

Answer Selection におけるトリプレットロスを用いた再学習 Re-learning with triplet loss in Answer Selection

鈴木 悠真[†] 堀内 靖雄[†] 黒岩 眞吾[†]
Yuma Suzuki Yasuo Horiuchi Shingo Kuroiwa

1. はじめに

Answer Selection (AS) とは、質問に対する複数の回答候補から、正答として適切な順に並び替える (ランク付する) タスクである[1]。一般的な AS に対するアプローチでは、質問と回答候補の埋め込み表現の距離の近い順に回答候補をランク付けする。そのため、埋め込み表現がランク付の精度に大きく影響する。そこで、本研究では文脈を考慮した上で意味を反映した埋め込み表現を得ることができる BERT[2]/RoBERTa[3]を用いた。BERT/RoBERTa を用いた AS の研究として Lasker ら[1]の研究がある。本研究では、文献[1]のモデルにおいて、さらなる精度の向上を目指してトリプレットロスを用いた再学習法を提案する。

2. 提案手法

本研究では、文献[1]のモデルにおいて、トリプレットロスを用いた再学習を行う。これは人間の学習方法を模倣したものである。人間は一度間違えた問題に対して対策をたて、次は間違えない様に学習する。機械も同じように学習させれば性能が上がると考え、埋め込み空間において、正解でないものを遠ざけ、正解であるものを近づけるトリプレットロスを用いた再学習を行う。

2.1 トリプレットロス[4]

本研究では、文献[4]で提案されている式 (1) のトリプレットロスを用いる。*anc*は基準 (anchor) となるデータでランダムに選ばれる。*pos*はアンカーと同一の категорияに属しているデータ (positive)、*neg*はアンカーと異なる categoriaに属しているデータ (negative) である。この三つ組をトリプレットと呼ぶ。

$$L(anc, pos, neg) = \max\{0, D(f(anc), f(pos)) - D(f(anc), f(neg)) + g\} \quad (1)$$

ここで、*f*は埋め込み関数であり、本稿では BERT/RoBERT を用いる。また、*D*は距離関数であり、本稿では *cos*類似度を用いた。式(1)を最小化することで、*anchor* と *positive* の埋め込み空間上での距離 $D(f(anc), f(pos))$ (以降 d_p) は小さくなる様に、*anchor* と *negative* の埋め込み空間上での距離 $D(f(anc), f(neg))$ (以降 d_n) は大きくなる様に学習される。*g*はマージンで、 d_p と d_n をどの程度離すかを表すハイパーパラメータである。

次にトリプレットの作成方法について説明する。本研究では、式 (2) で表される Semi-Hard Negative、式 (3) で表される Hard Negative を用いた。

$$\text{SemiHard-Negative: } d_p \leq d_n < d_p + g \quad (2)$$

$$\text{Hard Negative: } d_n < d_p \quad (3)$$

まず、*anc*、*pos* をランダムに選択し、 d_p を求める。次に、各式の条件を見たす d_n となるような *neg* を選択し、*anc*、*pos*、*neg* の組を作成する。Semi-hard Negative は、正解 (*pos*) より *anc* から離れていて、そこからマージン *g* 以内の不正解 (*neg*) を選択する方法である。*g* の値が小さいほど正解 (*pos*) に近い不正解 (*neg*) を選択し、*g* の値が大きいほど正解 (*pos*) から離れた不正解 (*neg*) も選択することになる。一方、Hard Negative は、正解 (*pos*) より *anc* に近い不正解 (*neg*) を選択する方法である。

