Longformer によるマルチホップ質問応答手法の比較

岡田直樹　　松崎拓也

東京理科大学 理学部第一部 応用数学科

1417023@ed.tus.ac.jp, matuzaki@rs.tus.ac.jp

1　はじめに

与えられた文書をもとに質問に答えるタスクである機械読解は、BERT [1] をはじめとする transformer ベースの事前学習モデルの登場以降、研究が活発になり、大きな進歩を遂げてきた。機械読解は単一文書からの抽出を行うタイプとマルチホップ型に大きく分けられる。SQuAD [2] に代表される前者のタスクは、単一の文書を参照して質問に答えることに焦点を当てた抽出型タスクであり、すでに人間のスコアを超えるモデルが提案されている。しかし、このようなタスクでは単一の文書しか参照する必要がないため、比較的容易なタスクとなっており、応用範囲は限定的である。一方、後者のマルチホップ型タスクは、単一もしくは複数の文書中から、複数箇所にある根拠を見つけなければ答えを導き出せないため、難易度が高く、より応用範囲が広いタスクであり、HotpotQA [3] や WikiHop [4] といったデータセットが開発されている。

本研究で扱う HotpotQA では、複数の文書が与えられ、その中から回答に必要な文書を見つけ出し、回答とその根拠となる文を見つけ出す必要がある。これまで提案されている手法は、グラフ構造を用いた手法 [5, 6, 7] とグラフ構造を用いずに BERT 等の事前学習モデルを利用した手法 [8, 9] に分けられる。

その中で、Beltagy らが提案した手法 [10] は、従来の transformer モデルよりも長い文書を扱える Longformer を利用して、複数の文書を同時に入力し、回答と根拠文選択を単一のモデルで行う。この手法は、根拠文選択と回答抽出を独立に行うことができる。ただし独立に行った場合には、不要な文書中の文やトークンも候補に含めて根拠文選択と回答抽出を行うことになる。実際に人が推論を行う際には、答えに関係する文書を見つけ出し、その文書にのみ注目して答えを見つけ出すだろう。この様子を再現し、不要な文を事前に取り除くことで、よりモデルの性能が上がる可能性がある。Beltagy ら [11] や Groeneveld ら [12] は、根拠文選択についてこれに似たアイデアを用いている。ただし、文書や文を誤って取り除いた場合には回答を導けないため、性能に悪影響を及ぼすことも考えられる。

そこで本研究では、HotpotQA に対して Longformer を利用する手法において、訓練時とテスト時に、いくつかの方法で根拠文と回答の抽出範囲を段階的に絞り込むことが、回答と根拠文を抜き出す精度にどのような効果を及ぼすかを検証した。その結果、段階的に絞り込みを行わないモデルよりも、回答と根拠文抽出の精度が向上することが確認できた。

Question: Where is the company that Sachin Warrier worked for as a software engineer headquartered?
Document 1: [1]Sachin Warrier is a playback singer and composer in the Malayalam cinema industry from Kerala. [2]He became notable with the song "Muthuchippi Poloru" from the film Thattathin Marayathu. [3]He made his debut with the movie Malarvaadi Arts Club. [4]He was working as a software engineer in Tata Consultancy Services in Kochi. [5]Later he resigned from the job to concentrate more on music. [6]His latest work is as a composer for the movie Aanandam.
Document 2: [7]Tata Consultancy Services Limited (TCS) is an Indian multinational information technology (IT) service, consulting and business solutions company Headquartered in Mumbai, Maharashtra. [8]It is a subsidiary of the Tata Group and operates in 46 countries.
Answer: Mumbai
Supporting facts: [4],[7]

