compositionality, In Proceedings of NIPS, pp. 3111–3119, 2013
- [6] Quoc V. Le and Tomas Mikolov: Distributed representations of sentences and documents. In Proceedings of the 31th International Conference on Machine Learning, ICML 2014, pp. 1188–1196, 2014