2.2 提案手法の流れ

本節では、提案手法の流れを説明する。まず従来手法[1]で BERT/RoBERTa に分類層を加えて質問と正答文のペアを正解として学習 (fine-tuning) する。このとき、一般的な AS に対するアプローチと異なり、入力は質問と回答のペア ([CLS] 質問 [SEP] 回答 [SEP]) であり、質問に対し回答が正答であるスコアを出力する。このスコアを用いて、降順に回答を並び替えることで回答をランク付けする。次に、学習時に「間違えた問題」を選択し、再学習用の学習セットとする。本稿での「間違えた問題」の学習セットとは、ある質問に対して誤答よりも下位となった正答と、その正答と入れ替わって上位となった誤答を指す。提案手法での BERT/RoBERTa への入力は、従来手法と同様に質問と回答のペアであり、再学習用の学習セットを対象に、トリプレットロスを用いて再学習を行う。一般的な AS に対するアプローチと異なり、作成される埋め込み表現は、質問と回答のペアの埋め込み表現である。そのため、トリプレットロスへの入力は、ランダムに選ばれた質問と回答のペア (*anchor*)、そのペアと同じラベル (正解なら 1、不正解なら 0) のペア (*positive*)、そのペアと異なるラベルのペア (*negative*) の 3 つの埋め込み表現である。この際、3 つの質問は同じである必要はない。これにより、未知の質問と回答のペアに対しても頑健な学習が可能であると考えた。

3. 評価実験

3.1 データセット

評価実験には、AS モデルの評価によく用いられる TREC-QA[5]と WikiQA[6]を用いた。各々の構成を表 1 に示す。いずれのデータセットも日付や名称、数値等の事実を問う質問とその質問に対する複数の回答候補のペアの集合になっている。

表 1 評価実験で用いるデータセット

Dataset	Question			Candidate Answer		
	Train	Valid	Test	Train	Valid	Test
TREC-QA	93	82	100	5919	1148	1517
WikiQA	2118	296	633	20360	2733	6165

[†] 千葉大学 Chiba University

3.2 評価方法

本研究では、評価指標として、MAP (Mean Average Precision) と MRR (Mean Reciprocal Rank) を用いた。

3.3 評価実験

3.3.1 評価実験 1

提案手法の有効性を確かめるため、RoBERTa-base[7]、BERT-base[8] を[1]の手法で fine-tuning したモデルと提案手法で再学習したモデルの比較を行った。提案手法のトリプレットロスを用いた再学習では Semi-hard Negative と Hard Negative の 2 つの方法を比較した。なお、マージン g は 0.2 として実験を行った。

3.3.2 評価実験 2

マージン g の値は、式 (2) を満たす neg を変化させるため、モデルの精度に影響する。Semi-hard Negative では、一般的 $g=0.2$ が用いられることが多いが[9]、 g による精度への影響を確認するため、提案手法 Semi-hard Negative において $g = 0.1 \sim 1.9$ と変化させる実験を行なった。

3.4 結果

3.4.1 結果-1

表 2 に評価実験 1 の結果を示す。

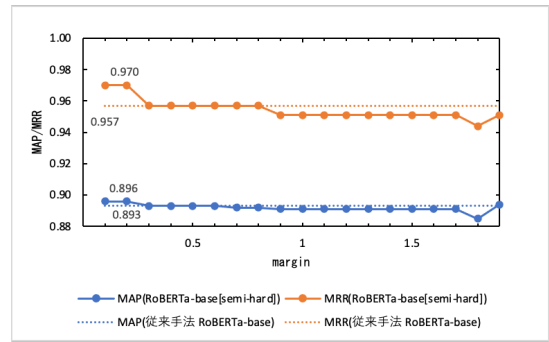
表 2 トリプレットロスを用いた再学習の評価結果

モデル	TREC-QA		WikiQA	
	MAP	MRR	MAP	MRR
従来手法 RoBERTa-base[1]	0.893	0.957	0.830	0.843
提案手法 RoBERTa-base [semi-hard]	0.896	0.970	0.845	0.857
提案手法 RoBERTa-base [hard]	0.887	0.950	0.840	0.854
従来手法 BERT-base[1]	0.821	0.877	0.801	0.809
提案手法 BERT-base [semi-hard]	0.830	0.881	0.816	0.828
提案手法 BERT-base [hard]	0.828	0.881	0.817	0.829