図 1　HotpotQA の問題例（gold 文書のみ）

2 背景

本項では、使用するデータセットの HotpotQA とベースとなる手法を説明する。

2.1 HotpotQA

本研究では、マルチホップ質問応答データセットである HotpotQA の Distractor setting で実験を行う。Distractor setting は答えを導くのに不要な文書を含む 10 個の文書から必要な文書を見つけ出し、回答を行うタスクである。文書は Wikipedia の記事から収集されており、質問に対し回答に必要な 2 つの文書（gold 文書）と bigram tf-idf によって収集された回答に必要の無い 8 つの文書（distractor 文書）が与えられ、その中から回答とその根拠となる文（根拠文）を抜き出すことが求められる。2 つの gold 文書のそれぞれに一文以上の根拠文が存在し、回答のタイプとしては yes/no タイプと文書から回答を抜き出すスパンタイプがある。また、必要な推論のタイプとしては bridge タイプと compare タイプがある。bridge タイプは、質問に対して、文書 1 →文書 2 のように文書をまたぐ推論を行なって根拠を見つけ回答する問題となっている。図 1 の例では、質問に直接関連する事柄として、文書 1 で「Sachin Warrier が Tata Consultancy Service で働いていた」ことが分かり、さらに文書 2 で「Tata Consultancy Service の本社が Mumbai にある」ことが分かり、答えの「Mumbai」を導ける。compare タイプは、二つのエンティティを比較して回答する問題となっている。

2.2 Longformer

従来の Transformer モデルでは、必要な記憶量と計算量が入力系列の長さの 2 乗のオーダーで増加するため、長い系列を扱うことができなかった。そのため、長い系列を扱う際には、文章の一部を切り捨てるか、文章をいくつかに分割する必要があり、文章全体の情報を失わないように複雑なモデルを構築する必要があった。

Longformer はこのような問題を解決するために以下の 3 つの attention パターンを導入している。Sliding window attention は、各トークンを取り囲む一定サイズの window 中でのみ attention をとる。Dilated sliding window は、Window attention に固定のギャップを導入することで、計算量を増加させることなく、より広い情報を捉えるようにした attention である。Global attention は、[CLS] トークンや質問文中のトークンなどの、特殊な役割をもつトークンに与え、タスク固有の表現を学習するために導入する attention である。

このうち、window attention 層と dilated window attention 層を積み重ねることで、より広い文脈を計算量を抑えながら捉えることのできるモデルとなっている。

2.3 Longformer による HotpotQA

本研究でベースとしている、Beltagy らが提案した HotpotQA に対する Longformer を利用した手法を説明する。HotpotQA のタスクは回答スパン抽出と根拠文予測の 2 つのタスクを含むが、Beltagy らの手法ではこれらを一つのモデルで行う。2 つの gold 文書と 8 つの distractor 文書の計 10 個の文書を連結し、さらに質問と繋げて入力とする。入力のフォーマットとしては、"[CLS] [q] question [/q] [p] $\text{sent}_{1,1}$ [s] $\text{sent}_{1,2}$ [s] ... [p] $\text{sent}_{2,1}$ [s] $\text{sent}_{2,2}$ [s] ..." のようになっている。ここで $\text{sent}_{i,j}$ は i 番目の文書の j 番目の文であり、[p]・[s]・[q]・[/q] はそれぞれ文書・文・質問の始まり・質問の終わりを表す特殊トークンである。また、先頭の [CLS] トークンに対する出力ベクトルは、回答タイプ分類（yes/no or スパン）に用いる。Longformer の global attention は、[CLS], [s], [p] の特殊トークンと [q] から [/q] の間の質問を表す部分のトークンに与えられる。