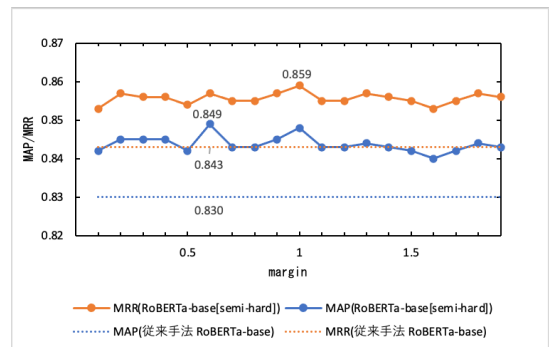
各データセットにおいて、提案手法の RoBERTa-base [semi-hard] で最も高い精度が得られた。提案手法の Semi-Hard Negative では従来手法と比べ、全ての条件で精度が向上した。また、Hard Negative では、TREC-QA の RoBERTa-base 以外で精度が向上した。これらより、提案手法の有効性が示唆された。

3.4.2 結果-2

図 1 に結果-1 で最も精度の高かった RoBERTa で g を変化させたときの、MAP、MRR を示す。図 1 (b) より、WikiQA では g によらず、再学習により MAP、MRR ともに精度が向上していることがわかる。 $g = 0.6$ や 1.0 で精度が高くなっているが、 g と精度の間に大きな相関は見られない。一方で、図 1 (a) の TREC-QA では、 g が 0.3 以上では再学習により精度が下がってしまっており、また、 g が大きくなるにつれ精度が下がる傾向も見られた。再学習に用いられたデータ数に着目してみると、WikiQA では $g = 0.2$ のときのトリプレット数が 3243 であるのに対し、TREC-QA では 865 であった。また、 $g = 1.9$ では、WikiQA が 4994、TREC-QA が 1031 と、TREC-QA において再学習データ数が極端に大きくなっているわけではなかつ



(a) : TREC-QA での実験結果



(b) : WikiQA での実験結果

図 1 RoBERTa-base において g を変化させた結果

た。一般的に用いられる $g = 0.2$ に対し、 g を変化させることによる知見は残念ながら得られなかった。

4. おわりに

本研究では、Answer Selection における解答精度向上を目的に、トリプレットロスを用いた再学習手法の提案を行った。実験の結果、再学習後のモデルを利用することで再学習前のモデル[1]と比べ、MAP、MRR 共に同等以上の精度が得られた。

今後は、再学習時のトリプレットの選び方の工夫や埋め込み関数として BERT/RoBERTa 以外を使用する手法を検討していく。

謝辞

本研究は JSPS 科研費 JP20K11860、JP21K02052 の助成を受けたものです。

参考文献

- [1] M. T. R. Laskar et al., "Utilizing Bidirectional Encoder Representations from Transformers for Answer Selection", arXiv preprint arXiv:2011.07208v1, 2020.
- [2] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", NAACL, pages 4171-4186, 2019.
- [3] Y. Liu et al., "Roberta: A robustly optimized bert pretraining approach." arXiv preprint arXiv:1907.11692, 2019.
- [4] J. Wang et al., "Learning Fine-grained Image Similarity with Deep Ranking", arXiv preprint arXiv:1404.4661, 2014.
- [5] M. Wang et al., "What is the jeopardy model? a quasi-synchronous grammar for qa.", EMNLP- CoNLL, 2007.
- [6] Y. Yang et al., "WikiQA: A Challenge Dataset for Open-Domain Question Answering.", EMNLP- CoNLL, pages 2013-2018, 2015.
- [7] Facebookresearch, RoBERTa: A Robustly Optimized BERT Pretraining Approach. (引用日: 2022 年 06 月 20 日) <https://github.com/facebookresearch/fairseq/tree/main/examples/roberta>.
- [8] Google-research, BERT. (引用日: 2022 年 06 月 20 日) <https://github.com/google-research/bert>.
- [9] F. Schroff et al., "FaceNet: A Unified Embedding for Face Recognition and Clustering", arXiv preprint arXiv:1503.03832v3, 2015.