文書選択、根拠文選択、回答抽出の方法は以下の通り。10 文書それぞれが gold 文書か否かを判定する文書選択タスクは、文書の始まりを表す特殊トークンに対する出力ベクトルに、根拠文選択のタスクは、文の始まりを表す特殊トークンに対する出力ベクトルに 2 層のフィードフォワードネットワークを適用して得られるスコアにクロスエントロピー損失を適用して訓練を行う。回答抽出タスクについては、Devlin ら [1] の SQuAD v1.1 への BERT の適用手法に [CLS] トークンによって回答タイプを予測する機構を加えて行う。具体的には文書の各トークンの出力ベクトルに線形層を適用して、回答スパンの開始・終了位置となることに対するスコアとし、[CLS] トークンの出力にはフィードフォワードネットワークを適用してクロスエントロピー損失をとる。モデル全体の訓練には、回答スパン・回答タイプ分類・文選択・文書選択のそれぞれの損失の線形和を全体の損失として、マルチタスク学習の形で

表 1　訓練手法における選択候補の絞り込み

	根拠文選択	回答抽出
train1	なし	なし
train2	あり	なし
train3	あり	あり（文書）
train4	あり	あり（根拠文）
train5	あり*	あり（文書）*
train6	あり*	あり（根拠文）*

*は候補とする部分のみを Longformer に入力

行う。

3　手法

本研究では、上記の Longformer モデルを使った手法をベースに異なる訓練・テスト方法を用いて、それらの差を調べた。Longformer への入力において、文を表す特殊トークン [s] は各文の先頭に付加し、全体の損失は、2.3 項と同様に定義する。ただし、回答抽出の損失については、文書中に答えとなる単語が複数出現することがあるため、以下の関数を用いた [13]。

$$-\log\left(\frac{\sum_{k\in A} e^{s_k}}{\sum_{i=1}^{n} e^{s_i}}\right) \tag{1}$$

ここで、A は回答トークンの集合、n は全トークン数を、s_i は i 番目のトークンに対する線形層からの出力を表す。

3.1　訓練方法

訓練の方法として以下の train1-6 の 6 つの手法を比較した。いずれにおいても、目的関数は全てのタスクの損失を足し合わせたものとする。また、事前訓練済みモデルとしては Longformer-base モデルを用いる。

train1　2.3 項で説明した手法で、それぞれのタスクで出力ベクトル全体を用いる。

train2　根拠文選択の訓練を行う際、文書選択で gold 文書であると判定された文書中の文のみを候補とする。回答抽出の範囲は train1 と同様。

train3　根拠文選択の訓練は train2 と同じ方法で行う。回答抽出の訓練を行う際、文書選択で gold 文書だと判定された文書のトークンのみを候補とする。

train4　根拠文選択の訓練は train2 と同じ方法で行う。回答抽出の訓練を行う際、文選択で根拠文であると判定された文のトークンのみを候補とする。

train5　根拠文選択・回答抽出の訓練の際、ともに文書選択で gold 文書と判定された文書と質問のトークンのみを連結したものを Longformer へ再度入力して、その出力ベクトルに対して根拠文選択と回答抽出の訓練を行う。

train6　根拠文選択の訓練は train5 と同じ方法で行う。回答抽出の訓練を行う際、文選択で根拠文であると判定された文と質問のトークンのみを連結したものを Longformer へ再度入力して、その出力ベクトルに対して訓練を行う。

絞り込んだ文が回答を含まない場合には、回答抽出の損失は計算せずに他のタスクの損失のみを考えた。

Question: Which current CBS NBA analyst played for the PORTLAND TRAILBLazers along with former UNLV teammate Stacey Augmon?

Document 1: [1]Gregory Carlton "Greg" Anthony (born November 15, 1967) is an American former National Basketball Association (NBA) player and is currently a television analyst for CBS Sports. [2] Anthony also contributes to Yahoo! Sports as a college basketball analyst and serves as a co-host/analyst on SiriusXM NBA Radio.

Document 2: [3]The 1998-99 NBA season was the 29th season for the Portland Trail Blazers in the National Basketball Association. [4] During the offseason, the Blazers signed free agents Jim Jackson and Greg Anthony, who would reunite with his former UNLV teammate Stacey Augmon. [5] Portland got off to a fast start winning 15 of their first 18 games, and went 35-15 in the lockout-shortened season, earning their fourth Pacific Division title and the first since 1991-92. [6] Their record qualified them for the #2 seed in the Western Conference. [7] The team earned their 17th straight trip to the playoffs, and 22nd in 23 years. [8] Head coach Mike Dunleavy was named Coach of The Year.

Document 3: · · · [9] The team also acquired Stacey Augmon and Grant Long from the Atlanta Hawks, but later on sent Augmon to the Portland Trail Blazers for Aaron McKie at midseason. · · ·

Answer: Aaron McKie(train1-test1), Greg Anthony(train4-test4)

Supporting facts: [1],[4],[7],[9](train1-test1), [1],[4],[7](train4-test4)

図 2　絞り込むことにより正答する例

3.2　テスト方法

train1-6 と同じ絞り込みを行う方法をそれぞれ test1-6 とする。文書選択についてはスコアの上位 2 文書のみを選択した。

表 2　テスト文書数：全文書

		test1		test2		test3		test4		test5		test6	
文書数	訓練方法	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup
全文書	train1	71.95	80.68	71.95	80.90	71.89	80.90	69.49	80.90	74.54	81.97	72.58	81.97
	train2	72.65	80.50	72.65	84.04	72.45	84.04	72.14	84.04	74.64	83.81	72.23	83.81
	train3	49.08	79.12	49.08	83.83	71.96	83.83	71.59	83.83	74.22	79.64	66.51	79.64
	train4	71.12	79.48	71.12	83.78	71.77	83.78	71.34	83.78	74.24	83.15	71.57	83.15
	train5	69.73	58.46	69.73	60.59	69.67	60.59	55.86	60.59	**75.87**	84.33	73.04	84.33
	train6	66.02	55.04	66.02	56.72	66.94	56.72	57.20	56.72	74.78	**84.69**	73.79	**84.69**
5 文書	train1	70.87	74.73	70.87	80.75	70.35	80.75	69.20	80.75	**74.30**	82.11	72.06	82.11
	train2	70.38	66.01	70.38	80.95	70.53	80.95	69.67	80.95	73.83	81.52	71.08	81.52
	train3	28.13	63.42	28.13	79.89	69.20	79.89	68.41	79.89	72.16	80.10	66.08	80.10
	train4	68.33	68.11	68.33	81.56	70.21	81.56	69.51	81.56	73.87	**82.83**	71.97	**82.83**
	train5	65.22	55.27	65.22	58.92	65.29	58.92	53.97	58.92	71.93	81.90	69.42	81.90
	train6	59.73	41.89	59.73	44.22	62.65	44.22	56.23	44.22	73.08	82.33	72.80	82.33

表 3　テスト文書数：5 文書

		test1		test2		test3		test4		test5		test6	
文書数	訓練方法	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup	Ans	Sup
全文書	train1	73.54	81.71	73.54	81.71	73.57	81.71	71.79	81.71	75.64	83.14	73.39	83.14
	train2	74.03	84.70	74.03	85.22	74.36	85.22	73.96	85.22	75.67	85.01	73.14	85.01
	train3	50.98	83.89	50.98	84.59	73.89	84.59	73.15	84.59	75.57.	80.85	67.58	80.85
	train4	72.97	84.42	72.97	85.04	73.35	85.04	72.92	85.04	75.28	84.34	72.68	84.34
	train5	72.98	77.34	72.98	78.27	73.34	78.27	69.12	78.27	**76.88**	85.70	73.96	85.70
	train6	70.23	76.81	70.23	78.07	71.22	78.07	68.84	78.07	75.82	**86.05**	74.69	**86.05**
5 文書	train1	74.62	85.06	74.62	85.14	74.62	85.14	73.99	85.14	**77.30**	85.04	74.90	85.04
	train2	74.65	84.56	74.65	85.91	74.83	85.91	74.52	85.91	76.75	85.18	73.92	85.18
	train3	32.90	83.96	32.90	85.52	74.24	85.52	73.82	85.52	75.85	84.09	69.16	84.09
	train4	73.77	84.88	73.77	85.79	74.94	85.79	74.66	85.79	76.76	85.72	74.84	85.72
	train5	71.50	77.57	71.50	78.80	71.72	78.80	68.49	78.80	75.80	**85.84**	73.11	**85.84**
	train6	69.67	76.13	69.67	77.93	70.74	77.93	67.24	77.93	76.60	86.33	76.49	86.33

4　実験

実験は、3 節で述べた手法で全文書を使って訓練したモデルと、5 文書に絞って訓練したモデルについて、全文書でテストした場合（表 2）と 5 文書でテストした場合（表 3）について調べた。5 文書への絞り込みには、Beltagy ら [11] の two-stage モデルと同様に、段落選択のみで訓練したモデルを用いた。

データは、公開されている訓練セット（90,447 問）のうち 80,000 問を訓練データ、残りを開発データとし、開発セットをテストデータとした。表の Ans は回答、Sup は根拠文の F1 スコアである。

表 2 から、根拠文選択を行う際に、train2-4 のように文書選択により候補を絞り込み、テスト時にも同様にすることで、性能が向上することが確認できた。回答抽出については、train3,4 のような候補の絞り込みでは性能の向上は見られなかった。一方 train5,6 のように、絞り込んだ文のみを入力すると、根拠文選択に関しては train2-4 と同等の効果しか得られなかったが、回答抽出はこの方法で行うことで最も良い結果が得られた。絞り込んだ文のみを入力する方法は、テスト時にのみ適用した場合でも性能の向上が見られる。また 5 文書へ絞ると性能は向上するが、訓練に 5 文書しか使わなかった場合、テストに全文書を使うと、性能が下がる傾向が見られることから、テスト時よりも厳しい設定で訓練する方が頑健性が保たれると分かる。

図 2 に、train1-test1 で誤答し、train4-test4 では正答した例を示す。train1-test1 は絞り込みを行わないために、文書 1,2 を gold 文書と判定したが、文書 3 から余分に根拠文を選択し、そこから誤った回答を抽出している。このような誤答は、数字や人名などが回答で、似た特性を持つ単語が文書中に複数存在する問題で見られた。このことからも、段階的な絞り込みの有効性が確認できる。

5　おわりに

本研究では、HotpotQA に対して Longformer を適用する際に、段階的な絞り込みを行うことによる効果を調べた。結果として不要な文を取り除いていくことで、一定の効果が得られることが確認できた。

参考文献

[1]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)*, 2019.

[2]Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[3]Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

[4]Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, Vol. 6, pp. 287–302, 2018.

[5]Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[6]Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[7]Ye Deming, Lin Yankai, Liu Zhenghao, Liu Zhiyuan, and Sun Maosong. Multi-paragraph reasoning with knowledge-enhanced graph neural network. *arXiv:1911.02170*, 2019.

[8]Michael Glass, Alfio Gliozzo, Rishav Chakravarti, Anthony Ferritto, Lin Pan, G P Shrivatsa Bhargav, Dinesh Garg, and Avi Sil. Span selection pre-training for question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[9]Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. Multi-hop reading comprehension through question decomposition and rescoring. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[10]Iz Beltagy, Matthew E. Peters, and Cohan Arman. Longformer: The long-document transformer. *arXiv:2004.05150v1*, 2020.

[11]Iz Beltagy, Matthew E. Peters, and Cohan Arman. Longformer: The long-document transformer. *arXiv:2004.05150v2*, 2020.

[12]Groeneveld Dirk, Khot Tushar, Mausam, and Sabharwal Ashish. A simple yet strong pipeline for hotpotqa. *arXiv:2004.06753*, 2020.

[13]Christopher Clark and Matt Gardner. Simple and effective multi-paragraph reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018.